Doctoral theses at NTNU, 2024:348

Ahmad Hassanpour

Modelling and Preserving Privacy in Online Social Networks

Doctoral thesis

NTNU
Norwegian University of Science and Technology
Thesis for the Degree of
Philosophiae Doctor
Faculty of Information Technology and Electrical
Engineering
Dept. of Information Security and
Communication Technology

NTNU
Norwegian University of
Science and Technology

Ahmad Hassanpour

Modelling and Preserving Privacy in Online Social Networks

Thesis for the Degree of Philosophiae Doctor

Gjøvik, September 2024

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Dept. of Information Security and Communication Technology

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of Science and Technology

Thesis for the Degree of Philosophiae Doctor

Faculty of Information Technology and Electrical Engineering
Dept. of Information Security and Communication Technology

ISBN 978-82-326-8286-7 (printed ver.)
ISBN 978-82-326-8285-0 (electronic ver.)
ISSN 1503-8181 (printed ver.)
ISSN 2703-8084 (online ver.)

Doctoral theses at NTNU, 2024:348

Printed by NTNU Grafisk senter

"I don't have any particular recipe. It is like being lost in a jungle and trying to use all the knowledge that you can gather to come up with some new tricks, and with some luck, you might find a way out."

(Maryam Mirzakhani.)

Declaration of Authorship

I, Ahmad Hassanpour, hereby declare that this thesis and the work presented in it is entirely my own. Where I have consulted the work of others, this is always clearly stated.

Signed:

(Ahmad Hassanpour)

Date:

Summary

In the digital age, online social networks (OSNs) have become essential platforms for communication, information sharing, and social interaction. However, this widespread use has introduced significant privacy concerns. Users often share personal information without a full understanding of the potential risks, which are compounded by the complex and dynamic nature of digital interactions. These interactions span multiple platforms, making privacy management increasingly challenging. The ease with which personal data can be accessed and potentially misused highlights the urgent need for more effective privacy protection mechanisms.

One critical aspect of this problem is the static nature of current privacy settings on most OSNs, which require manual adjustments. Users' privacy needs and preferences evolve over time and with varying types of content, but current systems do not adequately accommodate these changes. This often results in a mismatch between the desired level of privacy and the actual settings applied. There is a clear need for adaptive privacy settings that can respond to users' changing requirements more intuitively and efficiently. The primary goal of this thesis is to develop more accurate methods for measuring privacy leakage and to explore how ML models can mitigate, intensify, or alert users about potential privacy breaches. It aims to develop a conceptual framework for measuring privacy leakage on OSNs, helping users understand their privacy settings and associated risks. This framework will also provide a basis for users to make more informed decisions about their data sharing practices.

This research also examines the role of data linkability in privacy breaches within and across different OSNs. By analyzing how the linkability of data—where separate pieces of information can be connected to form a comprehensive user profile—affects privacy leakage scores, this work provides valuable insights into managing privacy risks. The study further investigates methods for continuously adjusting user privacy settings in real-time without compromising privacy, evaluating the effectiveness of deep learning models in automating privacy settings.

The introduction of advanced technologies like machine learning (ML) in managing privacy settings offers both opportunities and challenges. Generative AI (GAI) has recently gained popularity, capable of uncovering and potentially exposing sensitive information shared on OSNs. GAI models can extrapolate and recreate sensitive data, raising concerns about security and privacy. As these models become more advanced, they enhance user experience by providing personalized content but also pose risks of unintentional data leakage. This duality necessitates a thorough exploration of the implications of GAI on data privacy and security. This thesis explores the potential for private information disclosure through public data and GAI technologies, with a focus on facial features. It addresses the ”eyes-to-face” problem, where only the eyes are visible, and assesses the potential for GAI technologies to reconstruct the rest of the face, thereby compromising individual privacy. This analysis highlights the privacy vulnerabilities inherent in sharing partial biometric data and proposes methodologies to mitigate such risks.

Moreover, while ML models can provide more sophisticated and adaptable privacy controls, they also introduce issues related to transparency, fairness, and ethical considerations. The algorithms behind these models can be opaque, making it difficult for users to understand how their privacy is managed. Additionally, there is a risk that these models may perpetuate biases, leading to unfair or discriminatory outcomes. We investigated the intricate interactions between privacy, accuracy, and fairness in image classification tasks. Our study highlights a consequential trade-off between privacy or utility and fairness, as applying the generalization techniques.

In conclusion, the increasing use of online social networks (OSNs) has significantly amplified privacy concerns due to the dynamic and multi-platform nature of digital interactions. This thesis underscores the inadequacy of static privacy settings and the necessity for adaptive mechanisms that cater to the evolving needs of users. By developing a conceptual framework for measuring privacy leakage and examining the role of data linkability, this research provides critical insights into managing privacy risks. Furthermore, the exploration of advanced technologies like machine learning and generative AI reveals both the potential and the challenges of these tools in enhancing privacy controls. This work emphasizes the importance of balancing privacy, accuracy, and fairness, proposing innovative methods to mitigate privacy vulnerabilities, especially in the context of biometric data and image classification. The findings advocate for more transparent, fair, and ethical approaches in privacy management, paving the way for safer digital environments.

Acknowledgments

I would like to express my sincere gratitude to my principal supervisor Associate Professor Bian Yang, for his commitment, guidance, support, and advice throughout the entire journey of this research and writing of the thesis. He also permitted me to explore additional research areas of my interest and in various collaborations which have enabled me to establish my personal relationship within the academic community. In his humility, ways of assessment and dedication, my supervisor has become my mentor. My appreciation also extends to Professors Julian Fierrez and Christoph Busch, my co-supervisors for their pieces of advice in the course of this journey.

I would like to thank my co-authors and students for the collaborations that have resulted in publications included in this thesis. Through the collaborations and discussions with you, I was able to dive deeply into the research topics that I am passionate about. I would also like to thank the PriMA project and its coordinator Professor Raymond Veldhuis. Furthermore, I wish to extend my gratitude to my wonderful office mates and the academic and administrative staff of the Department of Information Security and Communication Technology, and the entire Norwegian society.

I remain very grateful to my wife, Yasamin who encouraged me during my low moments in this PhD journey and cushioned me with peace of mind. My deepest gratitude also goes to my family and my friends for supporting me throughout this journey.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the digital age, the proliferation of online social networks (OSNs) has revolutionized the way individuals interact, share information, and form connections across the globe. While these platforms offer unprecedented opportunities for communication and collaboration, they also raise significant concerns regarding privacy and data security. The vast amounts of personal information shared on these networks expose users to various privacy risks, from identity theft to unauthorized data mining and surveillance. Consequently, understanding and preserving privacy on OSNs is not only crucial for protecting individual users but is also imperative for maintaining the trust and integrity of these digital platforms.

This thesis addresses the critical challenge of modeling and preserving privacy in online social networks. It stems from the observation that while users benefit greatly from these platforms, they often lack control over their personal information and are unaware of how it may be used or exposed. The primary aim of this research is to develop robust models that can accurately measure privacy risks and implement effective mechanisms to protect users' data. By enhancing privacy on OSNs, this work seeks to empower users, giving them greater control over their digital personas and the security of their information.

The significance of this research is underscored by the dynamic nature of both technology and user behavior. Online social networks are continually evolving, with new features and complexities that can obscure how data is handled and shared. Additionally, users' perceptions of privacy and their interaction patterns are constantly changing, influenced by societal trends and personal experiences. This thesis proposes adaptive models that not only respond to these changes but also anticipate future developments in OSN functionalities and user engagement. Such proactive measures are vital for sustaining privacy and ensuring that the benefits of social networks do not come at the expense of users' security.

Furthermore, the methodology adopted in this research integrates multidisciplinary approaches, incorporating insights from computer science, behavioral studies, and ethical considerations. Through a series of focused research questions, this thesis explores different aspects of privacy in OSNs, including the measurement of privacy leakage, the linkability of user posts, the potential for exposing hidden private information, the continuous adjustment of privacy settings, and the balance between privacy and fairness in the application of deep learning models. By addressing these complex challenges, the thesis contributes valuable knowledge and practical solutions that enhance our understanding of privacy in online social networks and offer significant implications for users, developers, and policymakers alike.

1.1 Motivation and problem description

In the digital age, online social networks (OSNs) have become a central part of our daily lives, offering a platform for communication, information sharing, and social interaction. However, this widespread use of OSNs raises significant privacy concerns. Personal information is frequently shared on these platforms, sometimes without the users' full understanding of the potential risks involved. This situation is exacerbated by the complex and ever-evolving nature of digital interactions, where data is not only shared but also linked across multiple platforms, making privacy management increasingly challenging. The ease with which personal data can be accessed and potentially misused underscores the need for more effective privacy protection mechanisms.

Another critical aspect of this problem is the dynamic nature of privacy itself. As users navigate through different stages of life and engage with varying types of content, their privacy needs and preferences change. However, current privacy settings on most OSNs are static and require manual adjustments, which can be cumbersome and not always intuitive for all users. This gap often leads to a mismatch between the desired level of privacy and the actual privacy settings applied.

Furthermore, with the introduction of advanced technologies like machine learning in managing privacy settings, questions arise regarding the using these technologies, and their impact on user privacy. The integration of machine learning models in automating privacy settings presents a double-edged sword. While these models offer the potential for more sophisticated and adaptable privacy controls, they also introduce new challenges in terms of transparency, fairness, and ethical considerations. The algorithms driving these models can sometimes be opaque, making it difficult for users to understand how their privacy is being managed. Additionally, there is a risk that these models may inadvertently perpetuate biases, leading to unfair or discriminatory outcomes.

Generative AI, a technology that trains on extensive datasets to create new data mimicking the original, has surged in popularity recently. GAI models possess the capability to uncover and potentially expose sensitive information that has been previously shared on OSNs. This ability to extrapolate and recreate sensitive data highlights growing concerns about the security and privacy implications of using generative AI in various online platforms. As these models become more advanced, they not only enhance user experience by providing more personalized content but also raise critical questions about data privacy and the potential risks associated with unintentional data leakage.

1.2 Research aim, objectives and scope

The primary goal of this thesis is to develop more accurate methods for measuring privacy leakage and to explore how machine learning models can mitigate, intensify, or alert users about potential privacy breaches. This research will concentrate on several critical objectives to tackle the privacy issues arising from the evolution of OSNs as discussed below.

Firstly, it aims to develop a conceptual framework for accurately measuring privacy leakage on OSNs. This framework will help users gain a clearer understanding of their privacy settings and the potential risks associated with their online activities. It will also provide a basis for users to make more informed decisions about their data sharing practices.

The second objective is to examine the role of data linkability in privacy breaches both within and across different OSNs. This investigation will delve into the implications of interconnected structured and unstructured digital footprints. Specifically, we will analyze how the linkability of data—where separate pieces of information can be connected to form a comprehensive profile of a user—can either increase, decrease, or leave unchanged the privacy leakage score.

Third, this thesis aims to investigate methods for continuously adjusting user privacy settings in real-time, without compromising the privacy itself. This includes evaluating the effectiveness of deep learning models in automating privacy settings. By accomplishing these objectives, the research will contribute to the development of more robust and user-friendly privacy management tools for OSNs, enhancing user trust and security online.

In addition, this thesis will explore how private information can be disclosed through the use of public data and GAI technologies, enhancing our understanding of both the capabilities and the associated risks of these technologies. Specifically, this thesis will focus on facial features, which are considered sensitive information due to their classification as biometric data. The unique aspect of this study is the investigation of the ”eyes-to-face” problem, where users of OSNs may choose to obscure most, but not all, of their facial features. We will analyze scenarios in which only the eyes are visible, assessing the potential for GAI technologies to reconstruct the rest of the face and thereby compromise individual privacy. This analysis will not only highlight the privacy vulnerabilities inherent in sharing partial biometric data but also propose methodologies to mitigate such risks, thus offering a more comprehensive perspective on privacy management within digital environments.

1.3 Research questions

Based on the research aim, objective, and motivation, five research questions were formulated to guide this thesis work. The research questions are outlined as follows:

Research question 1 (RQ1): How to measure privacy leakage in online social networks?

OSNs often expose significant amounts of sensitive data. Users frequently, albeit inadvertently, share their private information without fully understanding the associated privacy risks. It is essential for users to be well-informed about their privacy levels and understand their position on a privacy scale. The aim of this research question is to develop a framework capable of evaluating the privacy impact of each user action within an OSN. This framework would adjust privacy settings in alignment with the user's preferred privacy limits.

Research question 2 (RQ2): Does the linkibility of a user's posts influence privacy breaches across or within online social networks?

In an age where digital footprints are sprawling and often interconnected, understanding the implications of linked posts is crucial for developing better privacy protection strategies and tools. The purpose of this research question is to explore whether the ability to link various posts by a single user across different platforms or within the same platform contributes to an increased risk of privacy violations. This research question underscores the critical need for robust privacy measures in the face of the growing interconnectedness of user data.

Research question 3 (RQ3): Is it possible to disclose hidden private information using already published data?

It seeks to explore the extent to which publicly available data can be utilized to uncover private information that an individual has not directly disclosed. One method involves examining the linkability of different data segments; by connecting separate pieces of publicly shared information, it might be possible to deduce private details not directly disclosed. But the purpose of this research question is to use of GAI technologies and investigate their impact in disclosing private information using already published data. These advanced algorithms can synthesize and interpret available data to potentially uncover hidden private information. This extension of the research is critical in understanding the depth to which modern technology can penetrate privacy barriers.

Research question 4 (RQ4): How to adjust privacy user settings while preserving privacy?

This inquiry delves into developing methods for dynamically updating a user's privacy settings on social media platforms, ensuring these adjustments do not compromise their privacy. This is particularly vital because users' privacy needs and the digital landscape are constantly evolving. As social media becomes more integrated into daily life, users often struggle to manually update their settings in response to new features, privacy risks, or changes in their own privacy preferences. Effectively addressing this question is important for enhancing user trust and safety on these platforms. Therefore, it aims to provide a more adaptive, user-centric approach to privacy management, reducing the burden on users to continuously monitor and adjust their settings. By automating this process while ensuring robust privacy protection, the research could lead to significant advancements in how social networks handle user data, potentially setting new standards for privacy and user control in the digital world.

Research question 5 (RQ5): To what degree are the deep learning models employed in automating privacy on OSNs balance privacy concerns and fairness?

Deep learning models are increasingly employed in assessing and adjusting users' privacy settings, playing a crucial role in how personal data is managed on various platforms. However, there is an uncertainty surrounding the efficacy of these models in striking a balance between maintaining user privacy and ensuring fairness in data handling and algorithmic decision-making. This concern raises important questions about the transparency, bias, and ethical implications of these AI-driven systems. It's essential to examine not only how these models process and protect sensitive information, but also how they make decisions that could potentially affect user experiences and rights. The impact of these models extends beyond individual privacy, influencing broader issues such as data discrimination and equitable access to digital services. Understanding and improving the balance between privacy and fairness in deep learning models is critical for fostering trust in digital ecosystems and ensuring that advancements in AI are both responsible and beneficial to all users. This exploration could lead to the development of more robust, ethical AI systems that respect user privacy while delivering fair and unbiased outcomes.

1.4 Background

This section presents relevant background and an overview of the thesis to facilitate a better understanding of the remaining aspect of it. Other useful discussions about the study framework, and aspects of the various approaches that were adopted in this work, have also been presented.

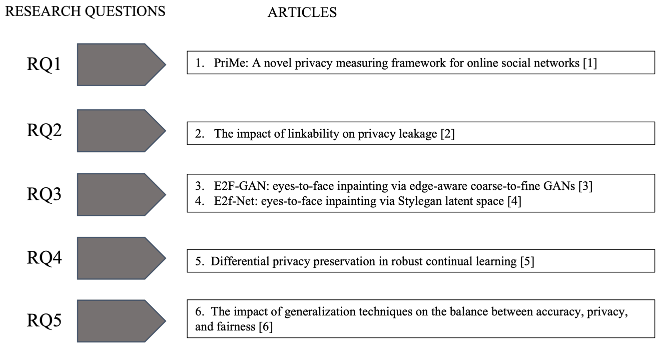

RESEARCH QUESTIONS
ARTICLES
RQ1
1. PriMe: A novel privacy measuring framework for online social networks [1]
RQ2
2. The impact of linkability on privacy leakage [2]
RQ3
3. E2F-GAN: eyes-to-face inpainting via edge-aware coarse-to-fine GANs [3]
4. E2f-Net: eyes-to-face inpainting via Stylegan latent space [4]
RQ4
5. Differential privacy preservation in robust continual learning [5]
RQ5
6. The impact of generalization techniques on the balance between accuracy, privacy, and fairness [6]

Figure 1.1: Research questions and their mapping to the articles.

1.4.1 Measuring privacy in online social networks

In the realm of privacy management within OSNs, two predominant methodologies emerge: statistical-based and machine learning (ML)-based approaches. Statistical-based methods hinge on two principal attributes: (i) the sensitivity of the information divulged and (ii) the extent of its visibility within the network framework. These methodologies typically operate on dichotomous (binary-valued) or polytomous (multivalued) variables, or a fusion of both. The essence of these methods is the computation of a privacy score, derived from the amalgamation of partial privacy scores for individual profile elements, such as email or phone number. This scoring system hinges on both the intrinsic sensitivity of the data point and the degree of exposure afforded by the user's privacy configurations.

In contrast, ML-based models primarily focus on the privacy assessment of unstructured data types, such as text and photographs. The foundational work in this domain was introduced by Maximilian et al. (2009), which proposed a pioneering framework for the calculation of privacy scores in OSNs. This framework was predicated on the evaluation of two core components: sensitivity and visibility of profile items. Sensitivity assessment is a nuanced process, given the diverse interpretations and legal definitions, such as those outlined in the GDPR. This complexity is heightened by the variable levels of sensitivity that different data types might hold for individual users. The notion of sensitivity in this context encapsulates the degree of risk associated with the public disclosure of user attributes, escalating as the sensitivity augments.

The concept of visibility, as delineated in the framework, is contingent on the user's locational dynamics within the network topology. The potential for private information leakage escalates with the user's network connectivity, whether direct or indirect. For instance, the sharing of a birth date deemed private by a user with their immediate circle can inadvertently lead to broader dissemination through that circle. The probability of such information leakage is a function of both the individual's visibility (i.e., the extent of interest in the user's information within the network) and the visibility of their connections. Hence, the framework underscores that a higher propensity for information leakage is correlated with increased visibility of both the user and their immediate network contacts.

1.4.2 Continual learning

Continual learning (CL), also referred to as lifelong learning, is a dynamic approach in machine learning where the model is designed to learn continuously, accumulating knowledge and adapting to new data over time. This approach is crucial for developing AI systems that can operate effectively in real-world environments, which are constantly changing and presenting new challenges. The core objective of continual learning is to mitigate the problem of catastrophic forgetting, which occurs when a neural network learns new information at the expense of previously acquired knowledge. This is a significant departure from traditional machine learning paradigms that typically rely on static datasets and often require retraining from scratch when new data becomes available. In continual learning, the model is trained on a sequence of data streams, ideally retaining the knowledge from earlier data while integrating new information.

1.4.3 Differential privacy

Differential privacy is a mathematical framework designed to quantify and control the privacy loss incurred when releasing information about a dataset. The concept of (ϵ, δ)-differential privacy provides a more nuanced approach than the standard ϵ-differential privacy, allowing for a small probability of additional privacy loss. The formal definition is as follows:

A randomized algorithm $\mathcal{A}$ is (ϵ, δ)-differentially private if for any two adjacent datasets D and D′, and for all sets S of possible outputs, the following holds:

$$\Pr[\mathcal{A}(D) \in S] \leq e^{\epsilon} \Pr[\mathcal{A}(D') \in S] + \delta \tag{1}$$

In this definition, D and D' are datasets that differ by at most one element, representing the notion of "adjacency" in the dataset space. The parameter ϵ (epsilon) represents the privacy loss, with smaller values indicating more privacy. The term e^{ϵ} is the base of the natural logarithm raised to the power of ϵ, dictating the multiplicative bound on the increase in likelihood of any outcome. The parameter δ, on the other hand, represents a small probability, allowing the privacy guarantee to be violated by a small amount. This relaxation with the δ term provides flexibility, particularly in scenarios where strict ϵ-differential privacy is too limiting or impractical.

The significance of (ϵ, δ)-differential privacy lies in its balance between data utility and privacy. While smaller values of ϵ and δ provide stronger privacy guarantees, they can also limit the usefulness of the output data. As such, selecting appropriate values of ϵ and δ is crucial and often involves trade-offs. This framework has become fundamental in various fields where the privacy of individuals in a dataset is paramount, such as in statistical analysis, machine learning, and data mining.

1.4.4 Generative AI

The rise of generative AI in OSNs is a double-edged sword, offering the ability to produce content that appears authentic while simultaneously posing significant risks for fabricating false information. This technology, which can create convincingly human-like posts, comments, and profiles, has potential applications in spreading disinformation, influencing public opinion, and executing privacy breaches. The proliferation of deepfakes, a form of generative AI that produces realistic fake videos and images, further exacerbates these risks. Deepfakes can be used for malicious purposes like revenge porn, blackmail, or political manipulation. Moreover, generative AI's role in creating personalized advertisements raises concerns about covert data collection and privacy infringement. The challenge lies in distinguishing fake content from genuine material as the technology evolves and in regulating its use, particularly in the context of rising fake news and misinformation. Solutions include developing algorithms to detect AI-generated fake content and creating privacy-preserving tools for data analysis to protect user privacy in personalized advertising.

Generative Adversarial Networks (GANs) have significantly impacted privacy in the digital realm. GANs consist of two neural networks, the generator and discriminator, working in tandem to create highly realistic synthetic data, often indistinguishable from real data. This capability has profound implications for privacy. On the one hand, GANs can generate synthetic datasets that mimic real user data, enabling researchers and companies to use data without compromising individual privacy. On the other hand, GANs pose significant risks in creating realistic deepfakes, which can be used to invade privacy, spread misinformation, and manipulate public opinion. The authenticity of digital content is increasingly questioned, necessitating robust detection methods and ethical guidelines for GAN use. The balance between leveraging GANs for beneficial purposes and safeguarding against their potential misuse remains a critical challenge in the contemporary digital landscape.

1.4.5 Measuring privacy and fairness

Privacy and fairness are essential elements of responsible AI. Privacy aims to protect individual data contributions from being identified in computational outputs, while fairness ensures equitable treatment across diverse groups. However, the interplay between privacy and fairness, particularly in light of recent advancements, is not fully understood. DP has been a significant approach in maintaining privacy in deep ML models, with recent methods striking a commendable balance between privacy and utility. For example, De *et al*. employed differentially private stochastic gradient descent (DP-SGD) with a 16-layer Wide-ResNet on CIFAR-10, achieving notable accuracy while preserving privacy. Additionally, Park *et al*. introduced a more generalized algorithm, Differentially Private Sharpness-Aware Training (DP-SAT), which outperforms DP-SGD in accuracy while providing comparable privacy protections.

Despite these advancements, challenges remain, such as vulnerability to Membership Inference Attacks (MIAs), which can reveal training samples. This risk is particularly prevalent when employing a more relaxed privacy constraint, leading to the ”Onion Effect” where removing the most vulnerable data layer exposes another previously considered safe. Moreover, the pursuit of privacy, especially through DP, can inadvertently compromise fairness. Studies have shown that DP may result in greater accuracy drops for unprivileged groups, leading to increased unfair outcomes. The impact of generalization techniques, used to enhance privacy-utility trade-offs, on fairness is also unclear, potentially introducing or exacerbating bias. Research by Wang *et al*. highlights this by using databases with inherent biases to investigate how ML algorithms modify bias in model outcomes, emphasizing the need for a nuanced understanding of these trade-offs in the development of ethical ML technologies.

1.5 Related work and identified gap

1.5.1 Measuring privacy leakage

The prevailing methodologies in privacy score evaluation for online social networks often view the privacy score as an aggregate of individual privacy scores corresponding to each element of a user's profile, such as email, relationship status, and mobile phone number. The influence of each profile item on the total privacy score is determined by two key factors: the inherent sensitivity of the information and the level of visibility that arises from the user's chosen privacy settings. This approach underlines a critical understanding of privacy management in digital spaces.

Tracing back to the origins of this approach, Maximilian et al.'s 2009 work stands as a pioneering effort. They developed a framework that quantifies privacy scores through an interplay between the sensitivity and visibility of personal information. This conceptual model has been a cornerstone for various subsequent researches that have adopted similar constituents sensitivity and visibility for calculating Privacy Leakage Scores (PLS).

However, the definition and application of these components require more in-depth analysis. When dissecting statistical-based methods, it becomes evident that traditional privacy metrics are employed to yield quantitative data on various aspects affecting user privacy. This includes, but is not limited to, attribute information, network environment, trust dynamics among users, and the nature of the published content. Despite this comprehensive coverage, two major issues emerge with statistical-based approaches. Firstly, they tend to be inefficient. The process typically involves isolating features, individually measuring them, and then amalgamating these measurements into a singular privacy score. Moreover, the subjectivity inherent in privacy as a concept raises significant doubts about the validity of any derived numerical value. Secondly, these methods overly rely on the artificial extraction of features. In the realm of privacy metrics research, the selection and importance weighting of features for accurate privacy leakage measurement remains a vexing issue. The identification of inter-feature relationships and their collective impact on privacy is yet another area that necessitates urgent attention.

In addition to these concerns, when the network environment of users is taken into account, the analysis often becomes exceedingly complex and cumbersome, especially considering the potentially millions of connections a single user might have. Previous strategies that sought to derive a user's privacy score by examining their entire network frequently result in inefficiencies and inaccuracies.

Turning towards ML-based methodologies, these have recently gained traction in addressing privacy leakage in unstructured data such as text and images. For instance, certain studies alert users to potential exposure of their biometric data. More importantly, ML-based approaches can uncover hidden patterns in data that might reveal private information. This is a significant leap from traditional methods. Despite these advancements, there is a palpable research gap in effectively integrating ML approaches with the traditional privacy metrics to enhance the accuracy and efficiency of privacy leakage assessments. Such integration could potentially address the existing inefficiencies and provide a more nuanced understanding of privacy in online social networks.

1.5.2 Disclosing private information via generative AI

In the burgeoning field of generative AI, one emerging research gap is understanding and mitigating the risks associated with the unintentional disclosure of private information via these technologies on OSNs. While generative AI offers innovative ways to create content, its potential to inadvertently reveal sensitive information has not been thoroughly investigated. The concern is that generative AI can synthesize realistic images or other forms of content that could unintentionally include or imply private data. For example, AI might recreate facial features from partial images, leading to identification risks even when users are cautious about sharing their photos. In OSNs, where users frequently post images, the risk of such unintentional exposure is heightened. This gap is particularly evident in scenarios like the "eyes-to-face" problem, where AI-generated images of a person's eyes might be used to reconstruct their entire face, thus compromising their anonymity and privacy.

1.5.3 Privacy settings automation through classification

The rapidly expanding area of automating privacy settings on OSNs offers substantial research opportunities, especially in the context of advanced machine learning techniques. A pertinent example in this context is the automation of specific privacy settings on platforms like Instagram, such as the "allow profile picture expansion" option. This option in Instagram's privacy settings controls whether or not other users can tap on your profile picture to see it enlarged. If the user enable this setting, other users will be able to tap on your profile picture to view a larger version. If the user disable it, the profile picture will only be viewable in the small format displayed on the user profile, helping to maintain more privacy. It is important to note that allowing users to expand a profile picture grants them access to a higher resolution image, thereby increasing their access to potentially sensitive data.

One approach to automating this process involves the classification of each user's images. Consider a hypothetical scenario wherein a user prefers to permit the expansion of images featuring their dogs, but not those depicting their own face. In this instance, a classifier could be employed to distinguish between these categories of images. Consequently, the user could configure the system to deactivate the expansion option when uploading a personal portrait, and activate it when posting images of their dogs. This method facilitates tailored privacy controls based on the content of the image.

A personalized automated system for adjusting privacy settings could be developed. Such a system would use CL to adapt to prevent storing data when a new class of images has been added by user and therefore, less resources will be used for training the classifier. Furthermore, incorporating DP into the system enhances privacy protection, ensuring that these adjustments do not compromise the user's personal data. Therefore, a system based on DP-CL could be implemented to classify each user's profile image. The final decision regarding the activation or deactivation of the "allow profile profile picture expansion" option can be determined based on the user's initial action for that specific category of images.

Incorporating findings on the enhancement of privacy in machine learning algorithms, particularly in the realm of CL and DP, offers a promising approach to this challenge. The research underscores the importance of balancing privacy protection and utility in machine learning models, especially when faced with the need for continuous learning from streaming data. The proposed methodology in the paper aims to effectively integrate DP into CL, striking a balance between maintaining data privacy and ensuring the model's utility.

1.5.4 Privacy and fairness interplay

The interplay between privacy and fairness in machine learning models for Online Social Networks (OSNs) is crucial due to the sensitive nature of user data and the potential for algorithmic bias. Ensuring privacy involves protecting user data from unauthorized access, but overly strict privacy measures can lead to inadequate data representation, potentially causing bias in model outputs. Conversely, to achieve fairness, machine learning models must be audited for biases, which requires a degree of transparency that could conflict with privacy preservation.

This balance is further complicated by the need for personalization in OSNs and compliance with regulatory frameworks like GDPR, which mandate both privacy protection and non-discrimination. Achieving a harmonious balance is key; models must be designed to personalize user experiences without infringing on privacy or causing unfair outcomes. This is not just a technical challenge but also a legal and ethical one. Maintaining user trust and engagement hinges on the platform's ability to navigate this interplay effectively.

Addressing the privacy-fairness balance is therefore essential in creating ethical, legally compliant, and user-friendly machine learning models for OSNs. This approach ensures responsible data usage and algorithmic fairness, fostering trust among users and aligning with regulatory standards. It is a delicate task, but one that is crucial for the sustainable and equitable operation of social networks in the digital age.

1.6 Summary of contribution

This thesis encompasses the publication of six papers that address the research questions posed. Initially, we introduced an adaptive privacy measuring framework named PriMe. This framework is designed to calculate a privacy leakage score for each user action within an OSN, and then adjust the privacy settings according to the user's preferred privacy scopes and boundaries. To ensure the accuracy of the privacy leakage scores, the framework considers various types of data, actions, and personal characteristics of each user.

Second paper investigate the impact of linkability between user profiles and shared content across various OSNs, a factor that has considerable implications for privacy leakage. We introduce a novel method for quantifying the linkability between profiles across multiple networks, based on key features and metrics that capture profile similarities. We applied this methodology to a dataset of user profiles across three online social networks named Flickr, Facebook, and Twitter. Our approach includes examining both structured and unstructured data related to user profiles, enabling us to offer a valuable understanding of linkability trends and identify potential privacy risks.

To assess the impact of GAI on revealing concealed private information, we published two papers concerning the eyes-to-face problem. In the first paper, we introduced a novel GAN-based deep learning model called Eyes-to-Face GAN (E2F-GAN). This model features two primary modules: a coarse module and a refinement module. The coarse module, supported by an edge predictor, extracts essential features from the periocular region and produces a preliminary output, which is then enhanced by the refinement module. In the second paper, we developed another GAN-based model named Eyes-to-Face Network (E2F-Net). This approach employs two specialized encoders to separate identity and non-identity features from the periocular region. These features are mapped to the latent space of a pretrained StyleGAN generator, leveraging its advanced capabilities and its rich, diverse, and expressive latent space without needing additional training. We also enhanced the StyleGAN output by implementing a new optimization technique for GAN inversion, aimed at locating the optimal code within the latent space.

To continuously adjust the privacy settings of a user profile, we proposed a methodology by which we cannot only strike a tradeoff between privacy and utility, but also mitigate the CF. The proposed solution presents a set of key features: (1) it guarantees theoretical privacy bounds via enforcing the DP principle; (2) we further incorporate a robust procedure into the proposed DP-CL scheme to hinder the CF; and (3) most importantly, it achieves practical continuous training for a CL process without running out of the available privacy budget.

In the sixth paper, we explore how different generalization techniques affect private and non-private learning in both biased and unbiased data scenarios. We assess the privacy risks associated with these scenarios through membership inference attacks (MIAs) and analyze the effects of removing samples that pose a high privacy risk, referred to as outliers. Additionally, we introduce a new metric called ABE, which simultaneously measures accuracy, privacy, and bias.

1.7 Conclusion

In conclusion, this thesis has significantly advanced the understanding and management of privacy within online social networks through its exploration of key research questions. We have demonstrated innovative methods and frameworks that measure and adjust privacy settings dynamically, addressing the urgent need for privacy management in an increasingly interconnected digital world. Notably, the PriMe framework sets a precedent in privacy management by calculating privacy leakage scores and adjusting settings in real-time, based on user preferences and actions. Furthermore, our investigations into the linkability of user data across platforms have highlighted potential privacy risks and provided strategies to mitigate these through better privacy protection mechanisms. These contributions underscore the importance of sophisticated privacy measures and provide a foundation for further research in protecting user privacy in online environments.

Additionally, our research delves into the possibilities of revealing hidden private information through advanced generative adversarial networks and explores the dynamics of continuously updating privacy settings without depleting privacy budgets. The Eyes-to-Face GAN and Network models represent significant strides in using deep learning to analyze and protect personal data. Through the deployment of these models, we have shown that it is possible to enhance the security and fairness of automated systems in managing user data, thereby fostering trust and ensuring equitable treatment in digital interactions. Our work not only contributes to the academic discourse on privacy and data security but also offers practical insights and tools that can be adopted by developers and policymakers to create safer and more respectful online spaces.

References

[1] A. Hassanpour and B. Yang, “PriMe: A novel privacy measuring framework for online social networks,” in 2022 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 2022.

[2] A. Hassanpour, M. M. Utsash, and B. Yang, “The impact of linkability on privacy leakage,” in Proceedings of the International Conference on Advances in Social Networks Analysis and Mining, 2023.

[3] A. Hassanpour et al., “E2F-GAN: Eyes-to-face inpainting via edge-aware coarse-to-fine GANs,” IEEE Access, vol. 10, pp. 32406–32417, 2022.

[4] A. Hassanpour, F. Jamalbafrani, B. Yang, K. Raja, R. Veldhuis, and J. Fierrez, "E2F-Net: Eyes-to-face inpainting via StyleGAN latent space. Pattern Recognition," 152, p.110442, 2024.

[5] A. Hassanpour, M. Moradikia, B. Yang, A. Abdelhadi, C. Busch, and J. Fierrez, "Differential privacy preservation in robust continual learning," IEEE Access, vol. 10, pp. 24273–24287, 2022.

[6] A. Hassanpour, A. Zarei, K. Mallat, A. Oliveira, and B. Yang, The Impact of Generalization Techniques on the Interplay Among Privacy, Utility, and Fairness in Image Classification, submitted to Privacy Enhancing Technologies Symposium 2025.

Chapter 2

PriMe: A Novel Privacy Measuring Framework for Online Social Networks

Ahmad Hassanpour, Bian Yang

Abstract

Online Social Networks are responsible for disclosing a large amount of sensitive information. Users unintentionally reveal their sensitive information and are unaware of the privacy risks involved. But the users should be well informed about their privacy quotient and should know where they stand on the privacy measuring scale. In this paper, we proposed an adaptive privacy measuring framework called PriMe that can measure the privacy leakage score for each action of a user in an OSN and subsequently adjust the privacy settings based on the preferred privacy scopes and boundaries. Various types of data, actions, and personal characteristics of each user have been considered to ensure the calculated privacy leakage score is accurate.

Keywords- Online social network, privacy leakage, measuring privacy.

2.1 Introduction

The ubiquity of information communication technologies which is leading to present-day digital society has changed the basic principles of human interaction. Although privacy, as one of these principles, has been noted for several decades ago [1], it is attracted a lot of attention in recent years. Online social networks (OSNs) (e.g., Facebook, LinkedIn, MySpace), as a particular type of virtual community, attempt to provide helpful functionalities including maintaining/increasing social relationships [2], finding users with similar interests, improving our knowledge [3], and financial benefits, the published data in such environments can violate various aspects of users' privacy [4]. In fact, users are virtually interacting continuously, and disclose various levels of private information about themselves or others unconsciously [5]. Therefore, OSNs are one of the main bridges of revealing personal information by allowing users to upload their footprints (e.g., text, images, and videos) and interact with others in a variety of ways. Moreover, by raising the number of users of an OSN which lead to more dissemination of information, as well as sharing different varieties of information within many OSNs, users' privacy concerns may increase. Additionally, the recent approval of the GDPR (General Data Protection Regulation) compels OSN service providers to provide more data protection settings and offer further control to OSN users over their personal data. In the following, some challenging problems which users and OSN providers are facing due to preserving the privacy of users have been discussed.

First, privacy is a multi-dimensional concept especially when it is under investigation in the OSNs context. Inspiring by Burgoon et al. [6], Zhang et al. [7] proposed a fourdimensional privacy concept including virtual territory privacy, factual privacy, interactional privacy, and psychological privacy. C1) virtual territory privacy: differing from physical privacy which is defined as the freedom from surveillance and unwanted intrusions upon one's space by the physical presence, touch, sights, sounds, or odors of others, in the virtual social context, there are no physical boundaries that help define the private territory. However, people still feel ownership of the digital belongings that they are entitled to or that are created by them (for example, web-logs, personal spaces, profile pages, etc.). C2) factual privacy: refers to the ability to control identifiable personal information about oneself. C3) interactional privacy: individuals may feel compelled by or uncomfortable under some circumstances relating to social interaction. For example, conversation requests may be initiated obtrusively or at inappropriate times. C4) psychological privacy: people need the freedom to express their own views and the capability to hide themselves from norms that they do not agree with. Psychological privacy protects the individual from intrusions upon one's thoughts, feelings, and values.

Second, each user usually has a scope in his/her mind before publishing any data on OSNs, and privacy requires keeping the published information in its predesignate scope. The work [8] defined the scope as S1) breadth (the distribution of audience), S2) depth (the degree of allowed usage), and S3) lifetime (the long life of the published data). When a piece of information is moved beyond its predesignate scope in any of these dimensions (accidentally or maliciously), a privacy breach occurs. Therefore, a breach may occur when information is shared with a party for whom it was not intended (disclosure), when information is abused for a different purpose than was intended, or when information is accessed after its intended lifetime. Aside from the scope, users in OSNs are contending with privacy three boundaries [9] including B1) disclosure (users try to handle the anxiety of disclosing their information in a public or private manner) B2) identity (the identity boundary is described as the ability to manage one's information with particular groups. For example, it shows users' behaviors in different situations: one at work and the other at a party) B3) temporal (it shows

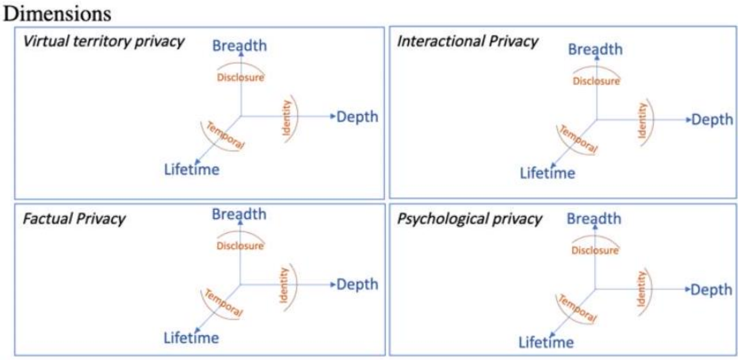


Figure 2.1: The relation between privacy dimensions, scopes, and boundaries has been presented. Each dimension has its own scopes (i.e., breadth, depth, and lifetime), and each scope has its own boundaries (i.e., disclosure, identity, temporal).

how the conduct of individuals may differ over time). Privacy has various dimensions, and users try to consider different scopes and have their own boundaries. Our notion of the relation between the dimensions, scopes, and boundaries has been presented in Figure 2.1 Each dimension has its corresponding scopes, and each user has a different boundary for each scope.

Finally, to perfectly utilize the provided functionalities of an OSN, users need to publish more information, thus, there is a tradeoff between optimal use of functionalities and user privacy. Moreover, another issue called the privacy paradox has been observed in users' online behavior [10][11]. Recent research has revealed discrepancies between user attitude and their actual behavior. More specifically, while users claim to be very concerned about their privacy, they nevertheless undertake very little to protect their personal data.

Considering all dimensions of privacy and users' scopes to preserve the privacy of users in OSNs is extremely challenging. As a solution, OSNs provide policies and privacy settings to control and adjust who can access users' profiles and posts [12], [13]. However, the privacy policies offered by the system are confusing and expressed in legal jargon that is difficult to understand. Furthermore, privacy settings are complex, time-consuming, and still insufficient to fully protect users' privacy [14]. Besides, OSN providers mostly store, process, analyze users' data, and sometimes sell them to third-parties for advertising and marketing purposes. Moreover, to prevent privacy breaches in OSNs, privacy has been investigated from various perspectives (i.e., social, legal, and technical) among researchers.

One of the effective solutions for preserving the privacy of OSNs' users is measuring the privacy leakage for each piece of published data. By doing this, users will be noticed the portion of privacy that might be violated. Considering the intended scope of users and more technically, extracting risks from published data such as comments and posts to calculate the privacy score is a demanding task. Interestingly, publishing some information that is risk-free for many users can be detrimental to others such as a criticism against religion or government since in some countries such criticisms are acceptable, while in other countries, will lead to difficulties. Different authors have proposed various techniques and methods from the algorithmic approach to statistical ones to score and measure privacy. The main two approaches for measuring privacy are statistical or machine learning (ML)-based. For each approach, several models have been proposed to measure the privacy of users in OSNs. The most well-known methods related to these approaches have been discussed in the related work section.

In this paper, we proposed PriMe which is an adaptive privacy measuring framework that can measure the privacy leakage score (PLS) for each action of a user in an OSN and adjust the privacy setting of each user based on the preferred privacy scopes and boundaries. Various types of data, actions, and personal characteristics of each user have been considered to ensure the calculated PLS is accurate. Moreover, we discussed why the previous methods for calculating PLS are not precise and proposed a new method.

2.2 Related works

From the technical point of view, the statistical-based methods mostly rely on two intuitive properties (i) the sensitivity of the information being revealed and (ii) the visibility of the revealed information within the network. The proposed methods are working on Dichotomous or Polytomous variables or a combination of them. On the other hand, ML-based models mostly try to measure the privacy of unstructured data (text, photo, etc.). In the following, we briefly review the proposed methods for both approaches.

2.2.1 Statistical-based approaches

Notably, A dichotomous variable takes only one of two possible values when observed or measured. The value is most often a representation of a measured variable (e.g., age: under $65/65$ and over) or an attribute (e.g., gender: male/female). A variable having more than two possible categories, either ordered or unordered called polytomous variable. For example, college matriculation could be described as a polychotomous variable: freshman, sophomore, junior, or senior. Table 2.1 summarizes several statistical-based methods for measuring privacy score the type of data (dichotomous, polytomous variables, or their combination), the proposed formulation, and a short description has been extracted for each paper. Most proposed methods, consider privacy score as a combination of the partial privacy scores of each one of his profile items (e.g., email, relationship status, mobile phone number). The contribution of each profile item in the total score depends on the sensitivity of the item and the visibility it gets due to the user's privacy settings.

one of the first attempts to design a privacy metric for online social networks was proposed by Maximilian et al. [15] in 2009. The authors have proposed a framework to calculate the privacy score based on the sensitivity β_i and the visibility $v(i,j)$ of profile items $i \in \{1,\ldots,n\}$ of user j in a social network.

$$PR(i) = \sum_i PR(i,j) = \sum_i \beta_i \times v(i,j) \tag{1}$$

Several other papers, listed in Table 2.1, proposed other methods for measuring PLSs based on the same components (i.e., sensitivity and visibility). In the following, we will review the definition of these components and how they have been used.

Sensitivity: Specifying the sensitivity of data is a challenging task since sensitive data can be a number of things. One of the easiest ways to evaluate is to think of personal data you would not want to be openly shared with just anyone. There are, of course, federal laws and regulations that set specific guidelines on what types of sensitive data must be protected, like financial information (e.g., Credit card numbers, bank account information, and social security numbers), government information (e.g., any document that is classified as secret or top-secret, restricted, or can be considered a breach of confidentiality), business information (e.g., accounting data, trade secrets, financial statements or accounts, and any sensitive information in business plans), personal information (e.g., addresses, medical history, driver's license numbers, or phone numbers). However, GDPR makes a clear distinction between sensitive and non-sensitive personal data. Article 9 of GDPR establishes special categories that require extra attention. Sensitive data, or special category data, according to GDPR is any data that reveals a subject's information including racial or ethnic origin, political beliefs, religious beliefs, genetic or biometric data, mental health or sexual health, sexual orientation, and trade union membership. Besides having various types of sensitive data, the level of sensitivity of each data type can be different for each user. For example, politicians publish their political opinions on OSNs without having any concerns.

Shortly, sensitive data is information most people would not want to share with others who don't have approval, and sensitivity shows the risk associated with the attributes of the user. when the sensitivity of an attribute increases, the risk posed by information disclosure of the individuals also

Table 2.1: Summarize of various privacy scoring solutions based on statistical-based approaches

Author and year	Approach/Data Type	Data Source	Proposed formulation	Description
Renner (2010) [16]	Dichotomous	Facebook	Risk = Negative consequence × Likelihood	defining privacy risk by considering two privacy metrics including negative consequence information leakage and the likelihood of information leakage.
Maximilien et al. (2009) [15]	Dichotomous	-	$PR(k,l) = \beta_k \times v(k,l)$ $\beta_k = \frac{(M-\lvert R_k\rvert)}{M}$	$PR(k,l)$ shows privacy score. β_k shows the sensitivity of k-th attribute, $v(k,l)$ shows the visibility of attribute k of user l, $\lvert R_k\rvert$ is the number of individuals that make their attributes publicly available, M is number of users.
Maximilien et al. (2009) [17]	Dichotomous	-	$PR(k,l) = \beta_k \times v(k,l)$ $\beta_k = \frac{(M-\lvert R_k\rvert)}{M}$	$PR(k,l)$ shows privacy score. β_k shows the sensitivity of k-th attribute, $v(k,l)$ shows the visibility of attribute k of user l, $\lvert R_k\rvert$ is the number of individuals that make their attributes publicly available, M is number of users.
Srivastava and Geethakumari (2013)[18]	Dichotomous/Unstructured	private dataset	$PQ(j) = \sum_k \beta_k \times v(k,l)$ $v(k,l) = \frac{\lvert R_k\rvert}{M} \times \frac{\lvert R_l\rvert}{M}$ $\beta_k = \frac{(M-\lvert R_k\rvert)}{M}$	$PQ(j)$ is final privacy score. β_k shows the sensitivity of k-th attribute. $v(k,l)$ shows the visibility of attribute k of user l, $\lvert R_k\rvert$ is the number of individuals that make their attributes publicly available, M is number of users.
Domingo-Ferrer (2010)[19]	Dichotomous/Structured	Simulation-based experiments	$PRF = \frac{\sum_{j'=1,j'\neq j}^{N}\sum_{i=1}^{n}\sum_{k=1}^{l}\beta_{ik}V\left(i,j',k\right)I\left(j,j',k\right)}{1+\sum_{i=1}^{n}\sum_{k=1}^{l}\beta_{ik}V(i,j',k)}$	Where j and j' are the two users in the social networks, k indicates the number of links between users and n indicates the number of attributes for a user. $\mathrm{I}\,(\mathrm{j},\mathrm{j}',\mathrm{k}) = 11$ If j' and j are k links away from each other, otherwise 0.
Nepali and Wang (2013) [20][21]	-	-	$PIDX = \frac{\sum_{k=1}^{m} p'_k s'_k}{\sum_{k=1}^{n} s_k} \times 100$	p'_k shows the visibility of each attribute, and s'_k shows the corresponding weight. n indicates the number of attributes, and m shows a subset of them which belongs to k-th user.
Talukder et al. (2010)[22]	Dichotomous/Structured	-	$S_\gamma^i = \sum_{k=0}^{q} \omega^{(k)} \psi_i^{(k)} \tilde{\psi}^{(k)}$	$\omega^{(k)}$ is the relative sensitivity vector for attributes, Privometer records the success and failure of the inferred attributes as a vector, called attribute matching vector, $\tilde{\psi}$. We also represent $\psi_{\mathrm{i}}^{(\mathrm{k})}$ as matching vector that records the matches between two attributes.
Petkos et al. 2015[23]	Dichotomous	-	$PQ(j) = \sum_k \beta_k \times v(k,l)$	$PQ(j)$ is final privacy score. β_k shows the sensitivity of k-th attribute. $v(k,l)$ shows the visibility of attribute k of user l.
Liu and Terzi (2010)[24]	Polytomous/Structured	Synthetic and private dataset	$P_{ij} = \frac{1}{1+e^{-\alpha_i(\theta_j-\beta_i)}}$	β_i shows the sensitivity of attribute i. α_{i} quantifies the discrimination power.3
Becker and Chen (2009) [25]	Polytomous	Facebook	Privacy measured based on inference detection	Try to infer attributes of each user.
Aghasian et al. (2017) [26]	Polytomous/Structured	Facebook, ResearchGate, LinkedIn, and Google +	$\text{Privacy} = \frac{\sum_{i=1}^{m}\beta_i \times F_{vis}(xi)}{m}$	β_i shows the sensitivity of attribute $i.F_{vis}(xi)$ indicates the visibility score for each attribute calculated by fuzzy rules.
Pensa and Di Blasi (2017) [27]	Polytomous/Structured	Facebook	Privacy measured based on sensitivity and visibility	measure the privacy risk of the users and help the users customize semi-automatically their privacy settings

Table 2.2: Summarize of various privacy scoring solutions based on machine learning-based approaches

Author and year	Approach/Data Type	Data Source	Machine learning Algorithm	Description
Li et al., (2020) [28]	Structured	Collect data from Sina Weibo	Deep neural network	Calculating privacy score by extracting profile information and graph structure information of users' friends.
Aghasian et al., (2020)[29]	Structured and unstructured (text)	Collect data from Facebook and Twitter	Fuzzy-based model	measure and warn users regarding the textual data privacy risks they have shared in online social platforms.
Aghasian et al., (2017)[30]	Structured	Collect data from Facebook, ResearchGate, LinkedIn, and Google+.	Statistical and fuzzy systems	specify the potential information loss for a user by using obtained privacy disclosure score
Yu et al., (2018) [31]	Unstructured (image)	public image sets, PicAlert and Mirflickr	Deep neural network	recommending fine-grained privacy settings for social image sharing by considering content sensitiveness of the images and trustworthiness of the users
Orekondy et al., (2017)[32]	Unstructured (image)	Visual Privacy (VISPR) dataset	Deep neural network	predict user specific privacy score from images in order to enforce the users' privacy preferences
Battaglia et al., (2020)[34]	Unstructured (text)	Collect data from social media	k-NN, decision tree (DT), Multi-layer Perceptron (MLP), SVM, Random Forest (RF), and Gradient Boosted trees (GBT)	Assign a score to any text sample according to its degree of sensitivity
Tseng et al., (2024)[39]	Unstructured (image)	Collect data from social media	Deep Learning	Localize sensitive objects
Zhao et al., (2023)[40]	Unstructured (image)	Collect data from social media	Deep learning	predict privacy for online images
Liu et al., (2023)[41]	Unstructured (image)	Collect data from social media	Decision Tree model	extract sensitive relations from the photos labeled private or public
Xompero et al., (2024)[42]	Unstructured (image)	PrivacyAlert Dataset	Deep Learning	identify and quantify objects relevant to privacy classification

increases. If $|R_k|$ shows the number of individuals that make their attributes publicly available, M is number of users, [15] calculates the sensitivity of k-th item by $\beta_k = \frac{(M-|R_k|)}{M}$.

Visibility: The probability of leakage of private information of a user also depends on the position of the user in network topology. If the user himself is directly or indirectly connected to a large set of nodes in the network, then the chances of information leakage through his neighbors increase. For example, if a user considers his birth date as a piece of private information and shares only with his friends, it is highly likely that any one of his friends may share such information further with his friends, thereby causing an information leakage. The probability of this leakage will primarily depend on the number of users in his vicinity in one or more hops. Therefore, the probability of leakage increases with the visibility of the user himself(i.e., the number of users who would be interested in the information of the user) as well as the visibility of his/her friends.

Assuming independence between items and users, we can compute $P_{i,j}$ to be the product of the probability of a 1 in the i-th row of R (i.e., $\frac{|R_i|}{N}$) and the probability of a 1 in the j-th column of R (i.e., $\frac{|R_j|}{n}$). That is, if $|R_j|$ is the number of items for which j sets $R(i,j) = 1$, we have $v(i,j) = \left(\frac{|R_i|}{N}\right) * \left(\frac{|R_j|}{n}\right)$. this notion does not measure the visibility for a specific item very accurately. Consider two users i and k in an OSN with 10 users, and a specific item j. Assume user i revealed 3 items out 10 items that existed in the OSN, and user k revealed 6 items, both revealed item j. Also assume the probability of revealing item j is 0.7 (i.e., $\frac{|R_j|}{n} = 0.7$). Therefore, the visibility score for user j and item j is $v(i,j) = 0.3 * 0.7 = 0.21$ and for the user k is $v(i,j) = 0.6$*$0.7 = 0.42$. The problem here is that the more a user discloses its information, the more visibility score will be charged for each disclosed item. Moreover, here, the visibility of an item is calculated without considering the users' network. Therefore, if user i and k have the same network (friends), their visibility scores for the same item (j) is not equal because one of them revealed more information.

Discussion on statistical-based approaches: The proposed statistical-based methods are using traditional privacy metrics to obtain quantitative statistics on all the aspects that affect users' privacy disclosure, including but not limited to attribute information, network environment information, trust between users, and publishing information content. However, these approaches face two problems. First, these approaches are inefficient. Most of these approaches first extract features, then measure them separately, and finally integrate them into a numerical value. In addition, the calculation method also faces various doubts because privacy is a virtual concept without a unifying principle, and any calculation is considered to be subjective and unconvincing. Second, this method relies too strongly on artificial fea-

Table 2.3: The proposed privacy leakage metrics for each data type

Data type	Proposed Metrics
user network	Number of friends, Measuring trust for each friend
structured / unstructured data	If the data include sensitive information, types of sensitive information (e.g., biometric, religion, political view), transparency of provided data, uniqueness.
action	if the action include sensitive information (e.g., post a content, like a post), lifetime of action.

ture extraction. In previous research on privacy metrics, feature extraction is a difficulty. Which features can be used for privacy measurement? Which features are more important to measure privacy leakage more accurately? What associations exist between these features? These problems urgently need to be solved. Meanwhile, when considering the network environment of users, there may be tens of millions of links around a user. Previous methods obtained only one user's privacy score after analyzing the whole network, which is undoubtedly inefficient and inaccurate [35].

2.2.2 Machine learning-based approaches

Apart from statistical-based approaches, ML-based models have been recently used to measure privacy leakage in unstructured data (text, photo, etc.). Some works like [38] can be used to warn the users when part of their biometric data is not hidden, however, ML-based methods can be used to infer the hidden patterns which can assist to disclose the privacy of users. To do this, these approaches try to extract informative private features from unstructured data. Table 2.2 presents some methods which used ML methods like deep learning to extract sensitive information or measure privacy leakage.

2.3 Proposed privacy adaptive meter

2.3.1 Framework

Figure 2.2 shows our proposed adapted privacy meter framework called PriMe including five main modules called User Data, Personal Attribute Analyzer, Privacy Leakage Metrics, Privacy Meter, and Adaptive Privacy Awareness. By considering various data types, actions, and privacy preferences, this framework allows to design and implement an adaptive privacy meter such that different dimensions, scopes, and boundaries of privacy will be measured and adapted for each user separately. Moreover, the framework is highly flexible due to our modular design, thus, some proposed modules can be changed depending on the OSN's requirements.

Moreover, we proposed a different way of measuring PLS comprising three main parameters called sensitivity, linkability, and visibility, leading to a more accurate PLS.

2.3.2 Users data

OSNs' Users generate and provide various types of data including their actions (e.g., like, reshare, add/remove/block to their friendship list), unstructured data (e.g., images, texts, videos), structured data (e.g., birth date, marital status, hometown), and user network (e.g., name of current friends, blocked friends). Undoubtedly, each of this information discloses the privacy of the user to a different degree. For instance, sharing a personal video clip that includes our biometric information (e.g., our face image) reveals more sensitive data compared to liking our friends' post that includes his face image. Therefore, to have a comprehensive privacy meter framework, all provided data types by users should be considered during the calculation.

2.3.3 Privacy leakage metrics

For each type of data, some metrics are calculated to assist the privacy meter module in measuring the PLS more accurately. These metrics extract some characteristics from the raw data or analyzed data. Table 2.3 shows some of our proposed metrics for different data types. By employing these metrics, we aim to convert each portion of data into a numerical value that indicates the sensitivity of each data segment. Thus, the proposed metrics serve multiple purposes. First, they help in identifying sensitive patterns for various portion of data that could indicate potential privacy risks. Second, they allow us to compare the sensitivity levels across different types of data, facilitating a more comprehensive privacy analysis and therefore, more accurate PLS. Lastly, these metrics are integral in developing strategies to mitigate privacy risks by highlighting areas that require enhanced protective measures.

2.3.4 Content analyzer

Measuring the sensitivity of published unstructured data is a highly challenging task. For instance, a political view can be revealed by a short text, an image, or posting a protesting clip by the user. To detect the revealed political view in each of these modalities, different types of AI algorithms are required (e.g., usually natural language processing algorithms are being used for analyzing texts, and computer vision algorithms for analyzing images and videos). Thus, the content analyzer should include several AI models which can detect various types of sensitive content in different unstructured data.

2.3.5 Personal attribute analyzer

Personal attribute analyzer has the responsibility of extracting static (e.g., big five traits) and dynamic (e.g., emotions) personal attributes from the shared data. These features assist us to measure privacy leakage more accurately. For example, if a user posts, likes, or shares more compared to other users, the value of the extravert attribute can increase for that user, and consequently, he is leaking his privacy. Moreover, measuring these attributes help us to develop an adaptive privacy measuring framework.

2.3.6 Privacy meter

Privacy meter is the main module of this framework which has the responsibility of calculating PLS based on the received raw data, personal attribute analyzer, content analyzer, and the calculated metrics by privacy leakage metrics module. It should continuously measure the PLS for each taking or withdrawing action. The main four submodules of privacy meter are the sensitivity calculator, visibility calculator, linkage calculator, and privacy leakage score calculator. Therefore, the privacy leakage score will be a function of three inputs $PR = F$ (sensitivity, linkage, visibility). The function F should be implemented in the privacy leakage score calculator, and each of the three parameters has its own block, explained in the following subsections.

Sensitivity Calculator: previous methods calculated the sensitivity based on the behaviors of users of a specific OSN which means the more users reveal a piece of information, the less sensitive score is considered for it. But article 9 of GDPR defined the categories of sensitive information, explained in section II (A). Therefore, if the content analyzer detects the seven categories of sensitive data (racial or ethnic origin, political beliefs, religious beliefs, etc.), a high sensitivity score will be assigned for that piece of information. Other information will be categorized into semi-sensitive and nonsensitive data. The semi-sensitive data refers to those data that some users may have concerns about revealing them like home address, phone number, working organization, or even some actions such as liking a post. Semi-sensitive and nonsensitive can be adaptively categorized for each user separately which will be done in the adaptive privacy awareness module, described in the next section.

The sensitivity calculator receives the required information from the content analyzer, privacy leakage metrics, or even user data modules. There-

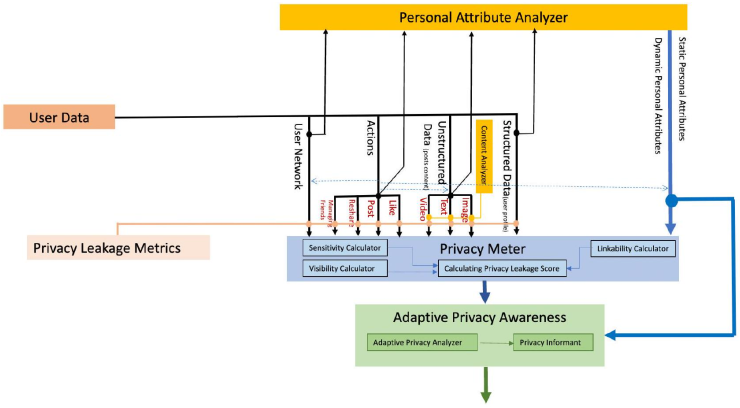

Personal Attribute Analyzer
User Data
User Network
Actions
Unstructured Data (posts content)
Content Analyzer
Structured Data(user profile)
Dynamic Personal Attributes
Static Personal Attributes
Managing Friends
Reshare
Post
Like
Video
Text
Image
Privacy Leakage Metrics
Privacy Meter
Sensitivity Calculator
Visibility Calculator
Calculating Privacy Leakage Score
Linkability Calculator
Adaptive Privacy Awareness
Adaptive Privacy Analyzer
Privacy Informant

Figure 2.2: Overview of proposed privacy measuring (PriMe) framework including five modules called user data, privacy leakage metrics, personal attribute analyzer, privacy meter, and adaptive privacy awareness.

fore, based on the received information, the type of sensitivity (sensitive, semisensitive, non-sensitive) will be measured and converted to a score.

Visibility Calculator: the proposed methods by previous works for calculating the visibility of a specific item are dependent on the visibility of other items shared by the user. Obviously, the more users can see a specific item, the more $PR = F($ sensitivity, linkage, visibility $) =$ sensitivity $*$ linkage * visibility

visibility score should be considered for it. For those users who do not have a small number of friends, the visibility of a published item should not be high even if he/she is published many other personal items/data. Moreover, a trustworthiness score should be considered for each user who existed in the network. Undoubtedly, the visibility score will decrease if the users in the network receive a high trustworthiness score since the revealed data will not share with other users in the network.

The visibility of an OSN should be considered as another factor for measuring the visibility of each published item. Some OSNs are open to search engines, thus, all users on the Internet can search for the content of the published information in a specific OSN. Besides, in some OSNs like LinkedIn some actions (e.g., like, share) lead to users who are not in our connections being able to see our published posts, resulting in more visibility of data.

Linkage Calculator: The linkability between two posts or a post and action can disclose more privacy and thus increase the PLS. For instance, user's political view can be revealed after liking several posts of a specific party. Therefore, the linkability between the provided information and actions should be considered during calculating the PLS (the dashed blue lines in Figure 2.2). Generally, for each portion of data d, the dependency and linkage with other portions that existed in the whole internet should be calculated, $d \perp \mathbf{I}$, where **I** demonstrates the set of all data on the internet and $\perp$ shows the linkability.

Privacy Leakage Score Calculator: After calculating the three main parameters, the PLS can be measured by simply multiplying the three parameters: Since the PLS should be measured continuously, thus, continual ML-based methods that preserved the privacy of users should be utilized [36][37].

2.3.7 Adaptive privacy awareness

Discrepancies between users' attitude and their actual behavior, and having different tastes and priorities for revealing information force a privacy framework to be adaptive. For instance, revealing biometric information is not important for some people while they do not like their political views to be disclosed. Therefore, to have an adaptive privacy informant, some personal characteristics of each user are required.

Adaptive Privacy Analyzer: After calculating the PLS by privacy meter for each action of a user, each reaction of the user will be monitored by this module. This assists to find a relation between the user's personality and his privacy preferences. By doing this, the preferred scopes and boundaries of a user can be fulfilled and measured continuously.

Privacy Informant: this module can inform the user about any privacy leakage after each action or adjust the privacy settings automatically.

2.4 Discussion

The characteristics of the proposed framework (i.e., considering all data types, analyzing the personal attributes of each user, measuring the PLS, and more importantly an adaptive privacy setting) lead to cover all aspects of privacy including dimensions, scopes, and boundaries. Regarding dimensions (i.e., C1-C4), using PriMe users can specify their own virtual territory, the identifiable information and psychological attributes of each user will be detected, and privacy settings will be adjusted such that it complies with the interactional privacy. In regard to the scopes (i.e., S1-S3), users can utilize the provided metrics (by privacy leakage metrics) for their published data (i.e., network, actions, and data), and decide about breadth, depth, and the lifetime of data. The adaptivity of the proposed framework allows for fulfilling all boundaries and thus, each user can choose its own identity, temporal, and disclosure.

Calculating the sensitivity, linkability, and visibility for each piece of data is not a trivial task, mostly because of two reasons. First, when the data is unstructured, extracting sensitive features should be done by some ML algorithms e.g., deep learning models, which need a large amount of training data. Moreover, the selected ML algorithm should be trained on each sensitive category separately, which might be different for each user. Second, calculating linkability between a large number of actions on an OSN such that the detected linkability leads to an increase or decrease of PLS is a difficult task.

2.5 Conclusion

In this paper, we proposed an adaptive privacy framework that can measure a score for each action of a user including posting, liking, adding someone to the network, etc. The proposed framework includes five main modules including User Data, Personal Attribute Analyzer, Privacy Leakage Metrics, Privacy Meter, and Correlation Analyzer. Moreover, we proposed a more accurate method for measuring PLS which comprises three parameters called sensitivity, linkage, and visibility. In future works, we will further elaborate on the PriMe's modules and provide practical solutions for calculating PLS with more details.

Acknowledgment

This work was supported by the Project Privacy Matters (PRIMA) under Grant H2020-MSCA-ITN-2019-860315.

References

[1] R. Gavison, ”Privacy and the Limits of Law,” The Yale law journal, 89(3), pp.421-471, 1980.

[2] X. Zhao, J. Yuan, G. Li, X. Chen, and Z. Li, ”Relationship strength estimation for online social networks with the study on Facebook,”Neurocomputing, 95, pp.89-97, 2012.

[3] M. Pavlovic, N. Vugdelija, and R. Kojic, ”The use of social networks for elearning improvement,” Hellenic Journal of Music, Education and Culture, 6(1), 2015.

[4] N. Kökciyan, and P. Yolum, ”Priguard: A semantic approach to detect privacy violations in online social networks,” IEEE Transactions on Knowledge and Data Engineering, 28(10), pp.2724-2737, 2016.

[5] J. Chen, J. He, L. Cai, and J. Pan, ”Disclose more and risk less: Privacy preserving online social network data sharing,” IEEE Transactions on Dependable and Secure Computing, 17(6), pp.1173-1187, 2018.

[6] J. K. Burgoon, ”Privacy and Communication,” In Communication Yearbook 6, M. Burgoon (ed.), Beverly Hills, CA: Sage, 1982.

[7] N. Zhang, C. Wang, and Y. Xu, ”Privacy in online social networks,” 2011.

[8] M. Beye, A.J. Jeckmans, Z. Erkin, P. Hartel, R.L. Lagendijk, and Q. Tang, ”Privacy in online social networks,” In Computational Social Networks, pp. 87-113,. Springer, 2012.

[9] L. Palen, and P. Dourish, ”Unpacking privacy for a networked world,” In Proceedings of the SIGCHI conference on Human factors in computing systems, 2003, pp. 129-136.

[10] S. Barth, and M.D. De Jong, ”The privacy paradox-Investigating discrepancies between expressed privacy concerns and actual online behavior-A systematic literature review,” Telematics and informatics, 34(7), pp.1038-1058, 2017.

[11] D.J. Solove, ”The myth of the privacy paradox,” Geo. Wash. L. Rev., 89, p.1, 2021.

[12] Y. Liu, K.P. Gummadi, B. Krishnamurthy, and A. Mislove, ”Analyzing facebook privacy settings: user expectations vs. reality,” In Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference, 2011, pp. 61-70.

[13] S.j. De, and A. Imine, ”Choosing the Right Privacy Settings,” In Privacy Risk Analysis of Online Social Networks, pp. 59-67, 2021.

[14] F.D. Stutzman, R. Gross, and A. Acquisti, ”Silent listeners: The evolution of privacy and disclosure on Facebook,” Journal of privacy and confidentiality 4 , no. 2, 2013.

[15] E.M. Maximilien, T. Grandison, T. Sun, T., D. Richardson, S. Guo, and K. Liu, ”Privacy-as-a-service: Models, algorithms, and results on the facebook platform,” In Proceedings of Web (Vol. 2), 2009.

[16] C. Renner, ”Privacy in Online Social Networks,” Thesis, 2010.

[17] E. M. Maximilien, T. Grandison, K. Liu, T. Sun, D. Richardson, and S. Guo, ”Enabling privacy as a fundamental construct for social networks,” In 2009 International Conference on Computational Science and Engineering, vol. 4, pp. 1015-1020. IEEE, 2009.

[18] A. Srivastava, G. Geethakumari, ”Measuring privacy leaks in online social networks,” In advances in Computing, Communications and Informatics (ICACCI), 2013 International Conference on, p. 2095-2100, 2013.

[19] J. Domingo-Ferrer, ”Rational privacy disclosure in social networks,” In International Conference on Modeling Decisions for Artificial Intelligence, 2010. p. 255-265.

[20] R.K. Nepali, and Y. Wang, ”Sonet: A social network model for privacy monitoring and ranking,” In 2013 IEEE 33rd International Conference on Distributed Computing Systems Workshops, 2013, pp. 162-166.

[21] Y. Wang, R.K. Nepali, and J. Nikolai, ”Social network privacy measurement and simulation,” In 2014 International Conference on Computing, Networking and Communications (ICNC), 2014, pp. 802-806. [22]

N. Talukder, M. Ouzzani, A.K. Elmagarmid, H. Elmeleegy, and M. Yakout, ”Privometer: Privacy protection in social networks,” In: Data Engineering Workshops (ICDEW), 2010 IEEE 26th International Conference on, 2010, p. 266-269.

[23] G. Petkos, S. Papadopoulos, and Y. Kompatsiaris, ”PScore: A Framework for Enhancing Privacy Awareness in Online Social Networks,” In: Availability, Reliability and Security (ARES), 10th International Conference on, 2015, p. 592-600.

[24] K. Liu, and E. Terzi, ”A framework for computing the privacy scores of users in online social networks,” ACM Transactions on Knowledge Discovery from Data (TKDD), 5(1), pp.1-30, 2010.

[25] J.L. Becker, ”Measuring privacy risk in online social networks,” University of California, 2009.

[26] E. Aghasian, S. Garg, L Gao, S. Yu, and J. Montgomery, ”Scoring users' privacy disclosure across multiple online social networks,” IEEE access, 5, pp.13118-13130, 2017.

[27] R.G. Pensa, and G. Di Blasi, ”A privacy self-assessment framework for online social networks,” Expert Systems with Applications, 86, pp.18-31, 2017.

[28] X. Li, Y. Xin, C. Zhao, Y. Yang, and Y. Chen, ”Graph convolutional networks for privacy metrics in online social networks,” Applied Sciences, 10(4), p.1327, 2020.

[29] E. Aghasian, S. Garg, and J. Montgomery, ”An automated model to score the privacy of unstructured information-Social media case,” Computers & Security, 92, p.101778, 2020.

[30] E. Aghasian, G. Saurabh and J. Montgomery, ”A privacy-enhanced friending approach for users on multiple online social
networks,” Computers 7, no. 3, 2018.

[31] J. Yu, Z. Kuang, B. Zhang, W. Zhang, D. Lin, and J. Fan, ”Leveraging content sensitiveness and user trustworthiness to recommend fine-grained privacy settings for social image sharing,” IEEE transactions on information forensics and security, 13(5), pp.1317-1332, 2018.

[32] T. Orekondy, B. Schiele, and M. Fritz, ”Towards a visual privacy advisor: Understanding and predicting privacy risks in images,” In Proceedings of the IEEE international conference on computer vision, 2017, pp. 3686-3695.

[33] T. Orekondy, M. Fritz, and B. Schiele, ”Connecting pixels to privacy and utility: Automatic redaction of private information in images,” In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8466-8475.

[34] E. Battaglia, L. Bioglio, and R.G. Pensa, ”Classification-based Content Sensitivity Analysis,” In 28th Symposium on Advanced Database Systems 2020, Vol. 2646, pp. 326-333.

[35] R. Jain, N. Jain, and A. Nayyar, ”Security and privacy in social networks: data and structural anonymity,” In Handbook of Computer Networks and Cyber Security, pp. 265-293, 2020.

[36] S. Farquhar, and Y. Gal, ”Differentially private continual learning,” arXiv preprint arXiv:1902.06497, 2019.

[37] A. Hassanpour, M. Moradikia, B. Yang, A. Abdelhadi, C. Busch, and J. Fierrez, ”Differential Privacy Preservation in Robust Continual Learning,” IEEE Access, 10, pp.24273-24287, 2022.

[38] A. Hassanpour et al., ”E2F-GAN: Eyes-to-face inpainting via edgeaware coarse-to-fine GANs,” IEEE Access, vol. 10, pp. 32406-32417, 2022.

[39] Tseng, Y.Y., Sharma, T., Zhang, L., Stangl, A., Findlater, L., Wang, Y., Tseng, D.G.Y.Y. and Gurari, D., 2024. BIV-Priv-Seg: Locating Private Content in Images Taken by People With Visual Impairments. arXiv preprint arXiv:2407.18243.

[40] Zhao, C. and Caragea, C., 2023. Deep Gated Multi-modal Fusion for Image Privacy Prediction. ACM Transactions on the Web, 17(4), pp.1-24.

[41] Liu, J., Li, L. and Li, N., 2023. Relationship privacy preservation in photo sharing. Online Social Networks and Media, 37, p.100268.

[42] Xompero, A., Bontonou, M., Arbona, J.M., Benetos, E. and Cavallaro, A., 2024. Explaining models relating objects and privacy. arXiv preprint arXiv:2405.01646.

Chapter 3

The Impact of Linkability On Privacy Leakage

Ahmad Hassanpour, Masrur Masqub Utsash, Bian Yang

Abstract

Online Social Networks are responsible for disclosing a large amount of sensitive information. Often, users unknowingly disclose vast amounts of sensitive and potentially (un)related data, oblivious to the associated privacy risks. Our research provides a comprehensive evaluation of the linkability between user profiles and shared content across various OSNs, a factor that has considerable implications for privacy leakage. We introduce a novel method for quantifying the linkability between profiles across multiple networks, based on key features and metrics that capture profile similarities. We applied this methodology to a dataset of user profiles across three online social networks named Flickr, Facebook, and Twitter. Our approach includes examining both structured and unstructured data related to user profiles, enabling us to offer a valuable understanding of linkability trends and identify potential privacy risks. Through our findings, we aim to inform the development of privacyenhancing technologies and contribute to improving the current privacy landscape within OSNs. Our research underscores the critical need for robust privacy measures in the face of the growing interconnectedness of user data across different social networks.

Index Terms-Linkability, Online Social Networks, Privacy Leakage.

3.1 Introduction

The rise of the World Wide Web has significantly changed the fundamentals of human interaction because of the increasing use of information communication technologies in the modern digital society. Online social networks (OSNs) (e.g., Facebook, Twitter, LinkedIn, Reddit) provide an environment through which individuals may interact, share knowledge, express their emotions, and establish and preserve relationships with other online users [28]. This advancement of technology is accompanied by huge privacy concerns as most of the users tend to publish a lot of valuable information in the form of both structured (e.g., name, phone number, address, workplace, school) and unstructured data (e.g., text, image, video) [33] without even knowing them consciously [6]. Therefore, OSNs serve as a crucial platform for exposing personal information by enabling users to share their activities and engage with others through different means which can lead to violation of users' privacy in various aspects [16].

Protecting users' privacy in OSNs is a multifaceted challenge that requires consideration of all dimensions of privacy, including personal, contextual, and societal factors [34]. Although OSNs offer policies and privacy settings to regulate user profiles and posts [7], but the used language is complex and difficult to understand, making users vulnerable to privacy breaches [29]. Moreover, OSN providers collect, process, and analyze user data, and may also sell this data to third parties for advertising and marketing purposes [24]. Consequently, researchers have investigated privacy from various perspectives, including social, legal, and technical, in order to prevent privacy breaches and improve privacy protections in OSNs.

Previous experimental findings have revealed conflicts between privacy controls and the functionalities offered by OSNs which allows a range of privacy exploits such as indicating a misalignment between users' desired level of privacy control and the actual outcomes achieved [19]. Moreover, users now a days tend to use multiple OSNs for separate purposes as the primary capabilities differ from one another and users disclose different types of private information within those platforms. As a result, being able to link one user's multiple OSN profiles can lead to increase privacy leakage because of the access to more diverse private information [5] [1]. Crosslinking multiple OSN platforms thus can facilitate profile and data correlation, leading to inadvertent information sharing and privacy breaches. In the context of measuring privacy leakage, several previous experiments [21] [26] [11] [22] suggested that the privacy quotient is calculated based on sensitivity (the level of confidentiality and potential harm if disclosed [23]) and visibility (the extent to which data can be accessed, viewed, or shared by other entities [14]). Inspired from there, Hassanpour et al. [15] proposed an adaptive privacy leakage calculating framework where an additional and important metric called linkage has been introduced which can be used alongside sensitivity and visibility in order to obtain a more accurate privacy leakage score calculation.

Our work is shown the impact of linkability on privacy leakage. We evaluate the degree of connection between identities and published content on distinct OSNs. In the context of OSN, the linkability score is particularly important for privacy leakage score calculation due to the vast amount of personal information that is shared and interconnected across different platforms. For example, if a user's Facebook and Twitter accounts are linked, it may be possible to infer additional information about the user based on their activity across both platforms which might increase the privacy leakage score. On the other hand, the disclosure of user interests on one OSN may be accompanied by the publication of conflicting or unrelated information on the same or other OSNs, resulting in a potential fluctuation or decrease in the privacy score due to the contradicting nature of the newly shared information.

The objective of this paper is to evaluate the degree of linkability between user profiles across multiple OSNs which can be referred as user profiling and also evaluate the linkability of users' published information among different OSNs. User profiling is the process of constructing a thorough description of a specific user based on that person's actions, preferences, interests, and other crucial characteristics, thereby gathering insights into their behaviors and preferences [12] [17]. To address this challenge, we propose a method for measuring linkability between profiles across multiple networks, based on a set of features and metrics that capture the similarity between the profiles. Additionally, we conducted an analysis to establish connections between users' posts across multiple online social networks (OSNs) with the aim of identifying the linkability of users' shared information. Our method provides a quantitative assessment of linkability, enabling the identification of potential privacy risks and informing the development of privacy-enhancing technologies. We apply our method to a dataset of user profiles across multiple social networks (i.e., Flickr, Facebook, Twitter) and present our findings, demonstrating the utility of our approach in measuring linkability between profiles and among published posts.

In order to accomplish our research objective, we used two main methods to evaluate linkability. The first strategy concentrated on looking at user profiles' related structured data, whereas the second strategy looked at unstructured data. We sought to obtain a thorough grasp of linkability trends by methodically examining both types of data. We then used the results from these two methodologies to make a firm judgement on the degree of linkability between user IDs. We were able to examine and quantify the linkability of profiles across various OSNs using this integrated technique.

3.2 Background and related works

For calculating the privacy leakage score, numerous scholars have suggested multitude of methodologies among which two major approaches are significant namely statistical-based and machine learning based models. The first method relies on two intuitive properties which are sensitivity and visibility. This statistical-based method works on Dichotomous variables (takes only one of two possible values) or Polytomous variables (having more than two possible categories, either ordered or unordered) or a combination of them. On the other hand, ML-based models mostly try to measure the privacy of unstructured data (e.g., text, image, video).

One of the pioneering efforts towards the development of a privacy metric for online social networks was put forward by Maximilian et al. [21] in 2009 where the authors have proposed a formulation (1) to calculate the privacy score based on the sensitivity β_i and the visibility $v(i, j)$ of profile items $i \in \{1, \ldots, n\}$ of user j in a social network.

$$PR(i) = \sum_i PR(i, j) = \sum_i \beta_i \times v(i, j) \tag{1}$$

On the other hand, over the past decade, there has been growing interest in the analysis of linkability between user profiles on online social networks (OSNs) [2]. Linkability refers to the ability to link or associate various pieces of information or activities to a specific individual or entity, even if that information or activity is intended to be anonymous or separate [32]. A number of previous studies have explored various aspects of this issue, such as, identification of common patterns in user behavior across multiple platforms for measuring linkability scores [20]. These efforts have contributed to a greater understanding of the privacy risks associated with social media use and have laid the groundwork for the development of more effective privacy protection measures. In order to identify the linkability among two different sources, Chandok S. [5] proposed two separate methods which can be used against identities drawn from same or separate OSNs. First, weighted sum method, where the linkability score is calculated based on the similarity of feature using a function of feature and metrics. In this method, Computation of linkability score is performed in two steps namely feature similarity indicator and linkability score calculator. Once the weights have been assigned, the weighted sum score is calculated for each pair of profiles. The score represents the degree of linkability between the profiles, with higher scores indicating a greater risk of privacy leakage. Second, probabilistic method, where the intuitive idea is that the linkability score relies probabilistically on the feature similarity values. This approach defines linkability score as the probability of discovering two identities that are identical based on how similar their attributes are. Goga et al. [13] found that it is possible to link a user across multiple OSNs using information inherited from posted content. The researchers focused on components such as geo-location, post timestamp, and writing style to analyze their approach, using Yelp, Twitter, and Flickr as examples. Labitzke et al. [18] had shown the possibility of profile correlations based on extracted friends lists even under legal and technical constraints.

However, upon thorough review of existing literature pertaining to the computation of privacy leakage scores in association of the consideration of linkability, limited references were identified. Hassanpour et al. [15] suggested that the linkability between posts or actions can significantly impact privacy, leading to an increase or decrease in the Privacy Leakage Score (PLS). He proposed a formula for calculating the privacy leakage score considering linkage score along with visibility and sensitivity.

$$\text{PLS} = \text{sensitivity} \times \text{linkage} \times \text{visibility} \tag{2}$$

The scarcity of relevant studies in this domain suggests a research gap in comprehensively addressing this particular aspect. The paucity of prior work underscores the need for further exploration and development of methodologies for accurately quantifying privacy leakage scores. By acknowledging this knowledge gap, our study contributes to the existing body of research by proposing novel approaches and methodologies in the calculation of privacy leakage score.

3.3 Design experiment

This section expounds on our preliminary computations for the linkability score metric, which serves as a measure of the degree of association between two distinct online identities belonging to the same user on different online social networks (OSNs). Specifically, we delve into the derivation of features from user provided information and activity, which are subsequently used to construct activity profiles. The efficacy of the linkability score is evaluated by examining how accurately it estimates the extent to which two given identities are linkable. This evaluation helps us understand the usefulness of the linkability score in identifying potential privacy breaches and mitigating them.

3.3.1 Targeted OSNs

In our study, we sought to test the efficacy of our proposed linkability score metric across multiple OSNs. To accomplish this, we created datasets from three distinct OSNs, Facebook, Twitter and Flickr. By employing diverse datasets, our study aims to transcend platform-specific and user-specific limitations, enabling broader generalizability of our findings across various user populations and online contexts. Here, we overview three OSNs used in our study.

Facebook is a social networking platform that allows users to create a personal profile, share text, photos and videos, connect with friends and family, join interest groups, and engage in various activities such as playing games and participating in online events. The Facebook users can decide to whom he wants to limit the information that he is publishing through customized option for privacy. As of 2022, it has the highest number of monthly active users among all online social networks worldwide, with approximately 2.96 billion users [10]. Every minute, about 400 new users register on Facebook. Simultaneously, over 510,000 comments, 293,000 status updates, and 136,000 photos are posted, with 4 million posts being liked [27].

Twitter is a micro-blogging OSN where registered users (known as tweeters) post short messages (called tweets), which can be include text, photo, or videos. Some Twitter users choose to make their tweets public, making them accessible to anybody, even those without a Twitter account. Whereas others only allow their so-called followers, or Twitter users who have specifically asked for and received access to their tweets. Political figures, journalists, sportsmen, and other celebrities have all joined Twitter, making it one of the most widely used and diversified OSNs today. As of 2022, Twitter has a monthly active user base of around 450 million, showing an audience growth of over 40% since 2018 [31].

Flickr is an online social network and cloud storage provider, specializing in the sharing of multimedia content, specifically photographs and videos. This platform allows users to annotate their multimedia content with text, which enhances the user experience by providing additional context to the content. In order to post or view restricted content on Flickr, an account is generally required. However, public content can be viewed by anyone without an account. Additionally, Flickr has a unique feature known as contacts, which is similar to the concept of friends or connections on Facebook and Twitter. As of 2022, the registered user base of Flickr exceeds 112 million, with 60 million being categorized as active users, defined as those who access the platform at least once a month [4].

3.3.2 Collected data

Our experiment was conducted using a dataset that consisted of both structured and unstructured data, which was obtained from Facebook, Twitter, and Flickr. As a measure to ensure compliance with the General Data Protection Regulation (GDPR), we exclusively extracted publicly posted information of users from these platforms. However, the initial challenge that we encountered was to identify users who had accounts across all three of the aforementioned OSNs. In order to overcome this obstacle, we leveraged the feature on Flickr that allowed users to mention their associated user accounts on other online social networks. This feature was instrumental in identifying our target users, which served as the ground truth for the dataset that we acquired.

Out of a random selection of 5,473 Flickr users, we were able to sort out 45 users who possessed registered accounts across all our targeted OSNs. The structured data that we collected consisted of users' name, location (both current and hometown), and user name, occupation and some other publicly available information. Alongside this, we obtained unstructured data that included bio, texts, images, and image captions. Our efforts in acquiring this data were aimed at analyzing the linkability score across diverse user pairs and evaluating the effectiveness of the score in estimating the likelihood of linkability between two given identities.

Moreover, we gathered a total of 2264, 1684, and 693 images from Flickr, Facebook, and Twitter, respectively, sourced from 45 users across these platforms. We employed web scraping tools and APIs provided by each social media platform to gather the data. Special care was taken to respect the rate limits and terms of service of these platforms.

3.3.3 Methodology

To ascertain the linkability between users' profiles, we employed two distinct approaches. The first approach involved user profiling, where we leveraged the structured data openly shared by the users. By analyzing attributes such as name, user name, and location, we aimed to link profiles that capture the essence of each user. The second approach focused on identifying the content correlation within an individual's data. This involved examining the relationships between published image files, to unveil patterns and associations that contribute to the linkability between profiles. By combining these approaches, we aimed to gain a comprehensive understanding of the linkability dynamics present in users' online profiles.

1) User Profiling Measuring: In order to calculate the similarity of the attributes from different OSNs, we used the bert-base-nli-mean-tokens model [25]. BERT (Bidirectional Encoder Representations from Transformers) is a pre-trained language model developed by Google [9] that uses a bidirectional transformer architecture to create deep contextual representations of words in text. It has shown state-of-the-art performance in various natural language processing tasks, such as text classification and sentence similarity [35]. 'Bert-basenli-mean-tokens' stands for BERT base model for natural language inference using mean pooling of the token embeddings which is a fine-tuned version of the original BERT language model.

Initially, we performed similarity calculations on the users belonging to the same OSN. We used cosine similarity metrics to calculate similarity. These similarity calculations were performed on various attributes of the users such as their name, location, and user name. Once we had obtained the similarity values for each attribute, we combined them to obtain an overall similarity score. To achieve this, we took the average of the similarity scores across all attributes for each pair of users. After calculating the similarity values for users within the same OSN and obtaining the average of the similarity values based on selected attributes, we proceeded to calculate the similarity for cross-OSN users. For this, we employed the same method of using the BERT-based embedding model to obtain the vector representations of the user attributes. Next, we computed the cosine similarity (3) between the vector representations of each pair of users from different

OSNs. This resulted in a similarity score for each pair of users that belonged to different OSNs. These similarity scores were then normalized to obtain a value between 0 and 1 . From the matrix of calculated values, we used Top-1 (identifying the single best choice or outcome among multiple options) and Top3 approach (considering 3 best choices instead of just one among multiple options) to identify the best linkability among the entities. For Top-1 approach, only the single best choice or outcome among multiple options would be considered, whereas for a Top-3 approach the three best choices would be included instead of just the best one [3].

Overall, this approach allowed us to estimate the linkability between users across different OSNs by leveraging the similarity between their attributes, as captured by the BERT-based embedding model.

2) Content Linkage Measuring: In this work, we compared and measure the linkilibity of posts' content between Flickr, Facebook, and Twitter for a specific user. We consider the image modality since Flickr manily are being used for posting pictures. To measure the linkilibilty between each pair of images, we first extract a representation vector for each image using a deep learning model called EfficientNetV2 [30] which is trained on ImageNet dataset [8]. Then, to calculate the similarity between each pair of images, we utilize cosine similarity as below:

$$\text{similarity } = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} \tag{3}$$

where $\mathbf{A}$ and $\mathbf{B}$ are the extracted representation vectors for the first (from Flickr) and the second (from Twitter or Facebook) images, respectively. For each image in Flickr, we found the most similar image in Facebook (or Twitter when we are comparing posts' content in Flickr and Twitter).

3.4 Results

This section provides an overview of the results obtained from the proposed linkability scoring methods. In particular, we present the computed linkability scores for the selected approaches and analyze the outcomes. These results serve as a basis for evaluating the effectiveness of the methods and their potential for accurate identity and content linkage across multiple OSNs.

3.4.1 Data analysis

In order to gain a better understanding of the collected data for the 45 shortlisted users, we conducted some statistical analyses. In our dataset we found 28.89% female and 71.11% male population who were selected totally ran-

Table 3.1: Gender distribution in dataset

Gender	Percentage
Male	71.11%
Female	28.89%

Table 3.2: Location disclosure rate

OSN	Location Disclosure
Facebook	75.55%
Twitter	93.33%
Flickr	97.77%

Table 3.3: Occupation disclosure

OSN	Work Info Disclosure
Facebook	66.66%
Flickr	73.33%

domly (Table 3.1). In contemporary times, online social networks have become

integral to daily life, and individuals frequently disclose their personal information, including contact details, birthdays, relationship statuses, and political and religious affiliations. While some may inadvertently reveal sensitive information, others may do so intentionally.

In our obtained dataset, users disclosed their personal information to various extend. Such as, the ratio of location (hometown and/or current location) disclosure is 75.55% for Facebook, 93.33% for Twitter and 97.77% for Flickr (Table 3.2). Among all, 73.33% of Flickr users shared their occupation and among the Facebook users, 66.66% disclosed that (Table 3.3). Users also shared bio which typically refers to a short written description or summary that a user includes on their profile to provide information about themselves. We found that **97.77**% Flickr users, 88.88% Twitter users and 60% Facebook users shared this type information on their profile which can directly cause privacy leakage if displayed to unintended audience (Table 3.4). We also had access to some other sensitive information as the users shared those publicly. Such as, 35.55% users shared their relationship status, 6.66% users disclosed their date of birth and 20% users mentioned their email address within the OSNs (Table 3.5). Among the unstructured data, every user shared images in their profile (at least one) but there was some variations for text data they shared. We also found 4.44% users who intentionally chose to protect their shared content from the mass public in Twitter.

Table 3.4: Short user introduction rate

OSN	Bio Disclosure
Facebook	60%
Twitter	88.88%
Flickr	97.77%

Table 3.5: Other sensitive info

Sensitive Info	Disclosure
Relationship status	35.55%
Date of birth	6.66%
Email address	20%

Table 3.6: Facebook vs. Flickr & Twitter

OSN	Accuracy against Facebook	
Name	Top-1	Top-3
Flickr	88.88%	91.12%
Twitter	71.11%	80%

3.4.2 Profiling linkage performance

After conducting our experiment on the acquired dataset, we found that a significant portion of the users were linkable using our proposed method. Specifically, our approach was able to identify connections between users across multiple online social networks with a high degree of accuracy. However, it is important to note that accuracy may vary depending on the specific attributes and features used in the analysis. In order to fully understand the effectiveness of our approach, we analyzed the accuracy rate that we obtained during the experiment.

During the evaluation process, we meticulously compared each online social network with the others and discovered varying accuracy rates for each pair of OSN. Such as, while calculating the linkability for Facebook against Flickr we found 91.12% accuracy and against Twitter 80% accuracy based on similarity score.

For the analysis of Flickr against Twitter we obtained 95.56% accuracy and against Facebook 97.78%.

During the calculation of Twitter against Flickr we get 97.78% accuracy for and Facebook we get 91.12% accuracy.

3.4.3 Content linkage performance

After linking user profiles, we generate separate poll of images for each user in different OSNs. Thus, using similarity score, the most similar image for

Table 3.7: Flickr Vs. Twitter & Facebook

OSN Name	Accuracy against Flickr	
	Top-1	Top-3
Twitter	88.88%	95.56%
Facebook	95.55%	97.78%

Table 3.8: Twitter Vs. Flickr & Facebook

OSN Name	Accuracy against Twitter	
	Top-1	Top-3
Flickr	91.11%	97.78%
Facebook	91.11%	91.12%

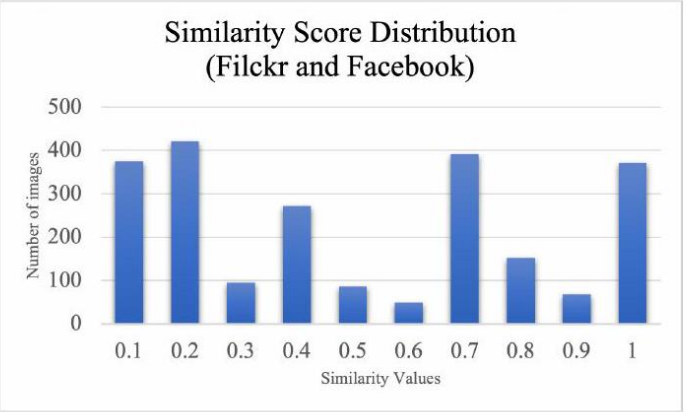


Figure 3.1: Similarity score distribution between Flickr (including 2264 images) and Facebook (including 1684 images) images. The x and y axis show the cosine similarity values and number of images, respectively.

each image in Flickr has been found in Facebook for each user. We done the same process between Flickr and Twitter polls. The distribution similarity score for both cases (i.e., Flickr-Facebook and Flickr-Twitter) have been shown in Figures 3.1 and 3.2. A similarity score between 0.9-1 shows a high degree of similarity between the images being compared. In this context, we operate under the assumption that a similarity score exceeding 0.8 (estimated empirically) indicates nearly identical images, distinguished by only a minimal degree of variation. Considering this threshold, as depicted in Figure 1, approximately 450 images that were shared on Flickr have also been published on Facebook.

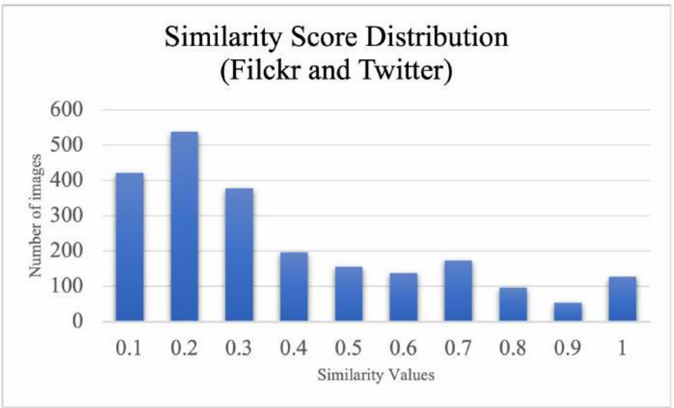


Figure 3.2: Similarity score distribution between Flickr (including 2264 images) and Twitter (including 693 images) images. The x and y axis show the cosine similarity values and number of images, respectively.

3.5 Discussion

The first part of our experiment aimed at linking profiles across various Online Social Networks (OSNs) using minimal profile attributes, specifically, name, user name, and location. Evidently, the accuracy of profile linkage would potentially increase if additional information was incorporated into the process. As can be deduced from Tables 3.6, 3.7, and 3.8, taking into account the top-1 accuracy, the most accurate results were achieved when information from Flickr profiles was used to establish a link with a corresponding profile on Facebook. This suggests a high degree of similarity or overlap between the information disclosed on Flickr and Facebook profiles. In simpler terms, it appears that users tend to share closely matching information across these two platforms. Therefore, if you have access to a user's Flickr profile, there's a higher likelihood of accurately linking it to the same user's Facebook profile compared to Twitter. This observation emphasizes the impact and value of shared data across multiple social media platforms in enhancing the accuracy of profile linkage.

Furthermore, the latter segment of our study indicates that a significant number of images shared on Flickr do not appear on other Online Social Networks (OSNs). This situation can result in a heightened risk of privacy breaches if profiles across different OSNs are linked. Take, for example, a sample of 2264 images uploaded on Flickr. Our findings reveal that roughly 20 percent of these images were also found on Facebook, and a smaller portion, about 8 percent, surfaced on Twitter. These statistics suggest a lower likelihood of successfully establishing a link between profiles on Flickr and Twitter due to the reduced overlap in shared content. However, it's essential to note that while the probability of linking is lower, the potential for privacy leakage escalates dramatically. The reason being, the content shared on these two platforms is distinctly different. Therefore, if a link is established, it would expose a broader range of the user's information, potentially revealing aspects of their personal lives that they intended to keep separate on these individual platforms. This underscores the critical need for users to be conscious of the data they share across different social networks, given the potential risks associated with profile linkage across multiple platforms.

It is important to clarify that our analysis on content linkage is currently focused solely on the image modality. However, this does not limit the application of our techniques. They can indeed be adapted for other unstructured data types, such as text and video. This would entail using appropriate deep learning models to extract representative vectors from these data types, and then leveraging cosine similarity as a measure of distance between these vectors, much like we have done with images.

3.6 Conclusion

Our research on Online Social Networks (OSNs) focuses on the privacy implications arising from linkability of user profiles and shared content across different platforms. We developed a method to quantify this linkability, utilizing key attributes (i.e., name, user name, location) to determine profile similarities. Applying this to profiles and content from Flickr, Facebook, and Twitter, we examined both structured and unstructured data, offering a valuable view of linkability trends and potential privacy risks. Our findings highlight that minimal profile attributes can significantly enhance the accuracy of profile linkages, particularly between platforms like Flickr and Facebook where data overlap is significant. However, we also found a substantial number of images shared on Flickr do not appear on other OSNs, reducing the likelihood of profile linkage but paradoxically increasing potential privacy leakage. This discovery underscores the need for robust privacy measures given the increased interconnectedness of user data across OSNs. Our research emphasizes the importance of user consciousness in data sharing across different OSNs, considering the potential privacy risks of profile linkage. Our work aims to inform the development of privacy-enhancing technologies and strategies to better protect user privacy in OSNs.

References

[1] E. Aghasian, S. Garg, L. Gao, S. Yu, and J. Montgomery, "Scoring users' privacy disclosure across multiple online social networks," IEEE Access, vol.

5, pp. 13118–13130, 2017.

[2] M. Backes, P. Berrang, O. Goga, K. P. Gummadi, and P. Manoharan, "On profile linkability despite anonymity in social media systems," in Proceedings of the 2016 ACM on Workshop on Privacy in the Electronic Society, 2016.

[3] M. Benedek, C. Mühlmann, E. Jauk, and A. C. Neubauer, "Assessment of divergent thinking by means of the subjective top-scoring method: Effects of the number of top-ideas and time-on-task on reliability and validity," Psychol. Aesthet. Creat. Arts, vol. 7, no. 4, pp. 341–349, 2013.

[4] Matic Broz. Flickr Statistics, User Count, amp; Facts (July 2023). 8 2022.

[5] S. Chandok and P. Kumaraguru, User identities across social networks: quantifying linkability and nudging users to control linkability. 2017.

[6] J. Chen, J. He, L. Cai, and J. Pan, "Disclose more and risk less: Privacy preserving online social network data sharing," IEEE Trans. Dependable Secure Comput., vol. 17, no. 6, pp. 1173–1187, 2020.

[7] J. Sourya and A. De, "Choosing the right privacy settings," in Privacy Risk Analysis of Online Social Networks, Springer, 2021, pp. 59–67.

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009.

[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional Transformers for language understanding," arXiv [cs.CL], 2018.

[10] S. Dixon. Facebook MAU worldwide 2023 - Statista, 52023.

[11] J. Domingo-Ferrer, "Rational privacy disclosure in social networks," in Modeling Decisions for Artificial Intelligence, Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 255–265.

[12] M. Eirinaki and M. Vazirgiannis, "Web mining for web personalization," ACM Trans. Internet Technol., vol. 3, no. 1, pp. 1–27, 2003.

[13] O. Goga, H. Lei, S. H. K. Parthasarathi, G. Friedland, R. Sommer, and R. Teixeira, "Exploiting innocuous activity for correlating users across sites," in Proceedings of the 22nd international conference on World Wide Web, 2013.

[14] R. Gross and A. Acquisti, "Information revelation and privacy in online social networks," in Proceedings of the 2005 ACM workshop on Privacy in the electronic society, 2005.

[15] A. Hassanpour and B. Yang, "PriMe: A novel privacy measuring framework for online social networks," in 2022 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 2022.

[16] N. Kökciyan and P. Yolum, "P ri g uard: A semantic approach to detect privacy violations in online social networks," IEEE Transactions on

Knowledge and Data Engineering, vol. 28, no. 10, pp. 2724–2737, 2016.

[17] N. Kökciyan and P. Yolum, "P ri g uard: A semantic approach to detect privacy violations in online social networks," IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 10, pp. 2724–2737, 2016.

[18] S. Labitzke, J. Dinger, and H. Hartenstein, "How i and others can link my various social network profiles as a basis to reveal my virtual appearance," DFN-Forum Kommunikationstechnologien. Gesellschaft für Informatik eV, vol. 4, 2011.

[19] Y. Li, Y. Li, Q. Yan, and R. H. Deng, "Privacy leakage analysis in online social networks," Comput. Secur., vol. 49, pp. 239–254, 2015.

[20] S. Liu, S. Wang, F. Zhu, J. Zhang, and R. Krishnan, "Hydra: Large-scale social identity linkage via heterogeneous behavior modeling," in Proceedings of the 2014 ACM SIGMOD international conference on Management of data, 2014, pp. 51–62.

[21] T. Michael Maximilien, T. Grandison, D. Sun, S. Richardson, and K. Guo, "Privacy-as-a-service: Models, algorithms, and results on the facebook platform," Proceedings of Web, vol. 2, 2009.

[22] R. Kumar Nepali and Y. Wang, "Sonet: A social network model for privacy monitoring and ranking," in 2013 IEEE 33rd International Conference on Distributed Computing Systems Workshops, IEEE, 2013.

[23] K. Nissim, S. Raskhodnikova, and A. Smith, "Smooth sensitivity and sampling in private data analysis," in Proceedings of the thirty-ninth annual ACM symposium on Theory of computing, 2007.

[24] S. Oukemeni, H. Rifà-Pous, and J. M. M. Puig, "Privacy analysis on microblogging online social networks: A survey," ACM Computing Surveys (CSUR), vol. 52, no. 3, pp. 1–36, 2019.

[25] L. Passaro, A. Bondielli, A. Lenci, and F. Marcelloni, "Unipi-nle at checkthat! 2020: approaching fact checking from a sentence similarity perspective through the lens of transformers," in CEUR WORKSHOP PROCEEDINGS, vol. 2696, CEUR, 2020.

[26] C. Renner, "Privacy in online social networks," Swiss Federal Institute of Tech, pp. 11–13, 2010.

[27] J. Shepherd, 33 Essential Facebook Statistics You Need To Know In. 2023.

[28] A. Srivastava and G. Geethakumari, "Measuring privacy leaks in Online Social Networks," in 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2013.

[29] D. Frederic, R. Stutzman, and A. Gross, "Silent listeners: The evolution of privacy and disclosure on facebook," Journal of privacy and confidentiality, vol. 4, no. 2, 2013.

[30] M. Tan and Q. V. Le, "EfficientNetV2: Smaller models and faster training," arXiv [cs.CV], 2021.

[31] Ash Turner. How Many Users Does Twitter Have? (Jul 2023), 62023.

[32] Y. Shin Van Der Sype and W. Maalej, “On lawful disclosure of personal user data: What should app developers do?,” in 2014 IEEE 7th International Workshop on Requirements Engineering and Law (RELAW), IEEE, 2014, pp. 25–34.

[33] C. Wang, Z. Tianqing, P. Xiong, W. Ren, and K. R. Choo, “A privacy preservation method for multiplesource unstructured data in online social networks,” Computers Security, vol. 113, 2022.

[34] P. Wisniewski et al., “Framing and measuring multi-dimensional interpersonal privacy preferences of social networking site users,” Commun. Assoc. Inf. Syst., vol. 38, pp. 235–258, 2016.

[35] S. Wu and M. Dredze, bentz, becas: The surprising crosslingual effectiveness of bert. 2019.

Chapter 4

E2F-GAN: Eyes-to-Face Inpainting via Edge-Aware Coarse-to-Fine GANs

Ahmad Hassanpour, Amir Etefaghi Daryani, Mahdieh Mirmahdi, Kiran Raja, Bian Yang, Christoph Busch, Julian Fierrez

Abstract

Face inpainting is a challenging task aiming to fill the damaged or masked regions in face images with plausibly synthesized contents. Based on the given information, the reconstructed regions should look realistic and more importantly preserve the demographic and biometric properties of the individual. The aim of this paper is to reconstruct the face based on the periocular region (eyes-to-face). To do this, we proposed a novel GAN-based deep learning model called Eyes-to-Face GAN (E2F-GAN) which includes two main modules: a coarse module and a refinement module. The coarse module along with an edge predictor module attempts to extract all required features from a periocular region and to generate a coarse output which will be refined by a refinement module. Additionally, a dataset of eyes-to-face synthesis has been generated based on the public face dataset called CelebA-HQ for training and testing. Thus, we perform both qualitative and quantitative evaluations on the generated dataset. Experimental results demonstrate that our method outperforms previous learning-based face inpainting methods and generates realistic and semantically plausible images. We also provide the implementation of the proposed approach to support reproducible research via (https://github.com/amiretefaghi/E2F-GAN).

INDEX TERMS Face inpainting, generative adversarial networks, image inpainting.

4.1 Introduction

Image inpainting is used to complete missing information or substituting undesired regions of pictures with conceivable and fine-grained content. It encompasses a wide extend of applications in fields of restoring harmed photos, editing pictures, removing objects, etc. [1], [2]. Many conventional methods typically use low-level and hand-crafted features from the cor-

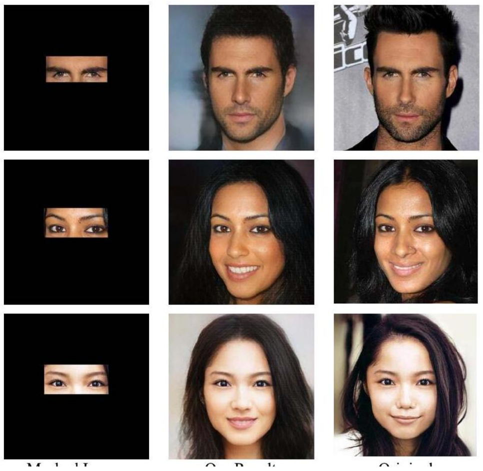


Figure 4.1: Example completion results of our proposed method in comparison with original images.

rupted input image and utilize the priors or additional data. By propagating the extracted features from visible and well-structured parts to the missing regions or by filling missed small areas by looking and melding comparative patches from the same or other images. In spite of the fact that these strategies have great effects in the completion of replicating structures, they are restricted by the accessible regions in an image and cannot create novel image substance. In recent years, learning-based strategies have been proposed to overcome these confinements by utilizing huge volumes of training data [3], [4]. Notably, despite of great achievements of learning-based methods in this task, they are limited by at least three challenges: the inpainted area should be C1) semantically filled based on overall scene, C2) continuously structured with unmasked regions, and C3) visually realistic.

Recently, deep convolutional neural networks (CNNs) and generative adversarial networks (GANs), known as learning-based methods, have been widely used for various image inpainting tasks including removing objects, noises, texts, and masks. Based on convolutional neural networks (CNNs) and using encoder-decoder network structure several works have been proposed for image inpainting [5]-[8]. For instance, Sidorov and Hardeberg [6] proposed an encoder-decoder network for denoising, inpainting and super-resolution for noised, inpainted and low-resolution images. Zhu et al. [5] proposed a patch-based inpainting method for various deep learning (DL) modules that have been proposed recently.

Coarse-to-fine based methods exploit one [9]-[11] or two [12]-[14]-stage architecture to complete content formation and texture refinement. A one-stage architecture (also termed coarse-and-fine architecture) consists of two parallel branches, coarse and fine, that extract two kinds of information simultaneously, coarse and fine information. The missed region, then, can be constructed from the extracted information. Alternatively, a two-stage architecture generates an intermediate coarse image after recovering structures in the first stage, and then feeds it to the second stage for improving the texture. Additionally, another category called structural guidance-based methods uses an assistance algorithm to provide more information for the main inpainting method. An edge and a contour generator have been used within a two-stage architecture in [15] and [16] respectively.

Although, it is worthy to mention that in face inpainting, besides the above-mentioned challenges (i.e., C1-C3), we are facing further requirements. Notably, a facial representation can be considered for the purpose of biometric recognition due to the special topology of different facial elements (i.e., forehead, eyes, eyebrows, nose, mouth, jaw, chin, cheek) and their distinctive characteristics [42]. Thus, revealing the hidden parts of a face by using other elements such that the topological face elements along with consistency in face attributes (e.g., demographic and other biometric information [43]) are preserved is a challenging task, yet it will have a strong impact on the feasibility of biometric recognition conducted by human experts (i.e. in forensic investigation [44]) or by machine learning [45] or hand-crafted algorithms [46]. Therefore, the requirements of face inpainting are as follows:

R1) the face topological structure should be reconstructed so that all elements are placed in the right position semantically and continuously. For this, first, the shape of the face (oval shape, square shape, round shape, etc.) should be predicted. Then all other elements should be placed proportionally within the predicted frame. Additionally, to look more realistic, the head pose should be naturally aligned and integrated with other elements. These requirements are the main challenges (i.e., C1-C3) of every inpainting method modified for face inpainting solutions. Since the aim of this paper is a special case of face inpainting where a large region of the face except eyes is hidden, besides R1, two other requirements which make the inpainting task more challenging should be considered. R2) Researchers have found that the area of skin around the eyes is useful to determine soft biometric information such as age or gender [17], [37]. The proposed inpainting model should utilize the color, texture, and size of eyes and eyebrows to estimate this kind of demographic attributes and inpaint other face elements according to the estimated features. R3) The proposed solution should preserve the identity-related biometric properties present in the eyes regions [18], [38] when generating the full face [39]. Noteworthy, this eye region is demonstrated to encode a large part of the identity information present in the face [44] enabling both person recognition and fake face detection [40].

Additionally, it is worth to mention that the hidden portion of the image can directly affect the performance of proposed solutions, and clearly large masks make meeting the referred requirements (i.e., R1-R3) more difficult. Considering this issue, the aim of this paper is to complete the face based on the eyes region (periocular region), our used mask type will cover most parts of the face.

In this paper, a novel DL-based architecture has been proposed such that it complies with the referred requirements (i.e., R1-R3, see Figure 4.1). Therefore, our contributions and novelties can be summarized as follows:
- In this work, an effective end-to-end solution for reconstructing the face based on just the eyes region has been proposed. This innovative GAN-based architecture called E2F-GAN benefits from the advantages of coarseto-fine, coarse-and-fine, and structural guidance-based architectures. The code for our proposed method is available in GitHub. [1]
- By using various loss functions during the training process [41], not only the quality of inpainted regions but also demographic and biometric features have been preserved and measured by several quantitative and qualitative evaluation metrics.
- A new dataset of masked faces called E2Fdb has been generated and made publicly available (same GitHub indicated before).
- In terms of selecting the most informative guidancebased method, we experimentally show that edges provide more structural and contextual information compared to landmarks.

4.2 Related works

In eyes-to-face inpainting, a face (a raw image indicated by $I^{H \times W \times N}$ hereafter) is corrupted by a binary image mask $\left(M^{H \times W \times N}\right)$, where H, W, and N show the height, width, and number of channels of the image respectively, and the corrupted image will be shown by I_m ($I_m = I \odot M$, where $\odot$ is the element-wise production). The inpainting model H takes I_m and M as input, and its output, reconstructed face, should fulfill the R1, R2, and $\mathbf{R3}(I \cong \hat{I})$. The proposed inpainting methods use different architectures and various types of masks. In this section, we review recent face inpainting methods based on DL architectures and widely used mask types.

4.2.1 Face inpainting methods

Apart from traditional methods which utilize low-level features extracted from the same image or a group of images, the learning-based strategy is the main focus of recent proposed methods due to using high-level features that enable them to inpaint the damaged regions semantically. In the following, we review several learning-based existing works that attempted to inpaint corrupted faces, similar to the aim of this paper.

The coarse-to-fine structure has been used in recent face inpainting tasks. Li et al. [19] proposed a generative-based coarse-to-fine structure that benefits from an attention layer to capture long dependency between features to generate more realistic images. Yu et al. [13] uses a coarse-to-fine structure to inpaint free-form masks. In the same context, Liu et al. [12] proposed a coarse-to-fine architecture with a novel attention layer. Chen et al. [19] proposed a coarseand-fine structure including a coarse network for extracting global semantic information and a fine network to extract multi-level local features. Besides the coarse-to-fine based strategies, another category so-called structural guidance uses additional information to assist the main inpainting module. Nazari et al. [15] leverage an edge generator first to recover the edges, and the corrupted image is fed to the image inpainting network along with predicted edges. Chen and Liu [16] use a dual branch network including texture and edge branches to extract features and recover structures and textures of missed regions. Some works estimate facial landmarks to assist the main inpainting network [20], [21]. In this paper, we will take the advantages of different architectures, i.e., coarse-to-fine, coarse-and-fine, and structural guidance.

The above-mentioned methods produce a unique result per each input. On the other hand, some approaches inpaint the corrupted regions differently per each execution for each specific input. Zheng et al. proposed a Variational AutoEncoders (VAEs)-based [22] dual pipeline including a reconstructive path that uses the ground truth to learn the prior distribution of missing regions and a generative path for which the conditional prior is connected to the distribution obtained in the reconstructive path. An unsupervised conditional framework based on generative adversarial networks for varied image inpainting that can learn conditional completion distribution has been proposed by Zhao et al. [23]. A similar approach using GANs to restore low quality face images was recently proposed in [47]. It should be noted that, in E2F-GAN, we need a unique output for each input even after several executions to fulfill the requirements R2 and R3.

4.2.2 Mask coverage

The used masks in face inpainting scenarios can be classified into two categories called free and fixed-form masks. In widely used free-form masks [8], [10], [15], [21], [22], [24], [26], there are irregular shapes randomly placed on the images (Figure 4.2a). Instead, in the fixed-form masks [13], [21], [24], [25], regular shapes cover part of the images which are located on the images randomly or purposefully (Figure 4.2b) [24], [25]. Since the aim of this paper is to complete the face based on eyes, our used mask type is in the latter category with a large-size mask ($\approx 75\%$ of the face).

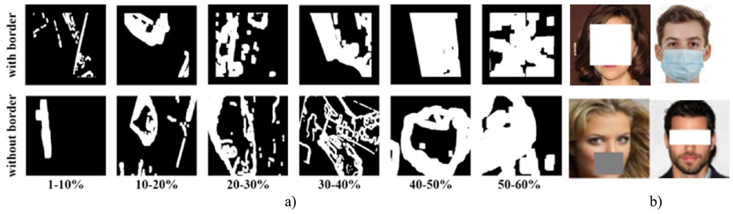


Figure 4.2: Examples of two types of widely used masks called free-form [26] (a) and fixed-form [24], [25] (b).

4.3 Proposed method

The overall network architecture of our proposed method, which is based on a coarse-to-fine architecture and includes two main modules called coarse and refinement, is shown in Figure 4.3. Different from others [2], [3], [7], [13], [19], both modules (i.e., coarse and refinement) are GAN-based networks, therefore, each of which includes a generator and a discriminator. The coarse module, which comprises a generator called coarse generator (C), has a dual encoder that follows the coarse-and-fine structure to capture global semantic features and extract multi-level features from the eyes region. Besides this module, a GAN-based refinement module which consists of a refinement generator (F) and a discriminator (D_2) has been utilized to improve the coarse outputs. Intuitively, the refinement network sees a more completed scene than the masked images, so its network can learn better feature representations than the coarse network. Therefore, our end-to-end method includes two GAN-based modules which are training to generate the final result. In the following subsections, each module is described in detail.

Notably, facial landmarks [21] or edges [15] are usually the most widely-used structural guidance in image inpainting tasks. In our proposed E2F-GAN, where the used mask covers most parts of the face, predicting both landmarks and edges is a challenging problem. As a consequence, our proposed method utilizes both landmarks and edges during our experiments, in an effort to use the most effective structure (e.g., landmarks or edges). For facial landmarks, we used the landmark prediction method proposed in [27] and for predicting edges, we used the edge predictor proposed by Nazari et al. [15]. Both methods have been trained again on our generated dataset that contains specific eye masks. As we will see in the experiments, our quantitative and qualitative metrics will show that the edge structural guidance provides more effective information for our coarse generator.

Therefore, in our final setup we use edges generated by an edge predictor (E_e) as structural guidance for C.

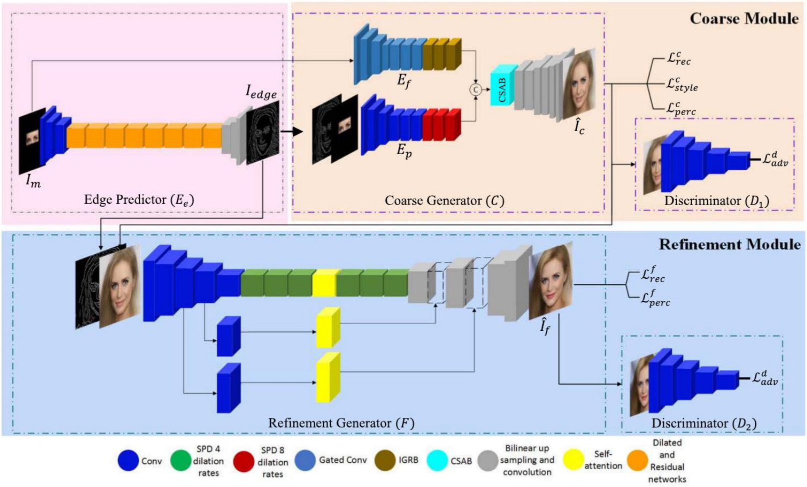


Figure 4.3: Overview of our architecture with three main modules including Edge Predictor (E_e), Coarse Module (C), and Refinement Generator ($\boldsymbol{F}$).

4.3.1 Coarse module

The proposed GAN-based coarse module is responsible for extracting the required features from the masked image and generating the first coarse result. To do this, we designed the module with three submodules including edge predictor (E_e), coarse generator (C), and discriminator (D_1). In the following, we explain the role of each network, its architecture, and the used loss functions.

4.3.1.1 Coarse generator

The coarse generator has the main responsibility for meeting the three requirements (i.e., R1-R3). Not only the biometric and demographic feature should be extracted from the periocular region, but also the initial coarse prediction should look realistic, and semantically and continuously structured. This is achieved using three networks: two encoders so-called fine encoder (E_f) and pose encoder (E_p), and a decoder. The encoder E_f deals with the finest features of I_m and E_p deals with the predicted structure of faces obtained from E_e. Therefore, first I_m is fed to E_e to predict edges of visible and hidden regions (I_{edge}) and then I_{edge} is concatenated with I_m to fed E_p. This assists to predict the pose of different elements of the face. Additionally, I_m will be fed to E_f with the aim of extracting identity attributes. Finally, the decoder will predict and inpaint the hidden regions based on the two feature maps received from E_f and E_p. In the following, we describe each of these networks and their roles in our scheme.

4.3.1.2 Fine encoder

The aim of using this encoder is mainly to extract demographic (e.g., age, gender) and biometric properties (e.g., identity, skin color) from I_m. Therefore, the skin color around the eyes, wrinkles, the size of eyes and eyebrows, the distance between two eyes, and other possible properties should be considered. On the other hand, it should be noted that due to the high coverage ratio of I_m, E_f is fed with a lot of unusable information (the black region). To prevent deteriorating the quality of output and filter out these pixels, the first seven blocks of E_f are configured as with gated convolutions (GC) [14]. These blocks contain parallel convolution layers with different sorts of activation functions which assist to extract an appropriate feature map and eliminate extracted features from the masked region. Then, three interleaved gated residual blocks (IGRB) [19] have been placed after GC blocks to extract multi-level features.

4.3.1.3 Pose encoder

For extracting coarse structure and global semantics features, and consequently preserving the quality as well as the structure of the predicted face, an encoder called pose encoder (E_p) has been placed in the Coarse Module (C). It has been fed by concatenation of I_{edge} and I_m. Doing this, a receptive field for recognizing face structures will be available for E_p. However, the inputs I_{edge} and I_m are both sparse. To extract a meaningful feature map, similar to [29], we used three spatial pyramid dilation blocks (SPD) after six convolution layers. Notably, SPD blocks contain parallel convolution layers with various dilation rates to extract a large receptive field from the given input image.

4.3.1.4 Decoder

To inpaint the coarse output based on features extracted by E_p and E_f, a decoder including seven layers (one attention layer and six upsampling convolution layers) has been used. In common encoder-decoder approaches, the decoder receives features directly from the encoder but in our proposed method, the decoder receives two types of features including low-level features extracted by large receptive fields that may lack detailed information (i.e., the output of E_p), and high-level detailed features with a small receptive field (i.e., the output of E_f). Thus, we use a CSAB as the first layer of the decoder to discriminate the more effective features from others by assigning more weights.

Channel and Spatial Attention Block (CSAB): According to the outputs of E_p and E_f, the input to the attention block contains two types of features: a) large receptive field that may lack detailed information and b) output of E_f, i.e., highlevel detailed features with small receptive fields. We adopt the concatenating operation to aggregate these two types of features. On the other hand, we may achieve redundant information about multi-level contextual information and this situation will not be efficient for our goals. Thereby, as shown in Figure 4.3, we adopt a specific attention block called channel and spatial attention block (CSAB) [19] to assign more weight to important features [48] and alleviate the interference of redundant features by channel and spatial attention. Hence, attention block composes of two main attentions which we will introduce. Convolution operation leads to local contextual information. Discriminative features representation is essential for inpainting. We leverage the attention mechanism to fulfill this desire. The channel attention emphasizes interdependent feature maps by exploiting the dependencies between channels. Meanwhile, the spatial attention encodes a wide range of contextual dependency within each channel, thereby improving the overall representation capability by gaining mutual for similar features.

4.3.2 Refinement module

The coarse module's output $\left(\hat{I}_c\right)$ consists of face coarse structure including placed face elements, stated face pose, specified color skin, etc., suffering from fine details. To add more details to the $\hat{I}_c$, we propose a GAN-based refinement module.

4.3.2.1 Refinement generator

Inspired by the U-Net architecture [28] and the refinement network proposed by [29], we proposed a more effective architecture by replacing some DL blocks with SPD and selfattention (SA) blocks which receive the concatenation of $\hat{I}_c$ and I_{edge} as its input. We have adopted SPD blocks with four dilation rates in the middle of our architecture to extract features with various receptive fields from input images and then used SA blocks between middle layers. SA benefits from the concept of self-similarity, which is useful for reclaiming the reconstructed pattern based on the remaining ground truth in a masked image. As mentioned before, the duty of this stage is that it should improve fine details of images, hence, we use reconstruction and perceptual losses to adjust the fine details.

Table 4.1: Quantitative results over EtoF dataset for EtoFGAN and other compared methods (PIC, LaFIn, EC). The best result of each column is boldfaced. ↑ indicates that the higher the number the better is the model and ↓ indicates the lower the number the better is the model.

Method	FID ↓	SSIM ↑	PSNR ↑	TV ↓	ℓ_1 Loss ↓
PIC	57.02	0.41	11.19	8.50	50.37
LaFIn	63.16	0.47	13.18	6.89	**40.94**
EC	70.63	0.42	12.67	5.27	121.08
E2F-GAN (ours)	**46.39**	**0.51**	**13.66**	**0.02**	41.54

4.3.2.2 Discriminator

To inpaint and generate more realistic high-quality faces, both coarse and refinement modules have been designed based on GAN structures, thus, two discriminators have the responsibility of evaluating the output of C and F. The coarse module's discriminator (D_1) receives $\hat{I}_c$ and consequently the refinement module's discriminator (D_2) has been fed by $\hat{I}_f$. We have combined the concept of SN-GAN [30] and PatchGAN [31] for these discriminators to distinguish real or fake images. Besides this combination, we have used the hinge adversarial loss function for our discriminators. These combinations and loss functions help us to train our discriminators faster and more stable, distinguishing real or fake images efficiently.

4.3.2.3 E2F-GAN end-to-end training

The E2F-GAN model is trained in a supervised and endto-end manner. We have defined four groups of loss functions [41] for various parts of our proposed method to achieve considerable results. To train C, we have utilized four specific loss functions including reconstruction loss, perceptual loss, style loss, and adversarial loss; and just reconstruction and perceptual losses have been used for training F. With the aim of having an end-to-end training process, we define the total loss $\mathcal{L}$ which consists of four groups of component losses as below:

$$\begin{aligned}\mathcal{L} = {} & \lambda_{\text{rec}} \left(\mathcal{L}_{\text{rec}}^{c} + \mathcal{L}_{\text{rec}}^{f}\right) + \lambda_{\text{perc}} \left(\mathcal{L}_{\text{perc}}^{c} + \mathcal{L}_{\text{prec}}^{f}\right) \\ & + \lambda_{\text{style}} \mathcal{L}_{\text{style}}^{c} + \left(\lambda_{\text{adv}}^{g} \mathcal{L}_{\text{adv}}^{g} + \lambda_{\text{adv}}^{d} \mathcal{L}_{\text{adv}}^{d}\right) \qquad (1)\end{aligned}$$

In the following, the formulation of the used losses and the notion behind each loss is described. The reconstruction loss $(\mathcal{L}_{\text{rec}})$ or per-pixel loss measures the pixel-wise difference between the synthesized image and the ground truth image. This loss is essential for maintaining texture information. It is calculated as the L1-norm between $\hat{I}_z$ and the corresponding ground truth $I_g.\mathcal{L}_{rec}$ is defined as follows:

$$\mathcal{L}_{rec}^{z} = \frac{1}{H \times W \times N} \sum_{i=1}^{H} \sum_{j=1}^{W} \left| \hat{I}_z(i,j) - I_g(i,j) \right| \tag{2}$$

where z is replaced with c or f depending on the $\mathcal{L}_{\text{rec}}$ is used for C or F, respectively.

It is worth to mention that, an element-wise loss cannot consider high-level semantics. Accordingly, recent research [19], [21], [22] suggests using perceptual distances based on a pre-trained network, VGG19 which was trained on the ImageNet. The perceptual loss $\left(\mathcal{L}_{\text{perc}}\right)$ measures the difference between features extracted from the various layers of the VGG19 network for $\hat{I}$ and its corresponding ground truth.

$$\mathcal{L}_{\text{perc}}^{z} = \sum_{l=1}^{L} \frac{|\hat{\varphi}_z - \varphi_g|}{N_l \times H_l \times W_l} \tag{3}$$

where $\hat{\varphi}_z$ and φ_g are extracted features from $\hat{I}$ and I_g respectively, and z is replaced with c or f depending on the $\mathcal{L}_{\text{perc}}$ is used for coarse or refinement, respectively. We extract features from L layers of the pre-trained network. relu1_1, relu2_1,relu3_1, relu4_1, and relu5_1 of the VGG19 utilized to calculate $\mathcal{L}_{\text{perc}}$ as well as $\mathcal{L}_{\text{style}}$ described below.

In order to provide richer texture, we also employ style loss $\left(\mathcal{L}_{\text{style}}\right)$. In style loss, a Gram matrix calculates the correlation between channels in a feature map. The style loss then calculates on the features map produced by the pre-trained VGG19 network.

$$\mathcal{L}_{\text{style}}^{c} = \sum_{l=1}^{L} \frac{1}{N_l \times N_l} \left\| \frac{G_l\left(\hat{I}_c\right) - G_l\left(I_g\right)}{H_l \times W_l \times N_l} \right\|_1 \tag{4}$$

where $G_l\left(.\right) = \varphi_l\left(.\right)^{T} \varphi_l\left(.\right)$ stands for the Gram matrix corresponding to $\varphi(.)$.

For generative adversarial learning, our discriminators are trained to distinguish between generated images and ground truth images. on the other hand, the generators strive to cheat the discriminators by hardening that classification. We employ hinge loss to train our model, L_{adv}^{d} and L_{adv}^{d} computed as follows:

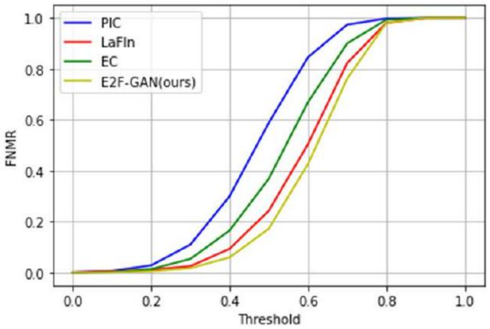


Figure 4.4: FNMR curve for our proposed method (E2F-GAN) and other compared methods (PIC, LaFIn, EC).

$$\mathcal{L}_{adv}^{g} = - E\left[D_1\left(C\left(I_m, \mathrm{I}_{\text{edge}}\right)\right)\right] - E\left[D_2\left(F\left(\hat{I}_c, I_{\text{edge}}\right)\right)\right] \tag{5}$$

$$\begin{aligned}\mathcal{L}_{adv}^{d} =& E\left[\mathrm{Relu}\left(1 + D_1\left(C\left(I_m, \mathrm{I}_{\text{edge}}\right)\right)\right)\right] \\ &+ E\left[\mathrm{Relu}\left(1 + D_2\left(F\left(\hat{I}_c, \mathrm{I}_{\text{edge}}\right)\right)\right)\right] \\ &+ E\left[\mathrm{Relu}\left(1 - D_1\left(I_g\right)\right)\right] + E\left[\mathrm{Relu}\left(1 - D_2\left(I_g\right)\right)\right]\end{aligned} \tag{6}$$

As mentioned before, we combine the used loss functions with appropriate weights as follows: $\lambda_{\text{rec}} = 1, \lambda_{\text{perc}} = 0.1, \lambda_{\text{style}} = 250, \lambda_{adv}^{g} = 0.1, \lambda_{adv}^{d} = 1$.

4.4 Experiments and discussion

In this section, we evaluate the E2F-GAN performance on a new generated face dataset (E2Fdb) based on CelebA-HQ. We compared our results with three other methods called EdgeConnect (EC) [15], Pluralistic Image Completion (PIC) [22], LaFIn [21]. To have fair comparison, the three methods have been trained using the E2Fdb. For quantitatively measuring the performance difference among the methods, we employ several statistical metrics. Moreover, to measure the amount of preservation of demographic and biometric features, we calculate False Non-Match Rate between original and inpainted faces. Using a competitive face biometric matcher [49] based on ArcFace [36].

4.4.1 Datasets

We conduct all experiments on our generated dataset called E2Fdb (available on project's GitHub page) extracted from the well-known CelebA-HQ

Table 4.2: The effect of various parts of E2F-GAN on final results.

Edge Predictor	Attention Block	Refinement module	FID ↓	SSIM ↑	PSNR ↑	TV ↓	ℓ_1 Loss ↓
x	✓	✓	64.89	0.34	12.18	12.91	45.99
✓	x	✓	50.12	0.48	13.24	6.33	43.97
✓	✓	x	75.21	0.46	13.22	5.49	42.13
✓	✓	✓	**46.39**	**0.52**	**13.66**	**0.02**	**41.54**

dataset [32], [49]. To extract the periocular region from each face image, the images are reshaped to size 256×256 and then by utilizing a landmark detector [27], eyes are detected, similar to [50]. Doing this, M and I_m are produced for each image. Moreover, we removed misleading samples including those eyes covered by sunglasses or faces that have more than 45 degrees in one angle (roll, pitch, yaw) leading to hiding one of the eyes by using WHENet [33] algorithms. Finally, the total number of samples is 24,554 among which 22,879 will be used for the training process and the rest, which is 1,685 images, for the test.

4.4.2 Evaluation metrics

We evaluate the image inpainting performance of the proposed model using quantitative and qualitative comparisons.

For quantitative comparison, two types of metrics called statistical and identity metrics have been measured. In the following, we describe each category and its corresponding metrics briefly.

4.4.2.1 Statistical metrics

We use five statistical metrics: ℓ_1 loss, Peak Signal to Noise Ratio (PSNR), Structural Similarity (SSIM) [34], Frenchet Inception Distance (FID) [35], and Total Variation (TV). Notably, the ℓ_1 loss shows the model's reconstruction ability for images. PSNR measures the visibility of errors between the ground truth I_g and image inpainting $\hat{I}$ to evaluate the image quality. SSIM aims at estimating the perceptual changes in the structural information, which shows human's subjective feelings more accurately than PSNR. FID is a widely used metric in the image generation field to measure the visual quality. TV assists to measure the amount of noise in the image by calculating the sum of the absolute differences for neighboring pixels.

4.4.2.2 Identity metrics

To measure the amount of preservation of demographic and biometrics characteristics after completing inpainting process, we calculate the False Non-Match Rate (FNMR). FNMR is the rate at which a biometric algorithm miss-categorizes two captures from the same individual as being from different

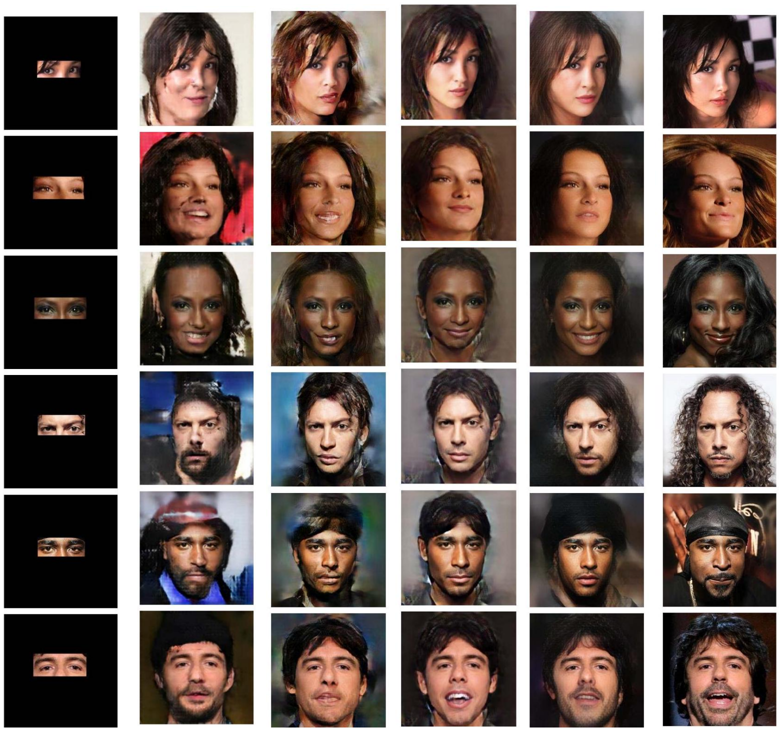


Figure 4.5: Quality comparison among PIC, EC, LaFIn, and our proposed method.

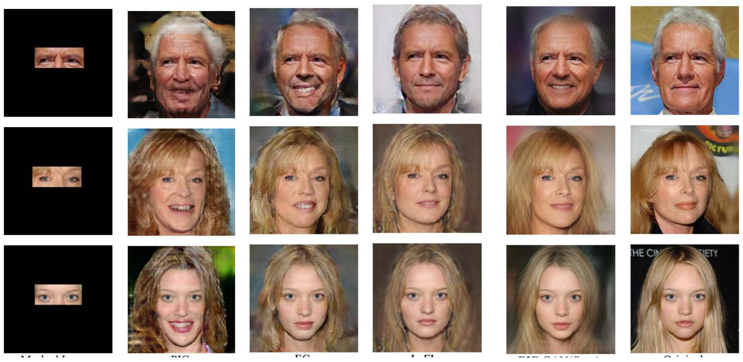


Figure 4.6: Illustration of image reconstruction at different age of the subject among our proposed method, PIC, EC, LaFIn, and original image.

individuals. Here, we assumed that $\hat{I}$ and I are two faces for the same individual and using ArcFace [36], we calculate the corresponding embedding vectors for each face, and finally calculate the cosine similarity between each pair. Finally, the FNMR for different thresholds is shown.

4.4.3 Comparison with existing work

By using the above-mentioned metrics and presenting some outputs, the results of our proposed method have been qualitatively and quantitatively compared against three state-ofthe-art approaches, named PIC, EC, and LaFIn. We trained the three methods over our generated dataset (i.e., E2Fdb) according to the best configurations of each method mentioned in the corresponding paper. In the following subsections, we present the results.

4.4.3.1 Quantitative comparisons

The results of the statistical metrics calculated on the validation set of E2Fdb including 1,675 samples are reported in Table 4.1. As can be seen from the numbers in Table 4.1, E2F-GAN is superior over PIC, LaFIn, and EC in most metrics, except for the ℓ_1 loss for which LaFIn works slightly better. Overall, our E2F-GAN outperforms the others by large margins in terms of FID, SSIM, PSNR, and TV metrics. More specifically, our large margins in FID and TV metrics demonstrate that our method can inpaint the masked image with much higher quality compared to other methods. Moreover, FNMR has been measured for E2F-GAN and other three compared methods as shown in Figure 4.4. For different thresholds, E2F-GAN has lower false non-match rate which shows the ability of our algorithm extracting identity information from the periocular region and transferring it to the reconstructed face. Notably, since the PIC method generates different outputs for a specific input, we executed this method five times and the best results have been reported.

4.4.3.2 Qualitative comparisons

Fig. 5 shows some faces generated by our model, PIC, EC, and LaFIn. Our model is able to generate high quality results and a large fraction of face structures including face shape, nose, mouth, forehead, etc. are appropriately placed with a plausible size. Moreover, to compare the quality of results in terms of gender and skin color, we present different faces in Figures. 4.5 and 4.6. As it can be observed, the quality of PIC and EC is really low compared to our and LaFIn results. Therefore, although like EC we used edge predictor in our scheme, there is a large margin between our outcomes.

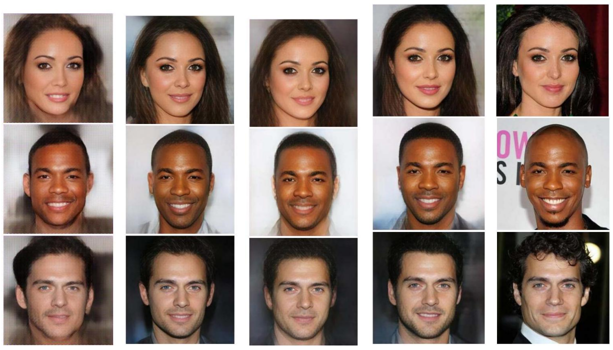


Figure 4.7: Illustrative comparison of the effect of various parts on final output.

Table 4.3: The effect of landmark or edge guidance on final results.

	FID ↓	SSIM ↑	PSNR ↑	TV ↓	ℓ_1 Loss ↓	Landmark Loss ↓
Landmark Guidance	48.56	0.44	**13.33**	7.31	42.48	**15.37**
Edge Guidance	**46.39**	**51.9**	13.66	**0.02**	**41.54**	15.42

Additionally, with aim of further investigation of the models' outputs regarding age and gender prediction based on the periocular region, we presented some challenging examples in Figure 4.6. That figure shows three faces including two elders (a man and a woman) and a young woman. As seen in those examples, E2F-GAN can assess the age based on periocular region and reconstruct the face with a reasonable quality.

4.5 Ablation study

In this section, firstly we qualitatively and quantitatively analyze the effect of three main components of our proposed model including the edge predictor, the refinement module, and the attention block. Table 4.2 and Figure 4.8 report statistical and identity metrics indicating the degree of effectiveness each of the three components in the performance of E2F-GAN. Specifically, the refinement network is the most conspicuous one which benefits the model by providing conformity and consistency among face components and skin texture around the eyes, such as wrinkles and skin color. The edge guidance contributes to ensuring that the structure of the face is well-preserved (see Figure 4.7). Visually, the effectiveness of the attention block may not seem tangible. However, the quantitative results demonstrate

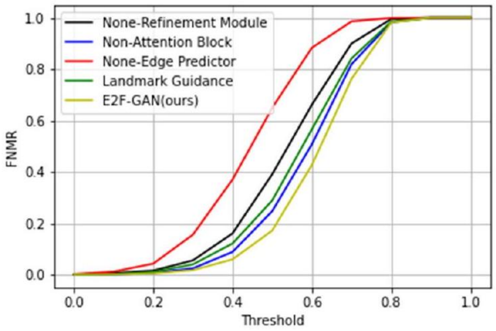


Figure 4.8: The impact of various parts of E2F-GAN on FNMR ratio.

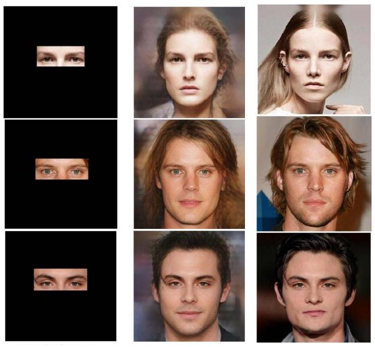


Figure 4.9: Illustration of gender preserve in our proposed method.

the advantages of attention block. We also compared the effect of edge and landmark predictors. As shown in Table 4.3, the edge guidance provides better values in most quantitative metrics specially for SSIM metric.

Finally, Figure 4.9 shows a few challenging examples for preserving the gender of the person based on the periocular region. Our observations show that E2F-GAN can preserve the gender of subjects with a high accuracy.

4.6 Conclusion

The aim of this paper is a particular case of face inpainting where we try to reconstruct the face based on just using the periocular region. To do this, we presented E2F-GAN, a GAN-based architecture that benefits from the advantages of coarse-to-fine, coarse-and-fine, and structural guidance-based architectures for face inpainting. It includes three main modules for extracting face's edges (edge predictor), coarse prediction of face elements (coarse generator) and refining the coarse predicted image (refinement generator). We analyzed E2F-GAN and compared it with other well-known face inpainting methods to measure the efficiency and quality performance. For doing this, we modified a widely used face inpainting dataset called CelebA-HQ such that the whole face except the periocular region is masked and used for E2F-GAN input, calling the resulting dataset E2Fdb. Our proposed inpainting algorithm E2F-GAN and the used dataset E2Fdb are both available in the project GitHub.

Several qualitative and quantitative metrics have been measured during our experiments to show the performance of E2F-GAN in terms of preserving identity and non-identity features of each face after inpainting. Experimental results show that our method outperforms previous learning-based face inpainting methods and E2F-GAN can generate realistic and semantically plausible images.

Future work includes analyzing biometric quality aspects of the resulting faces using recent objective measures [51], [52]; analyzing [49] and reducing [53] undesired biases in the face generation process; and combining multiple face generation approaches for better outputs [48].

Acknowledgment

(Ahmad Hassanpour, Amir Etefaghi Daryani, and Mahdieh Mirmahdi contributed equally to this work.)

References

[1] A. Criminisi, P. Pérez, and K. Toyama, ”Region filling and object removal by exemplar-based image inpainting,” IEEE Trans. Image Process., vol. 13, no. 9, pp. 1200-1212, Sep. 2004.

[2] O. Elharrouss, N. Almaadeed, S. Al-Maadeed, and Y. Akbari, ”Image inpainting: A review,” Neural Process. Lett., vol. 51, no. 2, pp. 2007-2028, 2019 .

[3] D. Pathak, P. Krähenbühl, J. Donahue, T. Darrell, and A. A. Efros, ”Context encoders: Feature learning by inpainting,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2016, pp. 2536-2544.

[4] H. Yamauchi, J. Haber, and H.-P. Seidel, ”Image restoration using multiresolution texture synthesis and image inpainting,” in Proc. Comput. Graph. Int., Jul. 2003, pp. 120-125.

[5] X. Zhu, Y. Qian, X. Zhao, B. Sun, and Y. Sun, ”A deep learning approach to patch-based image inpainting forensics,” Signal Process., Image Commun., vol. 67, pp. 90-99, Sep. 2018.

[03] https://github.com/amiretefaghi/E2F-GAN

[6] O. Sidorov and J. Y. Hardeberg, ”Deep hyperspectral prior: Single-image denoising, inpainting, super-resolution,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW), Oct. 2019, pp. 3844-3851.

[7] Y. Zeng, J. Fu, H. Chao, and B. Guo, ”Learning pyramid-context encoder network for high-quality image inpainting,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 1486-1494.

[8] H. Liu, B. Jiang, Y. Xiao, and C. Yang, ”Coherent semantic attention for image inpainting,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 4170-4179.

[9] Z. Guo, Z. Chen, T. Yu, J. Chen, and S. Liu, ”Progressive image inpainting with full-resolution residual network,” in Proc. 27th ACM Int. Conf. Multimedia, Oct. 2019, pp. 2496-2504.

[10] J. Li, N. Wang, L. Zhang, B. Du, and D. Tao, ”Recurrent feature reasoning for image inpainting,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 7760-7768.

[11] H. Zhang, Z. Hu, C. Luo, W. Zuo, and M. Wang, ”Semantic image inpainting with progressive generative networks,” in Proc. 26th ACM Int. Conf. Multimedia, Oct. 2018, pp. 1939-1947.

[12] H. Liu, B. Jiang, Y. Xiao, and C. Yang, ”Coherent semantic attention for image inpainting,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019 , pp. 4170-4179.

[13] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, ”Generative image inpainting with contextual attention,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 5505-5514.

[14] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. Huang, ”Free-form image inpainting with gated convolution,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 4471-4480.

[15] K. Nazeri, E. Ng, T. Joseph, F. Qureshi, and M. Ebrahimi, ”EdgeConnect: Structure guided image inpainting using edge prediction,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW), Oct. 2019, pp. 3265-3274.

[16] M. Chen and Z. Liu, ”EDBGAN: Image inpainting via an edge-aware dual branch generative adversarial network,” IEEE Signal Process. Lett., vol. 28, pp. 842-846, 2021.

[17] E. Bobrov, A. Georgievskaya, K. Kiselev, A. Sevastopolsky, A. Zhavoronkov, S. Gurov, K. Rudakov, M. D. P. B. Tobar, S. Jaspers, and S. Clemann, ”PhotoAgeClock: Deep learning algorithms for development of non-invasive visual biomarkers of aging,” Aging (Albany NY), vol. 10, no. 11, p. 3249, 2018.

[18] C. Rathgeb and C. Busch, Handbook Iris and Periocular Biometric Recognition. London, U.K.: Institution of Engineering and Technology (IET), 2017.

[19] M. Chen, Z. Liu, L. Ye, and Y. Wang, ”Attentional coarse-and-fine generative adversarial networks for image inpainting,” Neurocomputing, vol. 405, pp. $259 - 269$, Sep. 2020.

[20] L. Song, J. Cao, L. Song, Y. Hu, and R. He, ”Geometry-aware face completion and editing,” in Proc. AAAI Conf. Artif. Intell., vol. 33, no. 1, Jul. 2019, pp. 2506-2513.

[21] Y. Yang and X. Guo, ”Generative landmark guided face inpainting,” in Proc. Pattern Recognit. Comput. Vis., Oct. 2020, pp. 14-26.

[22] C. Zheng, T.-J. Cham, and J. Cai, ”Pluralistic image completion,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. $1438 - 1447$.

[23] L. Zhao, Q. Mo, S. Lin, Z. Wang, Z. Zuo, H. Chen, W. Xing, and D. Lu, ”UCTGAN: Diverse image inpainting based on unsupervised cross-space translation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 5741-5750.

[24] N. U. Din, K. Javed, S. Bae, and J. Yi, ”A novel GAN-based network for unmasking of masked face,” IEEE Access, vol. 8, pp. 44276-44287, 2020.

[25] Y. Zeng, J. Fu, H. Chao, and B. Guo, ”Learning pyramid-context encoder network for high-quality image inpainting,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 1486-1494.

[26] G. Liu, F. A. Reda, K. J. Shih, T. C. Wang, A. Tao, and B. Catanzaro, ”Image inpainting for irregular holes using partial convolutions,” in Proc. Eur. Conf. Comput. Vis., Sep. 2018, pp. 85-100.

[27] A. Bulat and G. Tzimiropoulos, ”How far are we from solving the 2D & 3D face alignment problem? (and a dataset of 230,000 3D facial landmarks),” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 1021-1030.

[28] O. Ronneberger, P. Fischer, and T. Brox, ”U-Net: Convolutional networks for biomedical image segmentation,” in Proc. Med. Image Comput. Comput.-Assist. Intervent, 2015, pp. 234-241.

[29] C. T. Li, W. C. Siu, Z. S. Liu, L. W. Wang, and D. P. K. Lun, ”DeepGIN: Deep generative inpainting network for extreme image inpainting,” in Proc. Comput. Vis. ECCV 2020 workshops, Glasgow, U.K., 2020, pp. $5 - 22$.

[30] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, ”Spectral normalization for generative adversarial networks,” in Proc. ICLR, Vancouver, BC, Canada, 2018, pp. 1-29.

[31] K. Lata, M. Dave, and K. N. Nishanth, ”Image-to-image translation using generative adversarial network,” in Proc. 3rd Int. Conf. Electron., Commun. Aerosp. Technol. (ICECA), Jun. 2019, pp. 186-189.

[32] T. Karras, T. Aila, S. Laine, and J. Lehtinen, ”Progressive growing of GANs for improved quality, stability and variation,” in Proc. ICLR, Vancouver, BC, Canada, 2018.

[33] Y. Zhou and J. Gregson, ”WHENet: Real-time fine-grained estimation for wide range head pose,” in Proc. 31st Brit. Mach. Vis. Conf. (BMVC), 2020.

[34] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, ”Image quality assessment: From error visibility to structural similarity,” IEEE Trans. Image Process., vol. 13, no. 4, pp. 600-612, Apr. 2004.

[35] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, ”GANs trained by a two time-scale update rule converge to a local nash equilibrium,” in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 6626-6637.

[36] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, ”ArcFace: Additive angular margin loss for deep face recognition,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 4690-4699.

[37] F. Alonso-Fernandez, K. Hernandez-Diaz, S. Ramis, F. J. Perales, and J. Bigun, ”Facial masks and soft-biometrics: Leveraging face recognition CNNs for age and gender prediction on mobile ocular images,” IET Biometrics, vol. 10, no. 5, pp. 562-580, Sep. 2021.

[38] F. Alonso-Fernandez, K. B. Raja, R. Raghavendra, C. Busch, J. Bigun, R. Vera-Rodriguez, and J. Fierrez, ”Cross-sensor periocular biometrics for partial face recognition in a global pandemic: Comparative benchmark and novel multialgorithmic approach,” 2019, arXiv:1902.08123.

[39] F. Alonso-Fernandez, R. A. Farrugia, J. Fierrez, and J. Bigun, ”Superresolution for selfie biometrics: Introduction and application to face and iris,” in Selfie Biometrics. Cham, Switzerland: Springer, 2019, pp. $105-128$.

[40] R. Tolosana, S. Romero-Tapiador, R. Vera-Rodriguez, E. Gonzalez-Sosa, and J. Fierrez, ”DeepFakes detection across generations: Analysis of facial regions, fusion, and performance evaluation,” Eng. Appl. Artif. Intell., vol. 110, Apr. 2022, Art. no. 104673.

[41] A. Morales, J. Fierrez, A. Acien, R. Tolosana, and I. Serna, ”SetMargin loss applied to deep keystroke biometrics with circle packing interpretation,” Pattern Recognit., vol. 122, Feb. 2022, Art. no. 108283.

[42] P. Tome, J. Fierrez, R. Vera-Rodriguez, and D. Ramos, ”Identification using face regions: Application and assessment in forensic scenarios,” Forensic Sci. Int., vol. 233, nos. 1-3, pp. 75-83, Dec. 2013

[43] E. Gonzalez-Sosa, J. Fierrez, R. Vera-Rodriguez, and F. Alonso-Fernandez, ”Facial soft biometrics for recognition in the wild: Recent works, annotation, and COTS evaluation,” IEEE Trans. Inf. Forensics Security, vol. 13, no. 8 , pp. 2001-2014, Aug. 2018

[44] P. Tome, J. Fierrez, R. Vera-Rodriguez, and J. Ortega-Garcia, ”Combination of face regions in forensic scenarios,” J. Forensic Sci., vol. 60, no. 4, pp. 1046-1051, Jul. 2015.

[45] R. Ranjan, S. Sankaranarayanan, A. Bansal, N. Bodla, J. C. Chen, V. M. Patel, C. D. Castillo, and R. Chellappa, ”Deep learning for understanding

faces: Machines May be just as good, or better, than humans," IEEE Signal Process. Mag., vol. 35, no. 1, pp. 66-83, Jan. 2018.

[46] P. Tome, R. Vera-Rodriguez, J. Fierrez, and J. Ortega-Garcia, "Facial soft biometric features for forensic face recognition," Forensic Sci. Int., vol. 257 , pp. $171 - 284$, Dec. 2015

[47] J. Hernandez-Ortega, J. Fierrez, I. Serna, and A. Morales, "FaceQgen: Semi-supervised deep learning for face image quality assessment," in Proc. 16th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), Dec. 2021, pp. 1-8.

[48] J. Fierrez, A. Morales, R. Vera-Rodriguez, and D. Camacho, "Multiple classifiers in biometrics. Part 2: Trends and challenges," Inf. Fusion, vol. 44, pp. 103-112, Nov. 2018.

[49] I. Serna, A. Morales, J. Fierrez, and N. Obradovich, "Sensitive loss: Improving accuracy and fairness of face representations with discrimination-aware deep learning," Artif. Intell., vol. 305, Apr. 2022, Art. no. 103682.

[50] R. Daza, D. DeAlcala, A. Morales, R. Tolosana, R. Cobos, and J. Fierrez, "ALEBk: Feasibility study of attention level estimation via blink detection applied to e-learning," in Proc. AAAI Workshop Artif. Intell. Educ. (AI4EDU), 2022

[51] J. Hernandez-Ortega, J. Galbally, J. Fierrez, R. Haraksim, and L. Beslay, "FaceQNet: Quality assessment for face recognition based on deep learning," in Proc. Int. Conf. Biometrics (ICB), Jun. 2019, pp. 1-8.

[52] J. Hernandez-Ortega, J. Fierrez, L. F. Gomez, A. Morales, J. L. Gonzalez-de-Suso, and F. Zamora-Martinez, "FaceQvec: Vector quality assessment for face biometrics based on ISO compliance," in Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. Workshops (WACVW), Jan. 2022, pp. 84-92

[53] A. Morales, J. Fierrez, R. Vera-Rodriguez, and R. Tolosana, "SensitiveNets: Learning agnostic representations with application to face images," IEEE Trans. Pattern Anal. Mach. Intell., vol. 43, no. 6, pp. 2158-2164, Jun. 2021.

Chapter 5

E2F-Net: Eyes-to-Face Inpainting via StyleGAN Latent Space

Ahmad Hassanpour, Fatemeh Jamalbafrani, Bian Yang, Kiran Raja, Raymond Veldhuis, Julian Fierrez

Abstract

Face inpainting, the technique of restoring missing or damaged regions in facial images, is pivotal for applications like face recognition in occluded scenarios and image analysis with poor-quality captures. This process not only needs to produce realistic visuals but also preserve individual identity characteristics. The aim of this paper is to inpaint a face given periocular region (eyes-to-face) through a proposed new Generative Adversarial Network (GAN)-based model called Eyes-to-Face Network (E2F-Net). The proposed approach extracts identity and nonidentity features from the periocular region using two dedicated encoders have been used. The extracted features are then mapped to the latent space of a pre-trained StyleGAN generator to benefit from its state-of-the-art performance and its rich, diverse and expressive latent space without any additional training. We further improve the StyleGAN's output to find the optimal code in the latent space using a new optimization for GAN inversion technique. Our E2F-Net requires a minimum training process reducing the computational complexity as a secondary benefit. Through extensive experiments, we show that our method successfully reconstructs the whole face with high quality, surpassing current techniques, despite significantly less training and supervision efforts. We have generated seven eyes-to-face datasets based on well-known public face datasets for training and verifying our proposed methods. The code and datasets are publicly available.

Keywords: Eyes-to-Face, Face Inpainting, Face Reconstruction, GAN Latent Space, StyleGAN.

5.1 Introduction

Face inpainting is the process of approximating the missing or masked face elements using the auxiliary data from around of the missing region. Thus,

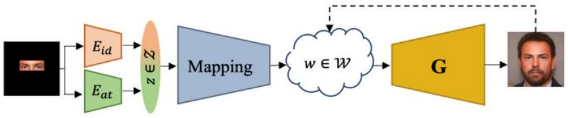


Figure 5.1: The proposed face reconstruction framework utilizes two encoders called identity (E_{id}) and attribute (E_{at}), to generate the latent code z. The latent code z is then mapped to the latent space $\mathcal{W}$ of a pre-trained generator shown by G. Finally, the output of G will be refined by finding the optimal point in $\mathcal{W}$ space using an optimizer (return arrow from G 's output to $\mathcal{W}$ space).

estimating those missing regions is vital in practice, particularly in face recognition under occlusions, and in general any image/video analysis application on low quality, uncontrolled, or in-the-wild acquisition conditions. Realistic approximation or inpainting despite being highly applicable, is known to be particularly hard task. This is mainly due to high photometric, geometric and kinematic complexities, and because the human face contains numerous independent, high dimensional characteristics that are not easy to approximate and also make it realistic for human perception [1]. Like other image inpainting tasks (e.g., scenes inpainting [3, 4], streets inpainting [2, 12]), some key requirements for face inpainting are:

- R1) the filled region in corrupted area should be semantically meaningful in relation to the face,
- R2) the original content (unmasked) and approximated content should be continuously assembled and consistent,
- R3) the inpainted image should be visually realistic and have high fidelity.

Reconstructing the corrupted/unavailable portions of a face such that the topological consistency between facial attributes are preserved (both identity and non-identity [2] attributes), is not a trivial task [6, 8]. One can however exploit that human faces share common geometrical and appearance distributions, which are then personalized for given subjects in specific conditions. General face geometry/appearance models have been used to ease face manipulation and completion for given subjects [2]. Notably, a specific facial representation deviating from or sampling the general model can be considered for the purpose of identity information completion due to the unique topology of different facial elements and their distinctive characteristics.

Among all facial elements, the eyes are one of the most expressive organs on the human face and contain discriminative features [39]. In this paper, we aim to reconstruct a face using the periocular region alone which we refer to as eyes-to-face inpainting (see Figure 5.1). Therefore, in addition to the above-mentioned challenges (i.e., R1-R3), another set of criteria for eyes-to-face inpainting are as follows:

- R4) The topological structure of the face should be reconstructed in such a way that all elements are placed in the proper position both semantically and continuously. For doing this, it is essential to predict the shape of the face precisely and place the face's elements (e.g., nose, chin, mouth) proportionally within the predicted frame. Moreover, the head pose should be aligned and integrated with other elements based on the appearance of eye.

- R5) The usefulness of skin around the eyes for determining demographic features (e.g., age, gender) has been shown in previous research [43]. The proposed inpainting model should therefore be able to estimate demographic characteristics using the color, texture, and size of the eyes and brows, then inpaint other facial attributes using the predicted attributes.

- R6) The proposed solution should preserve the identity-related features present in the eyes region when reconstructing the whole face.

It is also important to note that the performance of suggested solutions can be directly impacted by the image's masking, and it is obvious that bigger masks make it harder to achieve the referenced requirements (i.e., R1-R6).

Generally, synthetic and natural masks are considered in face inpainting scenarios, which can be classified into two categories called free- (irregular) and fixed-form (regular) masks. In widely used free-form masks [23, 24], there are irregular shapes randomly placed on the images (Figure 5.2(a)(b)), useful for inpainting irregular scratches. Instead, in the fixed-form masks, regular shapes cover some portions of the images which are placed on the images randomly or purposefully (Figure 5,2(c)-(e)) [21, 22, 25].

Recently, learning-based techniques such as deep convolutional neural networks (CNNs) and generative adversarial networks (GANs) have been widely used for a variety of image inpainting tasks, such as eliminating objects [7, 8], noises [9], texts [10], and masks [11]. Usually, the proposed CNN-based methods are classified into three categories including coarse-to-fine, coarse-andfine, and structural guidance-based methods. Coarse-to-fine based methods [13, 14] exploit two-stage architectures to complete content

[02] Here, identity features are facial features that can be used to verify the identity of a person using his face including demographic characteristics (e.g., age, gender, color skin), and non-identity features are characteristics like head pose, and facial expression.

formation and texture refinement. A two-stage design produces an intermediate coarse image after reconstructing structures in the first step, then feeds it to the second stage for texture improvement. The second category called coarse-and-fine [15, 16] consists of two parallel branches, i.e., coarse and fine, that extract coarse and fine information simultaneously and fill the missed regions using the extracted information. The final group of approaches, known as structural guidance-based methods, employs an assistance algorithm to provide additional information, such as edges [17, 18], contours [19], or landmarks [6], for the proposed inpainting method.

The rest of the paper is organized as follows. Recent works in face inpainting, latent space embedding, and GAN inversion are reviewed in Section 2. Limitations of related works and our contributions have been discussed in section 3. A detained description of the proposed method is provided in Section 4. The experimental results and ablation study are reported in Sections 5 and 6. Finally, sections 7 and 8 present discussion and conclusion.

5.2 Background and related work

In this section, we briefly review the most relevant research on face inpainting, latent space embedding, and GAN inversion in the following subsections.

5.2.1 Face inpainting

There are a few works that particularly attempted to reconstruct a face using the periocular region. Luo et al. [27] proposed a three-step solution called EyesGAN which includes two GANs to predict other parts of a face using the eyes region. They proposed a self-attention mechanism to extract informative and attention feature maps from convolution layers. Unfortunately, it is difficult to compare our results to EyesGAN due to its unavailability. Hassanpour et al. [18] proposed a GAN-based coarse-to-fine method called E2F-GAN such that the coarse module benefits from the coarse-and-fine architecture. They used edges as the guidance information for the designed coarse-to-fine network.

With the aim of face inpainting by placing randomly regular or irregular masks, several methods have been proposed recently. In order to produce more realistic images, Chen et al. [28] presented a generative-based coarse-to-fine structure that takes advantage of an attention layer to capture lengthy dependencies between features. Free-form masks are inpainted using a coarse-to-fine structure proposed by Yu et al. [13]. A novel attention layer in a coarse-to-fine design was suggested by Liu et al. [30] in the same context. Wang et al. [24] proposed a two-stage face inpainting method to detect the corrupted regions and then improve inpainting results using a top-down refinement network.

A few works proposed guidance-based techniques. An edge generator is used by Nazari et al. [17] to reconstruct the edges before feeding the corrupted image and predicted edges to the image inpainting network. In order to extract features and recover the structures and textures of missing regions, Chen and Liu [19] employ a dual-branch network with texture and edge branches. Some works estimate facial landmarks to assist the main inpainting network[6]. A unique output per each input is generated by the methods indicated above. In contrast, some other approaches inpaint the corrupted regions differently for each specific input. A dual pipeline based on Variational Auto-Encoders (VAEs) was proposed by Zheng et al. [26], with a reconstructive path that uses the ground truth to learn the prior distribution of missing regions and a generative path for which the conditional prior is connected to the distribution learned in the reconstructive path. Zhao et al. [32] have suggested a GAN-based unsupervised conditional framework for different image inpainting that can learn conditional completion distributions.

5.2.2 Latent space embedding

With the rapid evolution of GANs, many works have tried to understand and control their latent space for various image editing tasks [33]. Choosing which latent space to embed an image into a GAN image generator is a crucial design decision for editing flexibility and output quality. One of the most successful approaches for generating this embedding was described in the framework of StyleGAN, which has been followed extensively in the recent past $[34, 35]$. By using an 8-layer multilayer perceptron (MLP) to create a nonlinear mapping network M, StyleGAN [36] transforms a native $z \in Z$ to a style vector $w \in \mathcal{W}$. The $\mathcal{W}$ space is the name given to this intermediate latent space. The $\mathcal{W}$ space of StyleGAN contains more disentangled characteristics than the Z space does because of the mapping network M.

5.2.3 GAN inversion

GAN inversion tries to invert a given image back into a pretrained GAN model's latent space. The generator can then accurately rebuild the image from the inverted code. Learning-based, optimization-based, and hybrid methods are the three major strategies for GAN inversion with the purpose of projecting images into the latent space. Learning-based GAN inversion [39] typically involves training an encoding neural network $E\left(x; \theta_E\right)$ to map an input image, x, into the latent code z :

$$\theta_E^* = \arg_{\theta_E} \min \sum_n \mathcal{L}\left(G\left(E\left(x_n; \theta_E\right)\right), x_n\right) \tag{1}$$

where x_n denotes the n-th image in the training datase E_{Fand} $z = E\left(x; \theta_E^*\right)$. The objective in (1) is reminiscent of an autoencoder pipeline, with an encoder E and a decoder G. Throughout the training, the decoder G remains fixed. Furthermore, improving the latent vector is generally used to reconstruct a target image by optimization-based GAN inversion approaches [40].

$$\mathbf{z}^* = \arg_z \min \mathcal{L}\left(x, G\left(z; \theta_G\right)\right) \tag{2}$$

where x is a target image and G is a GAN generator parameterized by θ_G. The hybrid methods [41] exploit the advantages of both previously described approaches adjusting both the Encoder θ_E^* and the specific location in the latent space z^*.

5.3 Limitations of related works and our contributions

The existing face inpainting works use different strategies (e.g., coarse-to-fine, coarse-and-fine, guidance information) to address R1-R3. The coarse-to-fine structure has two limitations. First, the coarse result has to be reasonably accurate for an effective refinement, and second, the cascaded dilated convolutions smooth the details of features, resulting in blurry inpainting results [28]. Although structural information about the target image may assist and increase the performance of the inpainting generator in some cases, estimating that information can also slow the inference speed, increase the computational cost, and introduce the necessity of handcrafting auxiliary information (e.g., edge, contour, or landmarks) for different applications when guidance-based methods are used. Moreover, unlike approaches used in [26, 32], an eyes-to-face approach should generate a unique output for each input even after several executions to fulfill the requirements **R5** and **R6**.

We address these requirements and limitations by proposing a novel method to reconstruct a face using features extracted from the periocular region. A major part of inpainting is solely done using relevant pre-trained networks eliminating the need for additional training. The overall architecture of our proposed framework (see Figure 5.1) resembles coarse-and-fine architecture, but differs in several ways, detailed in the following. Our key idea is to directly map the extracted latent representation to the latent space of a pre-trained generator, as depicted in Figure 5.1. To extract a latent representation which includes identity (ID) and nonidentity (non-ID) attributes from the periocular region, we use a pre-trained face recognition method, shown by E_{id}, and a trainable network, shown by E_{at} respectively. We then map the resulting latent code Z to the latent space $\mathcal{W}$ of a pre-trained generator G, and evaluate the quality of the inpainting only on the

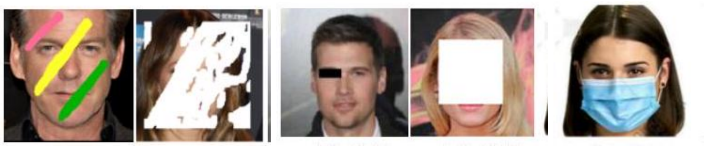


Figure 5.2: Examples of two types of widely used masks: free-form masks (a) (b), and fixed-form (c)-(e). The mask used in this work is shown in (f).

G 's output. specifically, we use $\mathcal{W}$ space to convert the extracted ID and nonID features into more disentangled space, and the Z space is created by concatenating ID and non-ID characteristics instead of using a Gaussian distribution. An optimizer is further used to find the optimal point in the $\mathcal{W}$ space based on the output of G in the last step as optimization-based GAN inversion technique, leading to address R4-R6 more precisely. This mapping empowers us to utilize a state-of-the-art pre-trained generator, inheriting its high-resolution and output diversity, with minimum training process. In our approach, the representation is split into two segments comprising separate and meaningful information (i.e., ID and non-ID information). Then the mapping network (M) is trained to extract the relevant information from the output of E_{id} and E_{at} to be merged into a proper representation of the target face. We will show that our method can effectively perform this task and inpaint the hidden region with high quality.

Further, we employ a large-size mask that covers about 75% of the face image since the goal of this paper is to complete the face based on the region of the eyes, unlike other existing works. Despite using large-size masks, we do not use any guidance information during our training in our work as compared to other related works [6, 17, 18, 19] to reduce training time and increase inference speed. Like other face inpainting methods, the performance of our face inpainting is dependent on the capabilities of the selected generator. Using StyleGAN as generator, the output of our proposed methods benefits from high image quality, outperforming all previous face inpainting methods we have compared against. In addition to being of the highest quality, our technique also successfully generates the entire face with realistic hair region, which is reported to help in identification tasks. Therefore, in contrast to state-of- the-art face inpainting methods, which need to train one or more generators [18, 26], we use a pre-trained generator reducing the training efforts. To validate our proposed method, several qualitative and quantitative metrics have been evaluated and compared with four state-of-the-art methods. Our experiments not only assess the quality of inpainted regions but also estimated demographic and ID features. Moreover, the effectiveness of reconstructing and maintaining identification elements on unseen faces, as well as the quality and diversity of faces, have been compared

Table 5.1: Comparative analysis of face inpainting methods.

Reference	one-to-one mapping	auxiliary independence	bio-facial reconstruction	high-resolution output (1024×1024)	pre-trained generator	focused on eyes-to-face	fulfilled requirements	mask type (coverage ratio)	Used losses
[17]	✓	x	x	x	x	x	R1- R4	free-form $(30-60\%)$	Perceptual Loss, Hinge loss, L_1 Loss, Style Loss, Adversarial Loss
[6]	✓	x	x	x	x	x	$\mathbf{R1-R4}$	free-form (30-60%), fixed form (50%)	Perceptual Loss, L_2 Loss, Style Loss, Total variation loss, Adversarial Loss
[28]	✓	✓	x	x	x	x	$\mathbf{R1-R4}$	fixed form (50%)	Hinge loss, Adversarial Loss
[13]	✓	✓	x	x	x	x	R1- R4	fixed form (50%)	L_1 Loss, Reconstruction Loss, Adversarial Loss
[19]	✓	x	x	x	x	x	$\mathbf{R1-R4}$	free-form $(30-60\%)$	L_1 Loss, Adversarial loss
[32]	x	✓	x	x	x	x	$\mathbf{R1-R4}$	fixed form (50%)	KL loss, Reconstruction Loss, Adversarial Loss
[26]	x	✓	x	x	x	x	R1- R4	free-form (30-60%), fixed form (50%)	L_2 Loss, KL loss, Adversarial Loss
[27]	✓	✓	x	x	x	✓	R1- R6	fixed-form (75%)	Perceptual Loss, L_1 Loss, L_2 Loss, KL loss, Adversarial Loss
[18]	✓	x	✓	x	x	✓	R1-R6	fixed-form (75%)	Perceptual Loss, Style Loss, Reconstruction Loss, Adversarial Loss
ours	✓	✓	✓	✓	✓	✓	R1- R6	fixed-form (75%)	Perceptual Loss, Style Loss, Identity Loss, Landmark Loss, Reconstruction Loss, Adversarial Loss

across all methods. Our approach has been demonstrated to outperform earlier work in addition to providing special benefits including the reconstructing of the full head and hair, preservation of ID and non-ID traits, and minimum supervision, which eliminates the need for a substantial training set.

It should be noted that different face elements impacts face recognition to varying degree as assessed in several works [54]. In this work, we attempt to extract the ID information that existed in the periocular region and preserve it in the reconstructed image. Our results show face recognition performance using inpainted images provides better accuracy than the periocular region alone, indicating our proposed algorithm not only preserves the ID information from the periocular region but also it can predict the dependent ID information and add it to reconstructed face for further recognition tasks. Further, seven new datasets of masked faces called E2F-StyleGANdb, E2F-CelebA-HQ, E2F-FFHQ, E2F-MS1MV2, E2F-LFW, E2F-CFP-FP, E2F-AgeDB-30 have been generated to train and evaluate our proposed method. Additionally, to measure the ID information in the reconstructed image, we generated two other datasets, described in section 4.

Table 5.1 presents a comparative analysis of our methodology against other related studies, focusing on key attributes essential for applications involving eye-to-face reconstruction.

5.4 Proposed method

5.4.1 Overview

As shown in Figure 5.3, given a ground-truth face image $\mathbf{I}_{gt} \in \mathbb{R}^{h \times w \times 3}$, and a binary mask $\mathbf{I}_m \in \mathbb{R}^{h \times w \times 1}$ (with value 1 for known pixels and 0 for unknown pixels), the input image $\mathbf{I}_{in} \in \mathbb{R}^{h \times w \times 3}$ is obtained as $\mathbf{I}_{in} = \mathbf{I}_{gt} \odot \mathbf{I}_m$, where $\odot$ denotes the Hadamard product. The goal is to inpaint the whole

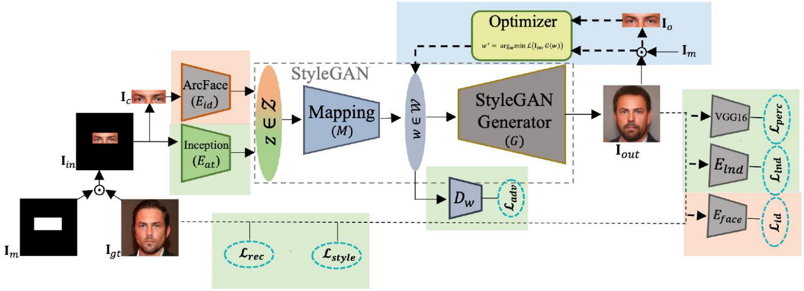
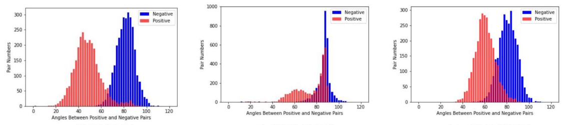


Figure 5.3: The overview of our proposed reconstructing face method (E2F-Net). Data flow and losses show by solid lines and dashed ones, respectively. First, the ID and non-ID features are extracted from masked $(\mathbf{I}_{in})$ and cropped-masked $(\mathbf{I}_c)$ images using encoders E_{id} and E_{at}, respectively. Through our mapping network M, the concatenated features are mapped to $\mathcal{W}$, the latent space of the pre-trained StyleGAN generator G. Finally, the optimal latent code in $\mathcal{W}$ space has been found using an optimizer. The blue, orange, and green highlights indicate our major contributions. The R1-R4 are being addressed by the green blocks. The block highlighted with blue is addressing R4-R6. The orange modules emphasize on R5 and R6.

face with preserving ID and other visual attributes, specifically pose, expression, and properly placing face elements with proper size. To extract ID and other attributes, we used two encoders denoted as E_{id} and E_{at} whose outputs are concatenated into z (i.e., $z = [E_{id}(\mathbf{I}_c), E_{at}(\mathbf{I}_{in})]$). Then we map the Z space to a new space called $\mathcal{W}$, and the new representation w feeds a generator. The generator generates a face based on both the ID and other facial attributes. Finally, we use an optimizer to ensure that the optimal point has been chosen in the $\mathcal{W}$ space to be fed to the generator. As depicted in Figure 5.3, the proposed E2F-Net consists of two encoders E_{id} and E_{at}, a mapping network M, a generator network called G (StyleGAN). A few additional pre-trained encoders are used for calculating corresponding multiple losses as described afterwards: feature encoder (E_{feat}), landmark encoder (E_{lnd}), and face encoder (E_{face}).

Notably, we use a state-of-the-art high-quality synthesize face generator called StyleGAN as the pretrained generator for all our experiments. Different from other GANs, StyleGAN features two latent spaces: $\mathcal{W}$, which is induced by a learned mapping from Z, and Z, which is generated by a fixed distribution. Since $\mathcal{W}$ is a more disentangled latent space than Z and is more suited to facilitate and accommodate image inpainting, we employ it to map the combined face code into it. We reduce the difficulty of learning to produce high-quality and high-fidelity images by employing this cutting-edge



a) LFW (95.6%)　b) CFP-FP (68.91%)　c) AgeDB-30 (88.28%)

Figure 5.4: Angle distributions of both positive and negative pairs on LFW, CFP-FP, and AgeDB-30. Red area indicates positive pairs while blue indicates negative pairs. All angles are represented in degrees.

generator (G). However, it is not simple to train the mapping between the latent space of the encoders $(\mathcal{Z})$ and $\mathcal{W}$. To assist M in anticipating features that lie within $\mathcal{W}$, we add a discriminator (D_w). To distinguish between real samples from StyleGAN's $\mathcal{W}$ space and M 's predictions, D_w is trained in an adversarial manner.

5.4.2 The architecture of the proposed method

The proposed E2F-Net has only three trainable modules: E_{at}, M, and D_w. The E_{id} encoder is a pre-trained face recognition model called ArcFace, trained on the edited version of the MS1MV2 dataset called E2F-MS1MV2, described in the next subsection. The E_{at} encoder is implemented as InceptionV3 [49]. The M and D_w both include four fully connected layers.

The generator, G is a pre-trained StyleGAN synthesis network, trained on FFHQ [36]. In the following subsections, we will explain each used module in detail.

5.4.2.1 Identity Encoder

To extract ID features from the periocular region, we utilized a face recognition model called ArcFace with a Resnet-50 backbone. To ensure that the features provided by ArcFace are well adapted to the periocular region, we retrained ArcFace model on a modified version of the MS1MV2 dataset called E2F-MS1MV2. To generate this dataset, all images in the MS1MV2 dataset were cropped to keep only the periocular region. By doing this, the recognition task is enforced to used eyes region alone. To validate the effectiveness of this model, we illustrate the angle distributions of both positive and negative pairs on edited versions of LFW, CFP-FP, and AgeDB-30, called E2F-LFW, E2F-CFP-FP, and E2F-AgeDB-30, in Figure 5.4. We can see that the periocular region can be very effective for face verification task, with verification accuracies for the three datasets E2F-LFW, E2F-CFP-FP, and E2F-AgeDB-30 resulting in $95.6\%, 68.91\%$, and 88.28%, respectively. Given the trained network parameters (θ_{id}), the attribute encoder (E_{id}) is fixed and used to obtain the attribute code $z_{id} \in \mathbb{R}^{512\times 1}$, i.e., $z_{id} = E_{id}\left(\mathbf{I}_c; \theta_{id}\right)$.

5.4.2.2 Attribute Encoder

E_{at} extracts non-ID features like pose, expression, illumination, skin color, etc. We used a pre-trained version of InceptionV3 [49] which has been trained on a large classification image dataset called ILSVRC 2012. Given the pretrained network parameters $\left(\hat{\theta}_{at}\right)$, the attribute encoder (E_{at}) is used to obtain the attribute code $z_{at} \in \mathbb{R}^{2048\times 1}$, i.e., $z_{at} = E_{at}\left(\mathbf{I}_{in}; \theta_{at} \mid \hat{\theta}_{at}\right)$. From the pre-trained parameters $\hat{\theta}_{at}$ we finetune until the final θ_{at} using appropriate loss functions as described in the next section.

5.4.2.3 Mapping Network

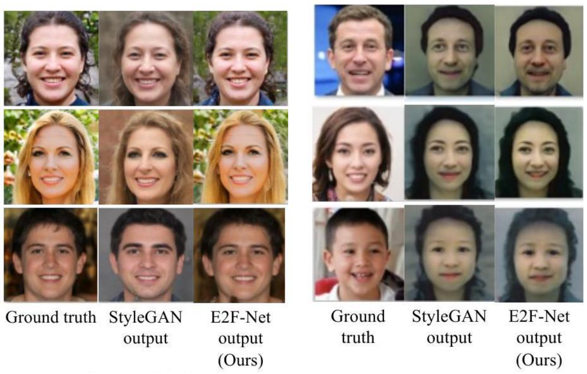
A multi-layer fully-connected neural network M, linearly maps the concatenated ID and non-ID attribute latent codes i.e., z_{id} and $z_{at}, z \in \mathbb{R}^{2560\times 1}$, to a stochastic style code $w \in \mathbb{R}^{512\times 1}$, where w lies in an extended stochastic latent space $(\mathcal{W})$. Let θ_M be the learnable parameters in M, then we have $w = M\left(z; \theta_f\right)$. Notably, M is comprised of four fully connected layers. Other network sizes are explored in the ablation section.

5.4.2.4 StyleGAN

In addition to producing impressively photorealistic, high-quality synthetic photos of faces, StyleGAN, an extension to the GAN architecture, proposes significant changes to the generator model and allows to control over the style of the generated image at various levels of detail by adjusting the style vectors and various noise parameters. Given the pre-trained StyleGAN network parameters (θ_G), the reconstructed face $(\mathbf{I}_{\text{out}})$ is the output of G i.e., $\mathbf{I}_{\text{out}} = G\left(w; \theta_G\right)$, with $\mathbf{I}_{\text{out}} \in \mathbb{R}^{h\times w\times 3}$.

5.4.2.5 Discriminator

The introduction of StyleGAN as a module in our method provides significant benefits (e.g., high realism of the output), but also comes with some challenges. In particular, it is not simple to train the mapping between the latent space Z and $\mathcal{W}$. To help M estimate features that lie within $\mathcal{W}$, we add a discriminator D_w, which is trained in an adversarial manner to distinguish between real samples from StyleGAN's $\mathcal{W}$ space and M's predictions. Note that we used E2F-StyleGANdb dataset for training D since we have latent code w for each sample and during training with E2F-CelebA-HQ we do not train this module.



(a) StyleGANdb (b) CelebA-HQ dataset

Figure 5.5: The middle column shows the output of StyleGAN before optimization and the right column after the optimization proposed in our E2F-Net.

5.4.2.6 Inversion via Optimization

To fully exploit the ability and explore the interpretability of well-trained StyleGAN models, GAN inversion has been proposed to find the optimal latent codes within $\mathcal{W}$ space. In the optimization section, we generate the reconstructed face by optimizing over the latent vector w :

$$w^{*} = \arg_{w} \min \mathcal{L} \left(\mathbf{I}_{in}, G(w) \right) \tag{3}$$

where $\mathbf{I}_{in}$ is target image and $G(w)$ is the output of StyleGAN generator. Equation (3) is a non-convex optimization problem. The used loss functions for finding the optimum w have been defined in the next section.

Algorithm 5.1: Latent Space Embedding for StyleGAN

```
Input: $\mathbf{I}_{\text{out}}$ StyleGAN output, $\mathbf{I}_{in}$ masked input, $\mathbf{I}_{\mathbf{c}}$ cropped input; $G$
  the pre-trained StyleGAN generator .
Output: optimum latent code $w^{*}$
Initialize latent code $w^{*} = w$
while not converged do
    $\mathbf{I}_{\text{out}\,_{c}} \leftarrow$ Cropped $G\left(w^{*}\right)$
    $\mathbf{I}_{\text{out}\,_{m}} \leftarrow$ Masked $G\left(w^{*}\right)$
    $\mathcal{L}_{opt} = \mathcal{L}_{perc}(\mathbf{I}_{\text{in}}\,, \mathbf{I}_{\text{out})} + \mathcal{L}_{id}(\mathbf{I}_{\text{c}}, \mathbf{I}_{\text{out}})$
    $w^{*} \leftarrow w^{*} - \eta \nabla_{w} \mathcal{L}_{opt}$
```

5.4.3 Training and losses

First, a synthetic dataset using StyleGAN called E2F-StyleGANdb has been created in the following manner. We sample 50,000 random Gaussian vectors and forward them through the pre-trained StyleGAN. Then the periocular region has been cropped for each image generated from the vectors. The Gaussian noise is transformed into a latent vector w in the forward process, from which we crop the image and capture both the image and the w vector. The E2F-StyleGANdb images are split into two parts for training and verifying (90% and 10% respectively) the proposed model. During the training, the latent vectors w are used as "real" samples for training the trainable modules. Figure 5.5 (a) shows the generated results by E2F-Net which are very close to the ground truth. Despite accurate results for the E2F-StyleGANdb dataset, this behaviour is not seen presented in real-world scenarios. To examine this, a modified version of the CelebA-HQ dataset [48] called E2F-CelebA-HQ has been created. As shown in Figure 5.5 (b), the gender and age of the person are preserved but the quality of outputs and identity of the person have not been preserved very well. To overcome this, the E2F-CelebA-HQ dataset has been used for training. The latent vectors have been obtained for all training samples of E2F-CelebA-HQ dataset by passing through E_{id}, E_{at}, and M. Similar to the previous attempt, the latent vectors w are used as "real" samples for training the trainable modules.

It is noteworthy to note that the E2F-Net model is trained in a supervised, end-to-end fashion. To achieve noteworthy results, we have used a variety of loss functions, including identity loss, landmark loss, perceptual loss, style loss, adversarial loss, and reconstruction loss for trainable components of our proposed method. More specifically, the adversarial loss $\mathcal{L}_{adv}$ ensures proper mapping to the $\mathcal{W}$ space. Identity preservation is encouraged using $\mathcal{L}_{id}$, that penalizes differences in identity between $\mathbf{I}_{gt}$ and $\mathbf{I}_{\text{out}}$. Attributes preservation is encouraged using $\mathcal{L}_{rec}$ and $\mathcal{L}_{\text{lnd}}$, which penalize pixel-level and facial landmarks differences, respectively, between $\mathbf{I}_{gt}$ and $\mathbf{I}_{\text{out}}$. In the following we describe all losses.

Perceptual Loss: Perceptual loss [51] has been utilized to guarantee the similarity of high-level structures to keep the structure information of the overall image. Therefore, instead of matching pixels between them, similar feature representations to the ground truth are required to achieve R1-R4. We calculate the perceptual loss $\left(\mathcal{L}_{\text{perc}}\right)$ by feeding the generated image $\left(\mathbf{I}_{\text{out}}\right)$ and the ground truth $\left(\mathbf{I}_{gt}\right)$ in a VGG-19 feature extractor. We then obtain feature maps φ^{gt} and φ^{out}, extracted from layer l of the pre-trained VGG19 network. The perceptual loss can be written as follows:

$$\mathcal{L}_{\text{perc}} = \sum_{l=1}^{N} \frac{\left\|\varphi_l^{gt} - \varphi_l^{\text{out}}\right\|_1}{C_l H_l W_l} \tag{4}$$

where H_l, W_l, and C_l represent the height, weight, and channel size of the l^{th} feature map, respectively. N is the number of feature maps that the VGG-19 feature extractor generates.

Style Loss: Perceptual loss aids in obtaining high-level structure and prevents the output image from deviating in content from the ground truth which assist to enhancing R1-R4. We still need to maintain consistent style elements like colors and patterns, though. This objective can be achieved by adding style loss $\left(\mathcal{L}_{\text{style}}\right)$ to the loss function. Similar to $\mathcal{L}_{\text{perc}}$, φ^{gt} and φ^{out} are extracted from VGG-19, and we define φ_l^{style} as the product of a features map (row vector) multiplied by its transpose:

$$\varphi_l^{\text{style}} = \varphi_l \varphi_l^T \tag{5}$$

We then obtain the style loss by comparing φ_l^{style} between φ^{gt} and φ^{out} :

$$L_{\text{style}} = \sum_{l=1}^{N} \left\| \frac{1}{C_l \times C_l} \frac{\varphi_l^{\text{style}\ gt} - \varphi_l^{\text{style}\ e_{\text{out}}}}{C_l H_l W_l} \right\|_1 \tag{6}$$

Identity Loss: We enforce the identity similarity between the reconstructed face $\mathbf{I}_{\text{out}}$ and the original face $\mathbf{I}_{gt}$ in the embedding space which used to achieve R5-R6. The identity loss is formulated as follows

$$\mathcal{L}_{\text{id}} = \left\|E_{\text{face}}\left(\mathbf{I}_{gt}\right) - E_{face}\left(\mathbf{I}_{\text{out}}\right)\right\|_1 \tag{7}$$

where $\|.\|_1$ is ℓ_1-norm. The E_{face} encoder is a pre-trained ArcFace model [31] with ResNet-50 backbone, trained on MS1MV2 dataset [44].

Landmark Loss: Because facial landmarks represent the potential poses of the face, we also include a sparse L_2 cycle consistency landmarks loss contributing to R1-R4. Using a pre-trained network named as E_{lnd}, landmarks are recovered. The landmark loss is formulated as follows

$$\mathcal{L}_{\text{lnd}} = \left\|E_{\text{lnd}}\left(\mathbf{I}_{\text{gt}}\right) - E_{\text{lnd}}\left(\mathbf{I}_{\text{out}}\right)\right\|_2 \tag{8}$$

A pre-trained landmarks network $\left(E_{\text{lnd}}\right)$ [50] has been used to predict 68 facial keypoints.

Reconstruction Loss: An additional loss is also used to encourage pixel-level reconstruction of $\mathbf{I}_{\text{out}}$. This loss is clearly motivated by our desire for $\mathbf{I}_{\text{out}}$ to be generally similar to $\mathbf{I}_{gt}$ and mainly address R1-R4. Notably, this loss can capture and preserve pixellevel information such as colors, illumination, and maintain texture information, not modeled by any other loss. It is calculated as the ℓ_1-norm between $\mathbf{I}_{\text{out}}$ and the corresponding ground truth $\mathbf{I}_{gt} \cdot \mathcal{L}_{rec}$ is defined as follows:

$$\mathcal{L}_{rec} = \alpha\left(1 - MS_{-}\text{ SSIM}\left(\mathbf{I}_{\text{gt}}, \mathbf{I}_{\text{out}}\right)\right) + (1-\alpha)\left\|\mathbf{I}_{\text{gt}} - \mathbf{I}_{\text{out}}\right\|_1 \tag{9}$$

where Multi-Scale Structural Similarity Index Metric (MS-SSIM) is calculated as in [13] and $\alpha = 0.84$.

Adversarial Loss: For adversarial loss, we use the non-saturating loss with R_1 regularization [42] :

$$\mathcal{L}_{adv}^{D} = -\mathop{\mathbb{E}}_{w \sim W}\left[\log D_w(w)\right] - \mathop{\mathbb{E}}_{z}\left[\log\left(1 - D_w(f(z))\right)\right] + \frac{\gamma}{2}\mathop{\mathbb{E}}_{w \sim \mathcal{W}}\left[\left\|\nabla_w D_w(w)\right\|_2^2\right] \tag{10}$$

$$\mathcal{L}_{adv}^{G} = -\mathop{\mathbb{E}}_{z}\left[\log D_w(f(z))\right] \tag{11}$$

Total objective: After defining the loss functions above, the total training objective can be expressed as:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{id}}\,\mathcal{L}_{\text{id}} + \lambda_{\text{lnd}}\,\mathcal{L}_{\text{lnd}} + \lambda_{\text{perc}}\,\mathcal{L}_{\text{perc}} + \lambda_{\text{style}}\,\mathcal{L}_{\text{style}} + \lambda_{\text{rec}}\,\mathcal{L}_{\text{rec}} \tag{12}$$

where $\lambda_{id}, \lambda_{\text{lnd}}$, λ_{perc} , λ_{style} and λ_{rec} are weights of corresponding losses, respectively. We set $\lambda_{\text{id}} = \lambda_{\text{rec}} = 1, \lambda_{\text{lnd}} = 0.001$, $\lambda_{\text{style}} = 0.1$ and $\lambda_{\text{perc}} = 0.01$ in our settings.

Optimizer Loss: Here we use ADAM optimization with Mean Square Error (MSE) and perceptual losses as the objective functions to find the optimal latent codes that can effectively approach $\mathbf{I}_{\text{out}}$ to $\mathbf{I}_{gt}$. This loss is aiding to address R4-R6. The loss function for optimization consists of two different loss terms including identity loss and perceptual loss:

$$\mathcal{L}_{opt} = \lambda_{\text{perc}}^{o}\,\mathcal{L}_{perc} + \lambda_{id}^{o}\mathcal{L}_{id} \tag{13}$$

We set $\lambda_{\text{perc}}^{o} = 0.01$ and $\lambda_{id}^{o} = 0.1$ in our settings.

Algorithm 1 shows the pseudo-code of the optimizer. Beginning with an appropriate initialization w, we look for an optimal vector w^* that minimizes the $\mathcal{L}_{\text{opt}}$, which assesses how similar the given image and the image produced by w^* are.

5.5 Experiments

The performance of the E2F-Net is assessed in this section using the newly created eyes-to-face datasets described in the next subsection. Our results have been compared with four methods: Pluralistic Image Completion (PIC) [26], EdgeConnect (EC) [17], LaFIn [6], and E2F-GAN [18]. To have fair comparison, the four methods have been retrained using the E2F-CelebA-HQ dataset. Five statistical metrics, described in subsection 4.3.1, have been used to quantitatively measure the performance difference among the methods. Additionally, we calculate the False Non-Match Rate (FNMR) between original and inpainted faces using a competitive face identification matcher [37] based on ArcFace [31] to assess the degree of retention of ID features.

5.5.1 Datasets

Experiments are conducted on seven generated datasets: E2F-StyleGANdb, E2F-CelebA-HQ, E2F-FFHQ, E2F-MS1MV2, E2FLFW, E2F-CFP-FP, E2F-AgeDB-30; which are all available on the project's webpage. The images are resized to 256×256, and then a landmark detector [53] is used to locate and clip the eyes in order to extract the periocular area from each facial image. Furthermore, we eliminated deceptive samples utilizing WHENet methods [52], such as those with sunglasses over their eyes or faces that were tilted more than 45 degrees in one direction (roll, pitch, yaw), which would have hidden one of their eyes.

- E2F-StyleGANdb: A high-quality image dataset that consists of 50,000 pairs of $(\mathbf{I}_{gt}, \mathbf{I}_{in})$ images collected from StyleGAN outputs. We randomly selected 45,000 images for training and the remaining 5,000 images for testing. Each image has been resized to 256×256.

- E2F-CelebA-HQ: A high-quality image dataset that consists of 24,564 portrait images collected from a publicly available dataset [48]. We randomly selected 22,879 images for training and the remaining 1,685 images for testing. Each image has been resized to 256×256.

- E2F-FFHQ: A high-quality image dataset with more variations, consisting of 70,000 face images from a publicly available dataset known as FFHQ dataset [20]. All samples are used for testing the proposed E2F-Net. Each image has been resized to 256×256.

- E2F-MS1MV2: The original version of E2F-MS1MV2 called MS1MV2 [44] includes 85k identities and 5.8M images. After applying a landmark detector to extract the periocular region from each image, 83.8 K identities and 5.6M images remained. This dataset has been used to train E_{id}.

- E2F-LFW: E2F-LFW is a modified version of LFW [45] including 6,000 pairs of faces in the validation part. After applying a landmark detector to extract the periocular region from each image, 5,996 pairs remained. This dataset has been used to evaluate E_{id}.

- E2F-CFP-FP: E2F-CFP-FP is a modified version of CFP-FP [46] including 6,000 pairs of faces in the validation part. After applying a landmark detector to extract the periocular region from each image, 5,998 pairs remained. This dataset has been used to evaluate E_{id}.

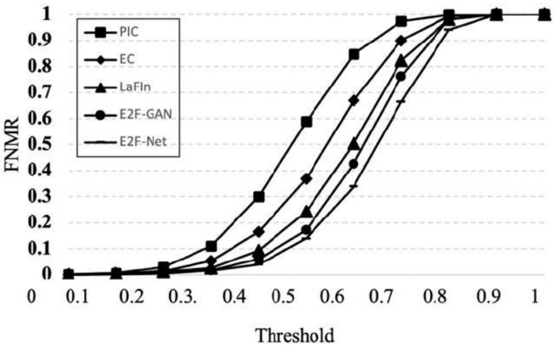


Figure 5.6: FNMR curves are displayed for our proposed method (E2F-Net) and other compared methods (PIC, EC, LaFIn, E2F-GAN) using the E2F-CelebA-HQ dataset.

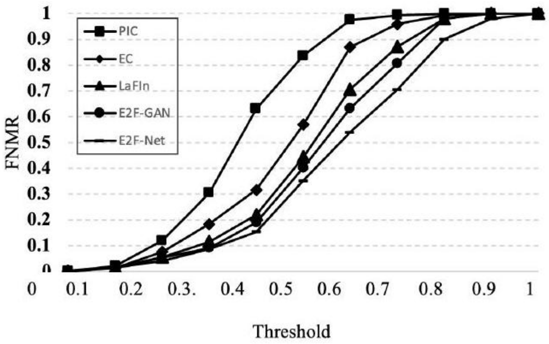


Figure 5.7: FNMR curves are displayed for our proposed method (E2F-Net) and other compared methods (PIC, EC, LaFIn, E2F-GAN) using the E2F-FFHQ dataset.

- E2F-AgeDB-30: E2F-AgeDB-30 is a modified version of AgeDB-30 [47] including 6,000 pairs of faces in the validation part. After applying a landmark detector to extract the periocular region from each image, 5,993 pairs remained. This dataset has been used to evaluate E_{id}.

5.5.2 Comparison methods

In this work, we compare our method with four state-of-the-art inpainting methods, which are summarized as follows:

PIC [26]: PIC takes advantage of a dual pipeline using variational autoencoders that consists of a reconstructive path that uses the ground truth to learn the prior distribution of missing regions and a generative path for which the conditional prior is connected to the distribution learned in the reconstructive path. It should be noted that, because the PIC approach produces distinct outputs for a certain input, it has been executed five times and the best outcomes have been reported.

EC [17]: By predicting the edges using an edge generator, EC feeds the damaged image and the predicted edges to the image inpainting network.

LaFIn [6]: LaFIn is an inpainting GAN-based network that uses predicted landmarks as guidance.

E2F-GAN [18]: E2F-GAN is a GAN-based coarse-to-fine method such that the coarse module benefits from the coarse-and-fine architecture. Moreover, an edge detector has been utilized to provide more information for the designed network.

5.5.3 Evaluation metrics

Through quantitative and qualitative comparisons, we assess the proposed model's face inpainting performance. Two different types of metrics, comprising five statistics and one identity measure, have been calculated for quantitative analysis. We give a brief overview of each category and the associated metrics in the sections that follow.

5.5.3.1 Statistical metrics

ℓ_1 loss [20]. A simple and popular loss function used in the generation of images is the pixel-wise ℓ_1 loss. This loss function measures the discrepancies between the synthesized content and the corresponding ground truth at the pixel level:

$$\ell_1\left(\mathbf{I}_{gt}, \mathbf{I}_{\text{out}}\right) = \frac{1}{hw} \sum_{i=1}^{h} \sum_{j=1}^{w} \left\| \mathbf{I}_{gt_{ij}} - \mathbf{I}_{\text{out}} \right\|_1 \tag{5.0}$$

Peak Signal to Noise Ratio (PSNR) [38]:

$$\operatorname{PSNR}\left(\mathbf{I}_{\text{gt}}, \mathbf{I}_{\text{out}}\right)=10 \log_{10}^{hw} \sum_{i=1}^{h} \sum_{j=1}^{W}\left(I_{gt_{ij}}-I_{\text{out}}\right)^{2} \tag{5.1}$$

Structural Similarity (SSIM) [13]. The SSIM describes the degree of structural similarity between two images:

$$\operatorname{SSIM}\left(\mathbf{I}_{\text{gt}}, \mathbf{I}_{\text{out}}\right)=\frac{\left(2 \mu_{\text{gt}} \mu_{\text{out}}+C_{1}\right)\left(2 \sigma_{\text{gt}} \sigma_{\text{out}}+C_{2}\right)}{\left(\mu_{\text{gt}}^{2}+\mu_{\text{out}}^{2}+C_{1}\right)\left(\sigma_{\text{gt}}^{2}+\sigma_{\text{out}}^{2}+C_{2}\right)} \tag{16}$$

where C_1 and C_2 are positive constants added to prevent cases in which the denominator is zero.

Frechet Inception Distance (FID) [30]. Utilizing the Wasserstein distance between the distributions of the actual and created images in the feature space acquired by the Inception model [30], this metric assesses the visual quality and variety of the generated images. The FID can be expressed as:

$$\operatorname{FID}\left(\mathbf{I}_{\text{gt}}, \mathbf{I}_{\text{out}}\right)=\left\|\mu_{\text{gt}}-\mu_{\text{out}}\right\|_{2}^{2}+\operatorname{Tr}\left(\sigma_{\text{gt}}+\sigma_{\text{out}}-2\left(\sigma_{\text{gt}} \sigma_{\text{out}}\right)^{\frac{1}{2}}\right) \tag{17}$$

In both SSIM and FID metrics, μ_{gt} and μ_{out} donate the mean values of $\mathbf{I}_{gt}$ and $\mathbf{I}_{\text{out}}$, respectively; while σ_{gt} and σ_{out} represent the covariance of $\mathbf{I}_{gt}$ and $\mathbf{I}_{out}$, respectively.

Total Variation (TV) [37]. TV calculates the total of the absolute differences for nearby pixels as formulated below to help quantify the degree of noise in the image:

where N_h and N_w are the number of pixels in $\mathbf{I}_{\text{out}}$ except for the last row and the last column, respectively.

5.5.4 Identity metrics

We used false non-match rate (FNMR) to measure the preserved ID attributes in inpainted images. More specifically, FNMR measures the miss-categorization rate for some pairs of face images where each pair belongs to the same individual. Here, we assumed that $\mathbf{I}_{\text{in}}$ and $\mathbf{I}_{\text{out}}$ are two faces for the same individual. Using E_{face}, the corresponding embedding vectors for each face have been obtained, and the cosine similarity for each pair of $\mathbf{I}_{\text{in}}$ and $\mathbf{I}_{\text{out}}$ has been calculated. Finally, the FNMR for different thresholds has been depicted.

5.5.5 Implementation details

We use StyleGAN pre-trained at 256x256 resolution in all our experiments. Training is performed using the Adam optimizer, with $\beta_1 = 0.9$ and $\beta_2 =$

Table 5.2: Quantitative results over E2F-CelebA-HQ dataset for E2F-Net and other compared methods (PIC, EC, LaFIn, E2F-GAN). The best result of each column is boldfaced. ↑ indicates that the higher the number the better is the model and ↓ indicates the lower the number the better is the model.

Method	FID ↓	SSIM ↑	PSNR ↑	TV ↓	ℓ_1 Loss ↓
PIC	57.02	0.41	11.19	8.50	50.37
EC	70.63	0.42	12.67	5.27	121.08
LaFIn	63.16	0.47	13.18	6.89	40.94
E2F-GAN	46.39	0.51	13.66	**0.02**	41.54
E2F-Net (Ours)	**45.85**	**0.53**	**13.78**	**0.02**	**40.36**

Table 5.3: Quantitative results over E2F-FFHQ dataset for E2F-Net and other compared methods (PIC, EC, LaFIn, E2F-GAN). The best result of each column is boldfaced. ↑ indicates that the higher the number the better is the model and ↓ indicates the lower the number the better is the model.

Method	FID ↓	SSIM ↑	$PSNR$ ↑	TV ↓	ℓ_1 Loss ↓
PIC	143.89	0.37	10.03	10.54	68.9
EC	134.63	0.37	10.84	7.59	197.84
LaFIn	97.48	0.43	11.32	6.98	55.29
E2F-GAN	101.27	0.45	11.52	**0.02**	53.64
E2F-Net (Ours)	**91.14**	**0.49**	**12.12**	**0.02**	**49.12**

0.999. On a single NVIDIA GeForce RTX 3090 GPU, the network is trained end-to-end with batch sizes of 16 and converges in roughly two days. It should be noted that this is quite effective considering that training StyleGAN would take more than 35 days on the same GPU.

5.5.6 Comparison with previous works

We qualitatively and quantitatively compare our results against four state-of-the-art approaches, i.e., PIC, EC, LaFIn, and E2FGAN, using the above-mentioned metrics and plotting some outputs. It should be noted that we trained the four methods using our own constructed training dataset, E2F-CelebA-HQ, using the best reported setups for each method described in the respective articles. The obtained results based on the E2F-CelebA-HQ and E2F-FFHQ validation datasets have been presented in the following subsections.

5.5.7 Quantitative comparisons

The results of statistical metrics calculated on the validation set of the E2F-CelebA-HQ dataset including 1,685 samples are reported in Table 5.2. As can be observed, E2F-Net is superior over PIC, EC, LaFIn, and E2F-GAN in most metrics except TV loss which is equal to E2F-GAN. Overall, the E2F-Net outperforms the other methods in terms of FID, SSIM, PSNR, and ℓ_1 metrics (addressing R1-R4). More precisely, the significant margins in FID and ℓ_1 measures show that, in comparison to previous approaches, our method can inpaint the masked image with a significantly greater level of quality. The large margin between our proposed method and others is also patent in Table 5.3, when the E2F-FFHQ dataset has been used as a validation set. We conducted t-tests to statistically validate the E2F-Net model's superior performance in metrics such as FID, SSIM, PSNR, TV, and l_1 loss, against other methods with p-values under 0.05 . Our statistical analysis revealed that our method outperforms other techniques across various metrics for both datasets, with the exception of the TV metric. Specifically, the TV metric showed no significant statistical difference when comparing our method with the E2F-GAN across both E2F-CelebA-HQ and E2F-FFHQ datasets.

Moreover, to measure the amount of preserving ID features, FNMR has been calculated for both datasets as shown in Figures. 5.6 and 5.7 (addressing R6). E2F-Net has a decreased false non-match rate at various thresholds, demonstrating the capability of our system to extract ID from the periocular area and transfer it to the reconstructed face.

Additionally, to quantitatively measure the preserved demographic information (e.g., age, gender) (addressing R5), we used OpenCV age and gender estimation library, to compare the reconstructed faces with ground truth. Notably, the E2F-CelebAHQ, the modified version of CelebA-HQ, is an ill-biased/imbalanced dataset over certain attributes such as gender, skin color, and age.

Regarding gender, 66 percent of training images are female, and 34 percent are male, indicating female gender representation over male gender during training. Furthermore, this imbalance also exists for validation images, 58 percent are female and 42 percent are male. Regarding age, four intervals are considered, and the percentage of each interval is reported in Table 5.4. The training and validation sets are imbalanced such that the second interval (i.e., between 15 and 40) has the maximum samples ($\approx 55\%$) and the fourth interval (i.e., older than 60) has the minimum samples (i.e., $\approx 0.1\%$ in training set and $\approx 1\%$ in validation set). Considering the validation set percentage as the baseline for each interval, our proposed method (E2F-Net) predicts the age attribute. Regarding gender, as shown in Table 5.5 our proposed method along with E2F-GAN and LaFIn methods can preserve the attributes of both groups very well (i.e., 41.7% out of 42% of men and 57.9% out of 58% of women).

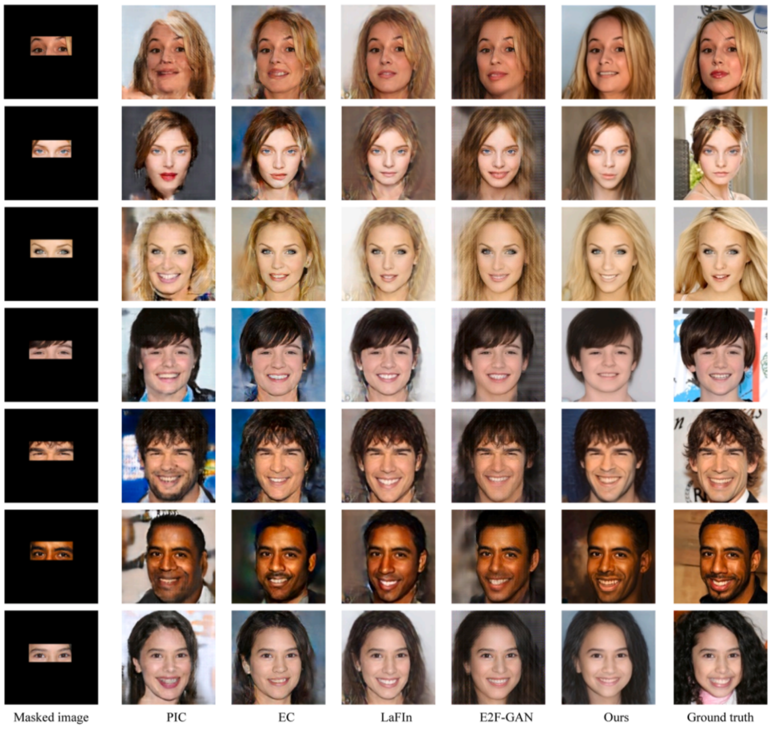


Figure 5.8: Quality comparison among PIC, EC, LaFIn, E2F-GAN, and our proposed method using E2F-CelebA-HQ dataset. very close to the baseline.

5.5.8 Qualitative comparisons

A few samples of results are displayed in Figures 5.9 and 5.10 for E2F-CelebA-HQ and E2F-FFHQ datasets, respectively. As it can be observed, the quality of PIC and EC is really low compared to E2F-GAN, LaFIn, and our results. Moreover, in comparison with other methods, our method generates high quality and highly structured faces (addressing R1-R4). Additionally, to measure the amount of preserved demographic information (addressing R5), we present a variety of faces in Figures 5.8 and 5.9. For instance, Figure 5.8 rows 2, 4, and 7 show very young man and women with faces reconstructed preserved well ID features. Similar high-quality results are demonstrated in Figure 5.10 rows 5 and 6 for two elderly men.

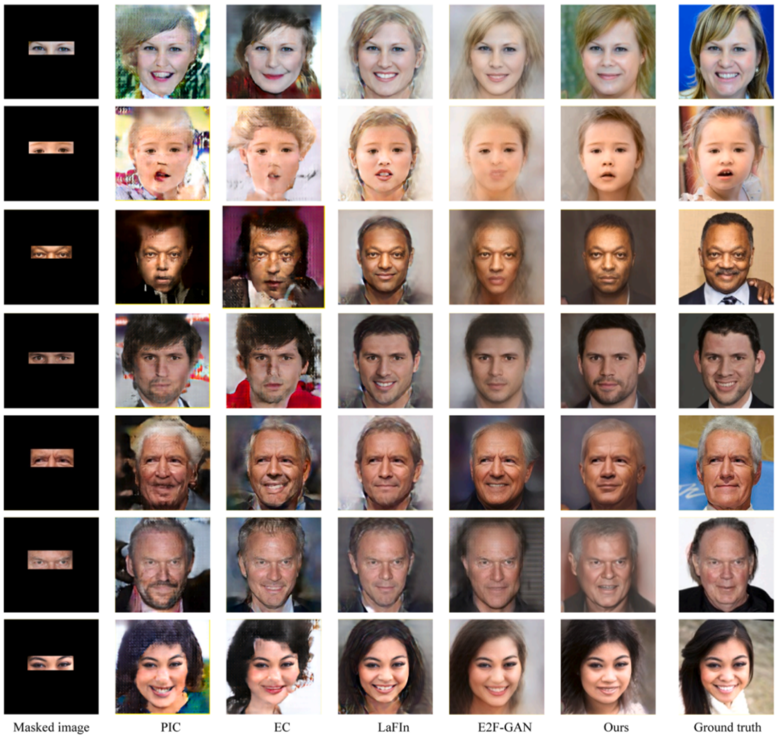


Figure 5.9: Quality comparison among PIC, EC, LaFIn, E2F-GAN, and our proposed method using E2F-FFHQ dataset.

5.6 Ablation study

Number of layers of the mapper (M). A fully-connected network has been used to map Z space latent codes to $\mathcal{W}$ space. Notably, in the original StyleGAN network an eight-layer fully connected network was proposed. To analyze the optimum mapper for this task, we have done the experiments using a different number of layers: 2, 4, and 8 layers. As shown in Table 5.6, a 4-layer fully-connected mapper generates better quantitative results.

Impact of optimizer: Table 5.7 shows quantitative metrics for the initial output of StyleGAN and after executing our latent embedding optimizer. We show the results of 25^{th} , 100^{th} , and 200^{th} iterations. Utilizing an RTX 3090 graphics card, we recorded the optimization time consumed after the 25^{th} , 100^{th} , and 200^{th} epochs, detailed in Table 5.7 and discussed in [28]. Notably, the consumed time before initiating the optimization process is 0.48

Table 5.4: Age Evaluation of our proposed method (E2F-Net), PIC, EC, LaFIn, and E2F-GAN using CelebA-HQ dataset. numbers are shown in percent.

	age<=15	15<age<=40	40<age<=60	age>60
E2F-CelebA-HQ (training set)	38.1	54.8	7	0.1
E2F-CelebA-HQ validation set (baseline)	37	55	7	1
PIC	24	48.9	5.6	0.4
EC	16.4	45.6	4.4	0.2
LaFIn	34.8	53.2	6.3	0.7
E2F-GAN	34.8	53.2	6.2	0.8
E2F-Net (ours)	35.1	53.9	6.5	0.8

Table 5.5: Gender Evaluation of our proposed method (E2F-Net), PIC, LaFIn, EC, and E2F-GAN using CelebA-HQ dataset. numbers are shown in percent.

	male	female
CelebA-HQ training set	34	66
CelebA-HQ validation set (baseline)	42	58
PIC	40.6	56.8
EC	40.1	56.6
LaFIn	41.7	57.9
E2F-GAN	41.7	57.9
E2F-Net (ours)	**41.7**	**57.9**

Table 5.6: Quantitative results over E2F-CelebA-HQ dataset using 2, 4, 8-layer mapper. for each metric, a triplet including the initial output of StyleGAN, the output of StyleGAN after 25 and 200 iterations on f have been reported respectively.

Method	FID ↓	SSIM ↑	PSNR ↑	TV ↓	ℓ_1 Loss ↓
2-layer mapper (initial/25/200)					
4-layer mapper (initial/25/200)	54.19/50.87/48.61	0.47/0.49/0.49	13.62/13.57/13.38	0.02/0.018/0.019	40.77/39.77/40.25
8-layer mapper (initial/25/200)					

second.

For all metrics, the optimizer has a positive impact. Figure 5.10 shows the impact of the optimizer on improving quality, identity features, and facial expression.

Impact of our inpainting method on face/periocular recognition: To measure if our proposed inpainting method preserves or adds further ID information in the reconstructed face, we created two datasets and used E_{id} (i.e., trained on periocular region) and E_{face} (i.e., trained on face) to calculate FMR and FNMR curves.

We used CelebA-HQ validation set to create the required datasets. First,

Table 5.7: the quantitative results of the initial outputs of StyleGAN and after optimizing the latent code in three different iterations over E2F-CelebA-HQ dataset and consumed time using 4-layer mapper have been reported.

Method	$FID\downarrow$	$SSIM\uparrow$	$PSNR\uparrow$	$TV\downarrow$	ℓ_1 Loss ↓	Time [sec]
Initial output	49.8586	0.479774	13.6780	0.0208	40.360	0.48
25th iteration	48.3770	0.49789345	13.6619	0.0186	39.417	2.04
100th iteration	$\mathbf{47.1548}$	$\mathbf{0.5002335}$	$\mathbf{13.5310}$	$\mathbf{0.0181}$	$\mathbf{39.384}$	6.4
200th iteration	47.2831	0.49711797	13.4402	0.0197	40.056	12.7

a dataset called sub-CelebA including 960 face pairs (480 positive pairs such that both images are belonging to the same subject and 480 negative pairs in which the images of each pair are belonging to different subjects) have been selected. Then all images within the sub-CelebA dataset are cropped such that just the periocular region remained, we called this dataset p-CelebA (p for periocular). Finally, again using sub-CelebA, we create the inpaint-CelebA dataset such that one of the faces in each pair is kept, and the other one is replaced with its reconstructed version. Therefore, to measure the performance of E2F-Net regarding preserving ID information, we feed the p-CelebA to E_{id} and inpaintCelebA to the E_{face} . The results including the angle distributions, the FMR, FNMR, and ROC curves for the above-mentioned datasets are shown in Figure 5.11. The FNMR curves drawn using the E_{face} increase more smoothly compared to FNMR curves drawn by E_{id}, and the FMR curve drawn by E_{face} decreases more quickly compared to FMR curves drawn by E_{id}, leading to less EER value (crossing point between FMR and FNMR curves). Notably, the large gap between FMRs curves demonstrates that the inpainting-based face recognition can reduce inter-class distances significantly. We hypothesize it as a result of our proposed architecture being able to extract a great amount of ID information from the periocular region and transmit to inpainted face. The accuracies for p-CelebA and inpaint-CelebA datasets are 90.81% and 96.04%, respectively.

Impact of other types of masks: while the current study focused on eyes-to-face task, we have checked the capability of E2F-Net for four other types of masks including free-form and fixed-form. Notably, since E_{id} is trained to extract identity information from periocular region, we cannot use masks that cover this region. The results are presented in Figure 5.12.

5.7 Discussion

Regarding preserving the ID information of each person (i.e., addressing R6), we have done two main experiments. First, the outputs of our proposed method are compared with other methods by calculating FNMR curves on two datasets (i.e., E2F-CelebA-HQ and E2F-FFHQ) shown in Figures 5.7 and

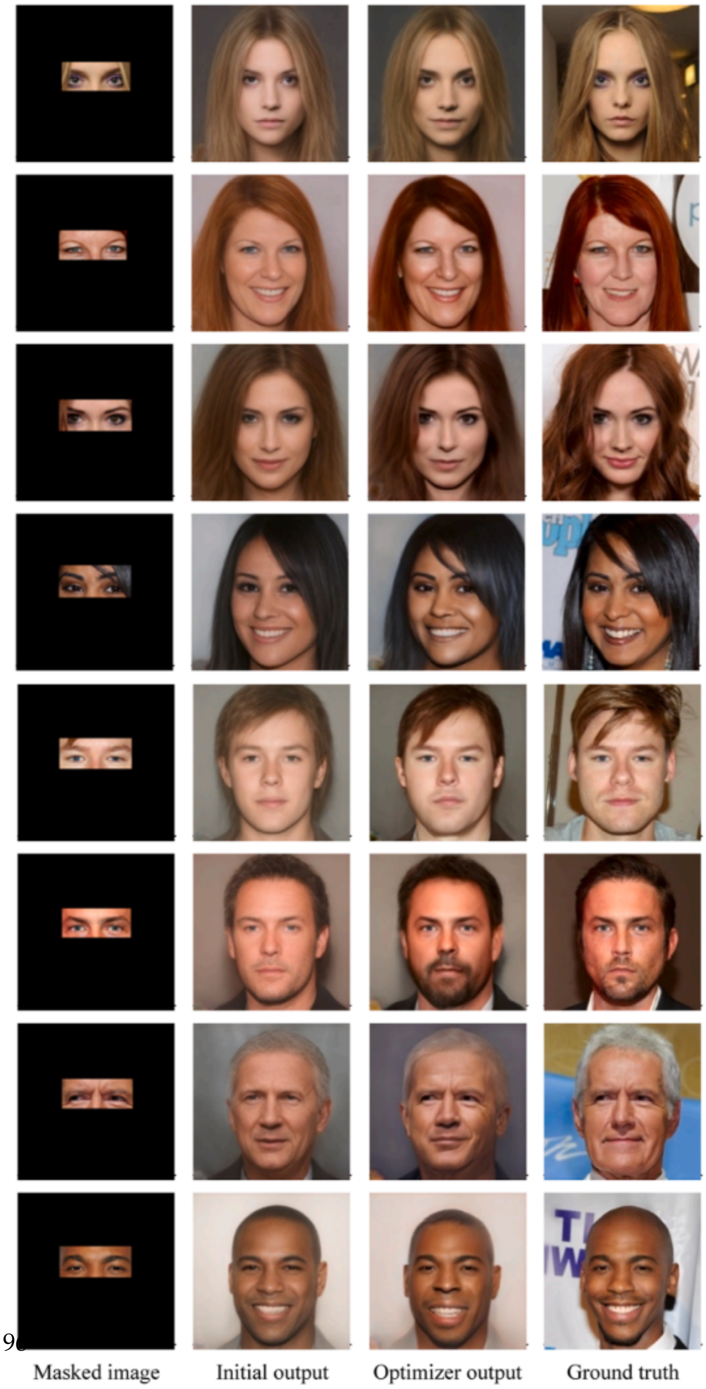


Figure 5.10: Impact of optimizer on improving quality, ID features, and facial expression of StyleGAN output.

5.8. Second, we have explored if inpainted full faces are more adequate for person ID compared to the input periocular images. Our conducted experiments (i.e., in section V) show that inpainted full face recognition improves the verification performance over periocular-based person verification. Common sense tells us that 1) inpainting properly keeps ID information, and 2) the state-of-the-art face recognition models work better in full face compared to periocular images.

Limitations of our Work: Although our proposed method can reconstruct a wide variety of faces while preserving ID features, there are two main limitations. First, the color of the scalp hair and eyebrow hair can result in different colors (as shown in Figure 5.13 row 1). Detecting and properly inpainting these elements may not be feasible without other cues if, e.g., part of the scalp hair is not visible in the periocular region, if the person hides part of the scalp hair (e.g., by a hat as shown in Fig 5.13 row 2), or misses part the of scalp hair (see Figure 5.13 row 3). Another issue for men is the difficulty to detect the existence of a beard on the face based on the periocular region. Second, the existence of occlusion or closed eyes may lead to reducing the quality of outputs as shown in Figure 5.14.

5.8 Conclusion and future work

In this paper, we show that a variety of faces can be reconstructed using only the periocular region by our proposed

GAN-based network called E2F-Net. For this purpose, a pre-trained face generator called StyleGAN has been used such that our proposed method benefits from not only minimum training process but also high image quality and diverse facial outputs. Moreover, to carefully extract ID features from the periocular region, we used a face recognition model called ArcFace which is retrained on E2F-MS1MV2 dataset, a generated identity recognition dataset based on the eyes region. Notably, we reveal that ID and non-ID features can be extracted from the eyes region and finally reconstruct the whole face based on these features. We conducted extensive experiments on two datasets including a high diversity of faces with different gender, ethnicity (e.g., Caucasian, Asian, African), pose (e.g., frontal, upward, downward), and expression (e.g., smiling, neutral), and show that our method successfully reconstructs the whole face with high quality.

Despite promising effectiveness, the proposed method still needs to be further improved: (1) the capacity of the generated GAN latent space through adversarial loss to represent the space from ground-truth data is challenging to measure. While current experiments with different mapper and discriminator architectures provide some insights, there's still uncertainty about the adequacy of the latent space representation. Future work should explore

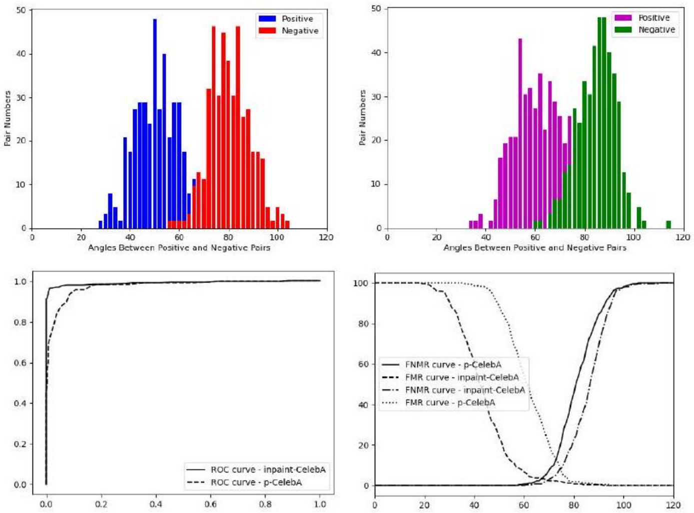


Figure 5.11: Comparing the performance of E_{face} and E_{id} fed by inpaint-CelebA and p-CelebA datasets, respectively. The first row shows the angle distributions for positive and negative pairs for both inpaint-CelebA (left) and p-CelebA (right) datasets. The ROC, FMR and FNMR curves for both datasets are shown in the second row.

novel ways to validate the representation capacity of the GAN latent space. (2) The size of the training dataset and its influence on generalization is another concern. With the current architecture relying on only three trainable components and relatively shallow networks for M, and D_w, the impact of a larger dataset may be limited. However, future studies could explore scaling up the architecture to leverage larger datasets more effectively. (3) To address the challenges associated with hair and eyebrow color consistency, the presence of occlusions, and the detection of facial hair from limited visual information. These factors currently impede the method's reliability in reconstructing facial features with high fidelity. (4) by systematically manipulating various facial elements, future research could yield valuable insights into the differential contributions of ID and non-ID features to facial recognition. (5) we fine-tuned the model's trainable components using high-resolution facial datasets (i.e., E2F-CelebA-HQ) which resulted in the model being specialized for high-resolution imagery. However, certain use cases, like public security and criminal identification, may not always have high-resolution images available. Therefore, subsequent research could focus on refining our model to perform effectively in situations where only lower-

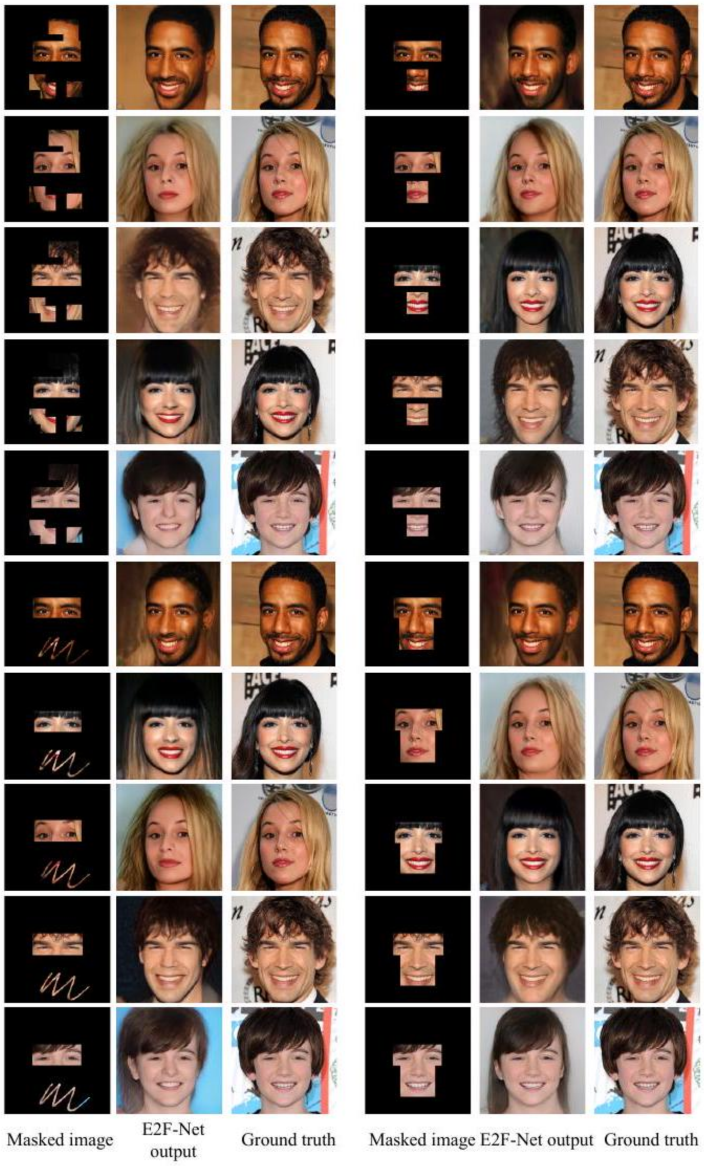


Figure 5.12: Comparative analysis of E2F-Net's performance with various mask types.

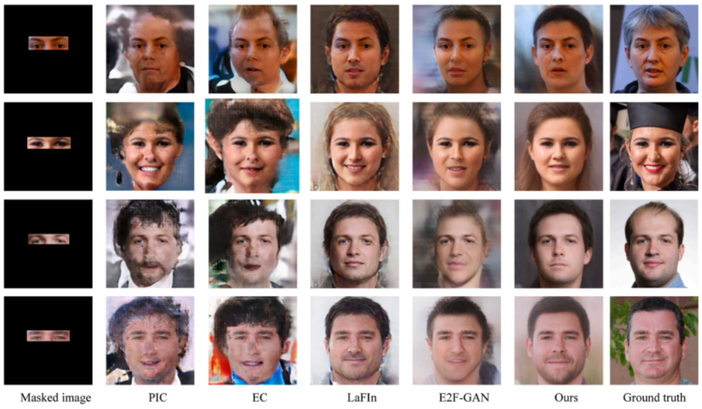


Figure 5.13: Quality comparison among PIC, EC, LaFIn, E2F-GAN, and our proposed method using E2F-FFHQ dataset.

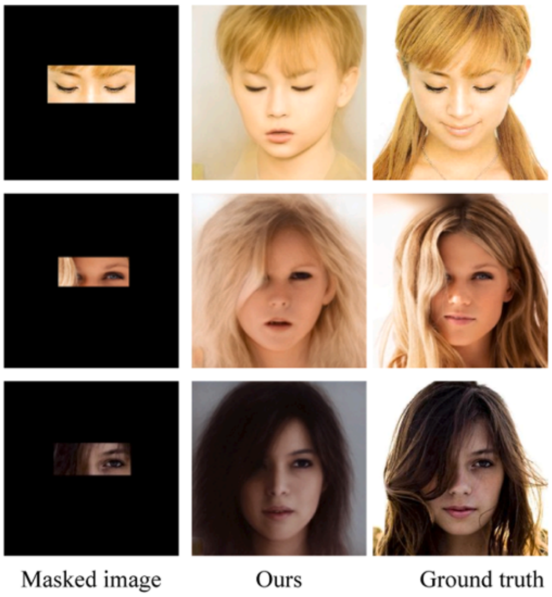


Figure 5.14: Example impact of closed eyes and eyes occlusion on output.

resolution images are accessible.

References

[1] M. Mori, The uncanny valley, Energy, vol. 7, no. 4, pp. 33-35, 1970.

[2] R.A. Yeh, C. Chen, T.Y. Lim, A.G. Schwing, M. Hasegawa-Johnson, M.N. Do, Semantic image inpainting with deep generative models, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2017.

[3] T. Wang, H. Ouyang, Q. Chen, Image inpainting with external-internal learning and monochromic bottleneck, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2021.

[4] R. Xu, M. Guo, J. Wang, X. Li, B. Zhou, C.C. Loy, Texture memory-augmented deep patch-based image inpainting, IEEE Trans. Image Process. 30 (2021) 9112-9124. https://doi.org/10.1109/TIP.2021.3122930.

[5] Y. Yang, X. Guo, J. Ma, L. Ma, H. Ling, LaFIn: Generative landmark guided face inpainting, arXiv [Cs.CV]. (2019). http://arxiv.org/abs/1911.11394.

[6] Y. Wang, A. Liu, R. Tucker, J. Wu, B.L. Curless, S.M. Seitz, N. Snavely, Repopulating street scenes, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2021.

[7] W. Quan, R. Zhang, Y. Zhang, Z. Li, J. Wang, D.-M. Yan, Image inpainting with local and global refinement, IEEE Trans. Image Process. 31 (2022) 24052420. https://doi.org/10.1109/TIP.2022.3152624.

[8] J. He, B. Xiao, X. Zhang, S. Lei, S. Wang, C.-T. Lu, Reducing noise pixels and metric bias in semantic inpainting on segmentation map, in: 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), IEEE, 2021.

[9] J. Cho, S. Yun, D. Han, B. Heo, J.Y. Choi, Detecting and removing text in the wild, IEEE Access. 9 (2021) 123313-123323.

[10] Y. Jiang, F. Yang, Z. Bian, C. Lu, S. Xia, Mask removal: Face inpainting via attributes, Multimed. Tools Appl. 81 (2022) $29785 - 29797$. https://doi.org/10.1007/s11042-022-12912-1.

[11] H. Xiang, Q. Zou, M.A. Nawaz, X. Huang, F. Zhang, H. Yu, Deep learning for image inpainting: A survey, Pattern Recognit. 134 (2023) 109046. https://doi.org/10.1016/j.patcog.2022.109046.

[12] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, T.S. Huang, Generative image inpainting with contextual attention, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, 2018.

[13] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, T. Huang, Free-form image inpainting with gated convolution, in: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), IEEE, 2019.

[14] Z. Guo, Z. Chen, T. Yu, J. Chen, S. Liu, Progressive image inpainting with full-resolution residual network, in: Proceedings of the 27th ACM International Conference on Multimedia, ACM, New York, NY, USA, 2019.

[15] H. Zhang, Z. Hu, C. Luo, W. Zuo, M. Wang, Semantic image inpainting with progressive generative networks, in: Proceedings of the 26th ACM International Conference on Multimedia, ACM, New York, NY, USA, 2018.

[16] K. Nazeri, E. Ng, T. Joseph, F. Qureshi, M. Ebrahimi, EdgeConnect: Structure guided image inpainting using edge prediction, in: 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), IEEE, 2019.

[17] A. Hassanpour, A.E. Daryani, M. Mirmahdi, K. Raja, B. Yang, C. Busch, J. Fierrez, E2F-GAN: Eyes-to-face inpainting via edge-aware coarse-to-fine GANs, IEEE Access. 10 (2022) 32406-32417.

[18] M. Chen, Z. Liu, EDBGAN: Image inpainting via an edge-aware dual branch generative adversarial network, IEEE Signal Process. Lett. 28 (2021) $842 - 846$.

[19] P. Isola, J.-Y. Zhu, T. Zhou, A.A. Efros, Image-to-image translation with conditional adversarial networks, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2017.

[20] W. Cai, Z. Wei, PiiGAN: Generative adversarial networks for pluralistic image inpainting, IEEE Access. 8 (2020) $48451 - 48463$.

[21] N. Ud Din, K. Javed, S. Bae, J. Yi, A novel GAN-based network for unmasking of masked face, IEEE Access. 8 (2020) $44276 - 44287$.

[22] Y. Li, J. Yan, J. Wang, GPG-NET: Face inpainting with generative parsing guidance, in: 2021 IEEE International Conference on Image Processing (ICIP), IEEE, 2021.

[23] J. Wang, S. Chen, Z. Wu, Y.-G. Jiang, FT-TDR: Frequency-guided transformer and top-down refinement network for blind face inpainting, IEEE Trans. Multimedia. 25 (2023) 2382-2392.

[24] X. Zhang, C. Shi, X. Wang, X. Wu, X. Li, J. Lv, I. Mumtaz, Face inpainting based on GAN by facial prediction and fusion as guidance information, Appl. Soft Comput. 111 (2021) 107626. https://doi.org/10.1016/j.asoc.2021.107626.

[25] C. Zheng, T.-J. Cham, J. Cai, Pluralistic Image Completion, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2019.

[26] X. Luo, X. He, L. Qing, X. Chen, L. Liu, Y. Xu, EyesGAN: Synthesize human face from human eyes, Neurocomputing. 404 (2020) $213 - 226$. https://doi.org/10.1016/j.neucom.2020.04.121.

[27] M. Chen, Z. Liu, L. Ye, Y. Wang, Attentional coarse-and-fine generative adversarial networks for image inpainting, Neurocomputing. 405 (2020) $259 - 269$. https://doi.org/10.1016/j.neucom.2020.03.090.

[28] E. Bayraktar, Y. Wang, A. DelBue, Fast re-OBJ: real-time object re-identification in rigid scenes, Mach. Vis. Appl. 33 (2022).

[29] H. Liu, B. Jiang, Y. Xiao, C. Yang, Coherent Semantic Attention for Image Inpainting, in: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), IEEE, 2019.

[30] J. Deng, J. Guo, N. Xue, S. Zafeiriou, ArcFace: Additive angular margin loss for deep face recognition, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2019.

[31] L. Zhao, Q. Mo, S. Lin, Z. Wang, Z. Zuo, H. Chen, W. Xing, D. Lu, UCTGAN: Diverse image inpainting based on unsupervised cross-space translation, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2020.

[32] Y. Shen, J. Gu, X. Tang, B. Zhou, Interpreting the latent space of GANs for semantic face editing, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2020.

[33] R. Abdal, Y. Qin, P. Wonka, Image2StyleGAN: How to embed images into the StyleGAN latent space?, in: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), IEEE, 2019.

[34] Z. Wu, D. Lischinski, E. Shechtman, StyleSpace analysis: Disentangled controls for StyleGAN image generation, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2021.

[35] T. Karras, S. Laine, T. Aila, A style-based generator architecture for generative adversarial networks, IEEE Trans. Pattern Anal. Mach. Intell. 43 (2021) 4217− 4228. https://doi.org/10.1109/TPAMI.2020.2970919.

[36] A. Mahendran, A. Vedaldi, Understanding deep image representations by inverting them, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2015.

[37] Y.-G. Shin, M.-C. Sagong, Y.-J. Yeo, S.-W. Kim, S.-J. Ko, PEPSI++: Fast and lightweight network for image inpainting, IEEE Trans. Neural Netw. Learn. Syst. 32 (2021) 252-265. https://doi.org/10.1109/TNNLS.2020.2978501.

[38] O. Tov, Y. Alaluf, Y. Nitzan, O. Patashnik, D. Cohen-Or, Designing an encoder for StyleGAN image manipulation, ACM Trans. Graph. 40 (2021) 1-14. https://doi.org/10.1145/3450626.3459838.

[39] M. Huh, R. Zhang, J.-Y. Zhu, S. Paris, A. Hertzmann, Transforming and projecting images into class-conditional generative networks, in: Computer Vision ECCV 2020, Springer International Publishing, Cham, 2020: pp. 17-34.

[40] Y. Alaluf, O. Tov, R. Mokady, R. Gal, A.H. Bermano, HyperStyle: StyleGAN inversion with HyperNetworks for real image editing, arXiv [Cs.CV]. (2021). http://arxiv.org/abs/2111.15666.

[41] L. Mescheder, A. Geiger, and S. Nowozin, Which training methods for GANs do actually converge?, In International conference on machine learning (pp. 3481 − 3490, 2018.

[42] F. Alonso-Fernandez, K. Hernandez-Diaz, S. Ramis, F.J. Perales, J. Bigun, Facial masks and soft-biometrics: Leveraging face recognition CNNs for age and gender prediction on mobile ocular images, IET Biom. 10 (2021) 562-580. https://doi.org/10.1049/bme2.12046.

[43] Y. Guo, L. Zhang, Y. Hu, X. He, J. Gao, MS-celeb-1M: A dataset and benchmark for large-scale face recognition, in: Computer Vision - ECCV 2016, Springer International Publishing, Cham, 2016: pp. 87-102.

[44] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, ”Labeled faces in the wild: A database for studying face recognition in unconstrained environments,” Technical report, 2007.

[45] S. Sengupta, J.-C. Chen, C. Castillo, V.M. Patel, R. Chellappa, D.W. Jacobs, Frontal to profile face verification in the wild, in: 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2016.

[46] S. Moschoglou, A. Papaioannou, C. Sagonas, J. Deng, I. Kotsia, S. Zafeiriou, AgeDB: The first manually collected, in-the-wild age database, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, 2017.

[47] T. Karras, T. Aila, S. Laine, J. Lehtinen, Progressive growing of GANs for improved quality, stability, and variation, arXiv [Cs.NE]. (2017).

[48] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2016.

[49] Z.-H. Feng, J. Kittler, M. Awais, P. Huber, X.-J. Wu, Wing loss for robust facial landmark localisation with convolutional neural networks, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, 2018.

[50] J. Johnson, A. Alahi, L. Fei-Fei, Perceptual losses for real-time style transfer and super-resolution, in: Computer Vision - ECCV 2016, Springer International Publishing, Cham, 2016: pp. 694-711.

[51] Y. Zhou, J. Gregson, WHENet: Real-time fine-grained estimation for wide range head pose, arXiv [Cs.CV]. (2020). http://arxiv.org/abs/2005.10353.

[52] A. Bulat, G. Tzimiropoulos, How far are we from solving the 2D & 3D face alignment problem? (and a dataset of 230,000 3D facial landmarks), in: 2017 IEEE International Conference on Computer Vision (ICCV), IEEE, 2017.

[53] N.H. Lestriandoko, R. Veldhuis, and L. Spreeuwers, The contribution of different face parts to deep face recognition, Frontiers in Computer Science, 2022, p. 89.

[54] P. Tome, J. Fierrez, R. Vera-Rodriguez, J. Ortega-Garcia, Combination of face regions in forensic scenarios, J. Forensic Sci. 60 (2015) $1046 - 1051$. https://doi.org/10.1111/1556-4029.12800.

Chapter 6

Differential Privacy Preservation in Robust Continual Learning

Ahmad Hassanpour, Majid Moradikia, Bian Yang, Ahmed Abdelhadi, Christoph Busch, Julian Fierrez

Abstract

Enhancing the privacy of machine learning (ML) algorithms has become crucial with the presence of different types of attacks on AI applications. Continual learning (CL) is a branch of ML with the aim of learning a set of knowledge sequentially and continuously from a data stream. On the other hand, differential privacy (DP) has been extensively used to enhance the privacy of deep learning (DL) models. However, the task of adding DP to CL would be challenging, because on one hand the DP intrinsically adds some noise that reduce the utility, on the other hand the endless learning procedure of CL is a serious obstacle, resulting in the catastrophic forgetting (CF) of previous samples of ongoing stream. To be able to add DP to CL, we have proposed a methodology by which we cannot only strike a tradeoff between privacy and utility, but also mitigate the CF. The proposed solution presents a set of key features: (1) it guarantees theoretical privacy bounds via enforcing the DP principle; (2) we further incorporate a robust procedure into the proposed DP-CL scheme to hinder the CF; and (3) most importantly, it achieves practical continuous training for a CL process without running out of the available privacy budget. Through extensive empirical evaluation on benchmark datasets and analyses, we validate the efficacy of the proposed solution.

INDEX TERMS Differential privacy, continual learning, deep learning, privacy.

6.1 Introduction

Recently, deep learning (DL) models have shown significant improvement as compared to the human decision making on different tasks [1]-[5]. Despite the striking results, since DL models are built upon the static models, they cannot be applied simply over data streams. More explicitly, a time frame of data stream may vanish soon due to storage constraints or privacy issues, which requires a dynamic training process to begin upon receiving the new data. This gap motivates the researchers to develop DL models, able to adapt frequently and resume learning over time. A typical example of such a system is human cognition by which one tends to learn concepts sequentially. One prominent feature of such a system is that old concepts might be revisited though it is not necessary to keep them in mind [6]. By contrast, conventional DL models cannot learn in this way and thus they suffer from catastrophic forgetting (CF) of old concepts upon learning new ones [7]. Hence, conventional DL (CDL) models often concentrate on static tasks whose data are shuffled to guarantee the independent and identically distributed (i.i.d.) requirement. Despite performance improvement, CDL models cannot be applied over data streams as the training data is revisited over several computations. To circumvent this issue while preventing the CF, described above, Continual Learning (CL) comes into play, aimed at gradually extending attained information to be exploited for future learning.

In real world, DL algorithms are extensively vulnerable to security attacks e.g., adversarial example where an adversary fool the DL via perturbation samples [8], [9]. Based on the knowledge of adversaries from the target model, the adversarial attacks belong to one of the main group of: white-box, gray-box, and black-box attacks. In blackbox attack model, the attacker is not able to access to the model weights; whilst in the white-box attack, the attacker has completely access to the architecture and weights of the model, comprised of countermeasure methods.

Gray-box attacks also presume that the attacker knows everything about the network and defense, except the parameters.

To confront with such attacks, three well known methods have been broadly used in several literature: fully homomorphic encryption (HE) [10], [11], [37], k-anonymity [12], and differential privacy (DP). Although the HE offers strong data privacy preservation, it is ineffective in DL models owing to the computational burden imposed due to the dimension of training datasets. On the other hand, k-anonymity also performs weakly when facing large datasets [13], [14]. Thus, both HE and k-anonymity are inefficient in case of data stream in which a large amount of data is coming in over a long period of time and it is not practically possible to keep the entire data set in memory at once.

Recently, DP has attracted a great deal of attention in DL-based solutions due to providing the capability of analyzing a dataset without disclosure of an individual's information for DL models [17]. The main goal of such a system is to control the cost of losing privacy, called privacy budget (PB), so that it should not exceed the predefined global privacy budget (GPB). Notably, without adding computational burden, it tries to preserve the privacy of data by perturbing the weights, objective function, or outputs of DL models systematically [15], [38]. The noise added to the dataset will affect the privacy-utility trade-off. Explicitly, upon increasing the amount of noise the dataset would be useless, while reducing the noise up to the little values will degrade the privacy. Concerning using DP in DL models, a differentially private version of the SGD algorithm, is proposed in [16], where the amount of random noise and the privacy budget (PB) constantly increase upon growing the number of training epochs which is in contrast to the limited PB in practice. Dwork and Roth [15] proposed a method for incorporating DP into distributed DL. They designed a practical framework that allows multiple clients to collaboratively train a DL model without sharing their training data.

To the best of our knowledge, despite the applicability of DP in DL models (DP-DL) [16], [17] and stream data [18] separately, there is no study on adding DP into CL models such that all characteristics of a CL process meet, so far. However, this task would be challenging, because on the one hand the DP intrinsically adds some noise that reduce the utility, and on the other hand the endless learning procedure of CL is a serious impediment. Thus, to compromise between privacy and utility in the proposed DP-CL, we need to rethink and redesign the existing DP-DL models to be adapted for the CL process. To elaborate further, on the one hand difficulties arise from two significant characteristics of the CL process as follows:

C1) The learner used in the CL process should be able to learn the new received data continuously and endlessly.

C2) To mitigate the CF, a small portion of data or model's parameters needs to be stored for future learner's computations.

On the other hand, a DP-enabled algorithm has two significant limitations as follows:

L1) Each computation of the DL algorithm not only increases the bound over data leakage, but also consumes a portion of predefined privacy budget (PB). Although it is desired that the leakage bound does not exceed the available PB, it has been shown in [16], [17] that a DL process run out of the PB after a few computations.

L2) DP tends to perturb the data or the algorithm's parameters by adding noise, leading to diminishing the utility.

In our proposed approach where we aim to add DP into CL, we encounter the following issues:

I1) the **L1** is in contrast to the **C1**, as the available PB is limited, preventing the CL process to be continued endlessly.

I2) Moreover, lowering the utility mentioned in L2 exacerbates the detrimental impact of CF described in C2, which motivates us to look for a robust design.

In this paper, we proposed a novel robust DP-CL approach by which we tackle these issues effectively. To the best of our knowledge, this is the first paper which studies the integration of DP into CL by addressing I1 and I2, concurrently. Against this background, our contributions and novelties can be summarized as follows:

- To address I1, (or more explicitly to be able to continue the training process endlessly without running out of the PB), at each iteration of the training process, the spent PB is measured for each training sample and learner. Once the resultant PB is being exceeded to the predefined GPB, the previous samples in the temporary memory are replaced by new zero-PB ones, coming from the data stream. Similarly, we will do the same approach to substitute the previous learner with a new zero-PB one.

- To overcome **I2** (or more explicitly to combat the CF), we further incorporate a robust procedure into the proposed DP-CL scheme, including three steps 1) adding a new noisy layer to the DL architecture, 2) refining the CL algorithm's objective function (OF), and 3) filling the episodic memory (EM) more effectively. We will detail throughout the paper that how each of these steps can help to increase the robustness of our proposed algorithm. We will experimentally show that each of these steps can assist to make the DP-CL process more robust against white-box attacks.

- To evaluate the effectiveness of the proposed robust DP-enabled CL process, different adversarial attacks have been used to fool the trained models. Particularly four types of white box attacks have been used including: 1) Fast Gradient Sign Method (FGSM) 29], 2) Iterative-FGSM (I-FGSM) [30], 3) Momentum Iterative Method (MIM) [28], and 4) the attack proposed by Madry et al. [29]. Our simulation results confirm that the proposed method yields the stable and steady outputs, even when facing of such strong attacks.

The rest of the paper is organized as follows. Recent works in the context of using DP in machine learning algorithms are reviewed in Section II. A brief description of CL models, DP, and adversarial attacks are presented in Section III as a preliminary. A detailed description of the proposed methodology is provided in Section IV. The experimental results and discussions are reported in Sections V and VI. Finally, Section VII presents the conclusion

6.2 Related works

So far, several papers have attempted to add DP to DL algorithms [16], [19]-[21]. This task would be challenging in terms of limited PB and the privacy-utility tradeoff requirement. Upon DL models progresses, for example when we aim to apply DP on those DL models using dynamic dataset, some other demands will ensue which exacerbate the abovementioned issues. Some of the most prominent demands, which are close in spirit to the requirements of CL, as we need here, are listed as follows:

R1: Endless execution

R2: Multiple usage of data subsets

R3: Capability of changing DP parameters during the execution

Satisfying all the R1-R3 together is hard, therefore related papers address only one or two of these requirements. Along this line, two recent DL-based papers of [22], [23] have enabled DP to work on growing databases (dynamic datasets). More explicitly, to address **R**1 Cummings et al. have considered a scheduler to re-execute the DL algorithms whenever the new received data is sufficient [23]. To achieve the desired privacy loss, the privacy parameter (ϵ) is reduced upon increasing the size of dataset.

In order to jointly address **R1** and **R3**, one can partition the data stream into blocks. After applying the DP on data blocks, each of which is fed into an individual learner, the learners' outputs are aggregated [24]. Accordingly, the conventional composition theorem can be exploited to calculate the privacy loss at the block level. Now, deploying the conventional composition theorem, the data blocks incur no privacy loss from the previous learners and thus the requirements of **R1** and **R3** are supported. However, it is against **R2** as each learner cannot access other learners' blocks.

In another scenario, aimed at addressing **R2** and R3, Lecuyer et al. have proposed a DP-DL platform including several pipelines, each of which comprised a DL algorithm, training endlessly from the growing database. Note that, since each block of data might be used by different DL algorithms corresponding to the pipelines, calculating the PB spent by the whole pipelines would be challenging. To reach this goal, the authors of [22], have proposed the so-called block composition theorem by which the DL algorithms are executed till the PB consumption of each block [1] does not exceed the predefined GPB. To achieve the desired accuracy, with the aim of re-training the pipelines, either the relevant PB of each pipeline or the number of available samples is doubled. Therefore, each pipeline can continue till the consumed PB is smaller than GPB, violating **R1**.

6.3 Preliminaries

6.3.1 Continual learning

A typical CL process, e.g., A-GEM [25], has generally two important features. First, the used learner in the CL process should be able to learn the new received data continuously and endlessly (growing database). In other

[01] We can interchangeably use the word of ″block″ and ″sample″ throughout the paper.

words, the commonly used CL model can be fed by consecutive parts of a data stream, each of these parts includes multiple number of samples and corresponds to a particular task. Second, a small part of data will be stored model's parameters or training data for future learner's calculations to prevent catastrophic forgetting. Thus, CL refers to the ability of a system to learn over time from a continuous stream of data without having to revisit previously encountered training samples.

First, the i th sample of the training set D includes a triplet (x_i, t_i, y_i), where $x_i \in X_t$ is a feature vector, $t_i \in T$ is a task descriptor, and $y_i \in Y$ is a target vector. In general, CL algorithms aim to learn a predictor $f_\theta : X \times T \rightarrow Y$ in which θ denotes the relevant tunable parameters of predictor f.

To get more insight, in the following we succinctly explain A-GEM [25]. Using the A-GEM algorithm, the detrimental impact of catastrophic forgetting can be alleviated by allocating an episodic memory (EM), which is denoted by M and equally divided between total T tasks, to store some training samples randomly for each task k. These stored samples assist the DL model to maintain its performance for previous tasks. For a total number of T tasks, if we let D_k represents the relevant data with respect to previous tasks, i.e., $k \leq T$, the abovementioned explanations can be mathematically formulated as the following constrained optimization problem

$$\min_\theta \mathcal{L}_{AG}\left(f_\theta, D_t\right), \text{ s.t. } \mathcal{L}_{AG}\left(f_\theta, M_k\right) \leq \mathcal{L}_{AG}\left(f_\theta^{t-1}, M_k\right) \forall k < t \tag{1}$$

where the objective function $\mathcal{L}_{AG}\left(f_\theta, D_t\right)$ stands for the loss of the A-GEM model on the current task t. Using the stored data of previous tasks in $\mathrm{EM}\left(M_k\right)$, the constraint attempts to reduce the loss of the model with respect to the loss of previous tasks.

6.3.2 Differential privacy

The DP technique prevents the disclosure of information corresponding to individual records of database D against any adversarial processing. Using DP, the records are contaminated with noise through a randomized algorithm $A : B \rightarrow R$. The DP is often characterized by the parameters (ϵ, δ) where the privacy budget $(\mathrm{PB})\epsilon > 0$ and the broken probability $\delta \in [0, 1]$ are control parameters to tune the strength of the privacy preservation. Thus, given the randomized algorithm A, the following inequality must hold true to satisfy the (ϵ, δ)-DP:

$$P[A(D) \in O] \leq e^\epsilon P\left[A\left(D'\right) \in O\right] + \delta \tag{2}$$

where $\{D, D'\} \in B$ are two neighboring inputs and $O \subseteq R$ represents any subsets of outputs. Besides, $P[\cdot]$ denotes the probability function with the space over the coin flips of the algorithm A. The Eq. (2) implies that if we change a tuple in the database slightly, the output distribution does not vary significantly.

Now, in the following we invoke the definitions of some basic concepts used in DP, which lay the grounds for a better understanding.

1. Privacy loss [15]: Privacy loss is a random variable dependent on the random noise added to the algorithm. For neighboring databases D, D', auxiliary input aux, and an outcome $O \subseteq R$, define the privacy loss at O is defined as:

$$c\left(O; A, aux, D, D'\right) = \frac{P[A(aux, D) = O]}{P\left[A\left(aux, D'\right) = O\right]} \tag{3}$$

2. Gaussian mechanism [15]: This mechanism will be used in this paper. Using this kind of mechanism the white Gaussian noise $\mathcal{N}\left(0, \sigma^2\right)$ is added to the output entries. Given $\epsilon \in (0, 1]$, the Gaussian mechanism with $\sigma \geq \sqrt{2 \ln\left(\frac{1.25}{\delta}\right)} \cdot \frac{\Delta_A}{\epsilon}$ is $(\epsilon, \delta)-$ DP and the l_2 sensitivity parameter Δ_A therein is defined as $\Delta_A = \max_{D,D'} \left\|A(D) - A\left(D'\right)\right\|_2$.
3. Composition theorem: If we consider several DP subroutines, each of which applied into separate algorithms to reach a specified privacy level, incorporation of these DP subroutines relying on the composition property significantly degrades the privacy such that it is less than that of achieved by a single subroutine. In particular, based on one kind of composition theorem, namely "basic composition theorem" [26], considering ℓ subroutines each of which is $(\epsilon, 0)$ differentially private, the privacy of an algorithm including a combination of these subroutines is degraded up to the bound of $(\epsilon\ell, 0)$ as compared to the single subroutine.

6.3.3 Adverserial examples

Adversarial examples are a kind of attack against ML models, where the attacker add a small perturbation $\alpha \triangleq \{a_i\}_{i=1}^{I} \in R^I$ to the given input $x \triangleq \{x_i\}_{i=1}^{I} \in R^I$ of the DL model, resulting in a considerable change at the output $y \triangleq \{y_i\}_{i=1}^{c} \in R^c$. The perturbation is usually specified by a l_{p^-} norm ball of radius μ, i.e., $\Theta_\mu \triangleq \{\alpha : \|\alpha\|_p \leq \mu\}$ where $p \in \{1, 2, \infty\}$ [27]. To evaluate the robustness of the proposed method, particularly four well-known white box [2] attack algorithms are utilized to generate the adversarial samples: i) Fast Gradient Sign Method (FGSM) [27], ii) Iterative-FGSM (I-FGSM) [30], iii) Momentum Iterative Method (MIM) [28], and iv) the attack proposed by Madry et al., [29], are utilized to generate the adversarial samples.

6.4 Proposed robust dp-enabled continual learning

In this section we present the notion of adding DP to A-GEM algorithm and then make the proposed DP-CL model robust. Thus, by considering the characteristics of CL processes (i.e., $\mathbf{C1}$ and C2) and created limitations by DP (i.e., L1 and L2), we address the A-GEM requirements and finally propose a scheme for a DP-enabled CL process the during the next subsections (i.e., 4.1 and 4.2). Then, to overcome catastrophic forgetting and reduce the impact of attacks, in subsection 4.3 we add robustness methods to DP-CL: 1) modifying the DL architecture, 2) refining the objective function (OF) of the A-GEM algorithm, and 3) filling the EM more effectively.

6.4.1 Adding DP to CL process

Given the properties of DP, as discussed above, the problem of adding DP to CL would be challenging. First, adding perturbations to the learner(s) will effect on the training accuracy and consequently worsen CF. Moreover, the composition theorem, imposes some predefined bounds for DP algorithms, including the number of subroutines (iterations (k)) and privacy parameters (ϵ, δ). As per requirements of a CL process these variables need to be updated and thus a CL-based composition theory must satisfy the three following requirements:

R1: Endless execution

R2: Handling the concern of overlapping data stored in EM

R3: Capability of updating DP parameters during the execution

Hence, it is required to think about how to satisfy each of $\mathbf{R1} - \mathbf{R3}$ which are responded to, in the sequel.

1. How to add DP while CL is executed endlessly (Addressing R1.)?

The everlasting approach of CL is a serious impediment to deploy either of the proposed solutions in [22] or [23]. In particular, if one intends to add DP to CL, the limited GPB hinders the process to be continued. To deal with these problems, we here propose a novel learning procedure, comprised of several learners in $L \triangleq \{l_1, l_2, \ldots\}$, each of which is trained sequentially on a specific part of the data stream. Before exceeding the PB consumed by each learner from the GPB, we add a zero-PB (ZPB) learner to the process. This newly added learner starts from the point where the previous one has been halted and would be continued using the untouched data coming from the dynamic database $S = \{b_1, b_2, \ldots\}$ (and/or the stored data in EM $M = \{b_i, b_j, \ldots\}$, where b_i shows the i th block of database).

Based on the discussion above, selecting an appropriate composition theorem is of vital importance to calculate PB for each training step through

which, we can determine the halting time of the current learner (l_c), learning the current task (t_c). We here use the moments accountant algorithm (MAA) [16], appropriate for computing the PB for each data access in the DL models. When l_c runs out of the PB, computed by MAA, this learner is left out and added to
the set of trained learners L, i.e., $L \triangleq \{l_1, l_2, \ldots, l_{c-1}, l_c\}$ and the learning process will be continued via the next ZPB learner l_{c+1}. There are some technical concerns which must be considered in our design, listed as follows:

- The significance of GPB parameter values (ϵ_g, δ_g) : More explicitly, a large selection of the GPB leads to higher privacy leakage, despite yielding higher accuracy due to injecting less noise into the current learner l_c as well as using fewer number of learners for the whole process. In contrast, although upon reducing the GPB the leakage is decreased, the accuracy is degraded, as well. The performance degradation originates from the fact that, using small GPB values not only more noise is fed into the current learner l_c, but also more number of learners must be deployed.

- Keeping the performance while deploying multiple learners: In the case the PB of l_c reaches to the GPB in the middle of learning t_c, leading to degrading the performance of upcoming learner l_{c+1}, we proposed early starting ES) strategy that assists to predict the termination of l_c. More clearly, the random initial values of learning parameters θ_{c+1} which are going to be used by l_{c+1} have not been optimized for the current task t_c. To prevent this issue, we propose the ES strategy where the remaining PB, i.e., $\left(PB_r \triangleq GPB - PB_{l_c}\right)$ of the l_c is compared with the required PB of $t_{c+1}\,(PB_{c+1})$, and the l_c continues if and only if $PB_r > PB_{c+1}$. To estimate PB_{c+1}, since the noise magnitude and the sampling probability (Gaussian probability) is equal during the training process of each learner, it is trivial to calculate the consumed PB of next iterations or the required PB for the next task (i.e., $PB_c = PB_{c+1}$). Doing this, l_c will not be halted in the middle of training a task, and each learner starts its training procedure from the beginning of a task.

2. How to add DP while subsets of data are used repeatedly (Addressing R2.)?

A serious impediment to deploy either of the proposed solutions in [22] or [23] in a CL process, is the data coming from the stream as well as samples stored in the EM to avoid catastrophic forgetting (CF). Note that, although the learners observe most of the data coming from the stream just once, a small portion stored in the EM is observed several times. For each observation the corresponding learner consumes the PBs associated with a portion of the sample stored in EM. Thus, if the spent PB of each stored sample in EM (PB_{b_i}) exceeds the GPB, the privacy is compromised. In the following, we elaborate this further.

The samples in EM that have been observed repeatedly, might be observed in different iterations of the learners' training process. Depending on the privacy loss of the l_c used at each iteration, a portion of the sample's PB will be consumed and can be stored in $PB_{b_i} = \left\{PB_{m_i,l_k}, \ldots, PB_{m_j,l_h}\right\}$, where PB_{m_i,l_k} stands for the consumed PB of i th iteration of the learner l_k. By doing so, we can calculate the total consumed
PB for the sample via feeding PB_{b_i} to the Block Composition Theorem (BCT) [22]. Tracking the behavior of PB_{b_i}, if it exceeds the GPB, we no longer use that sample in our CL procedure

Remarkably, to avoid the CF, EM should include some samples for each task. Thus, ZPB samples will be randomly replaced from the stream with ones that are removed at each iteration. We also proposed other different strategies for replacing new samples described in subsection 4.3 (c) to make the DP-CL process more robust. By following this strategy, we can use a subset of data (those stored in EM and their consumed PB is less than GPB) multiple times. Therefore, since there is a limitless of data in real world CL scenarios, the halting of l_c will not occur because of limitation in data PB.

3. Adaptivity in the choice of DP parameters during the CL process. (Addressing $R3$.)

To address the privacy-utility tradeoff, the proposed DPCL process benefits from an adaptive training procedure such that controls the utility of DP-CL models by using new data and/or changing DP parameters. The block composition theorem allows us to train the used CL algorithms with different PB. For those tasks that have high number of samples in their training set, we will be able to adjust small PB leading to decrease privacy leakage and vice versa. If a model does not reach the pre-defined quality criteria (e.g., an accuracy target) until specific iteration and $PB < GPB$, the model can decrease the added noise (σ) to its weights, results in expediting increasing accuracy, although PB reaches GPB earlier. On the other hand, if a model reaches the pre-defined quality criteria in a specific iteration and $PB < GPB$, then the model can increase the added noise to its weights to increase the privacy of the model.

6.4.2 DP-CL architecture

The proposed (ϵ_g, δ_g)-DP-CL Architecture includes three main modules called Learners' Managing Unit (LMU), Privacy Meter Unit (PMU), and Data Man-

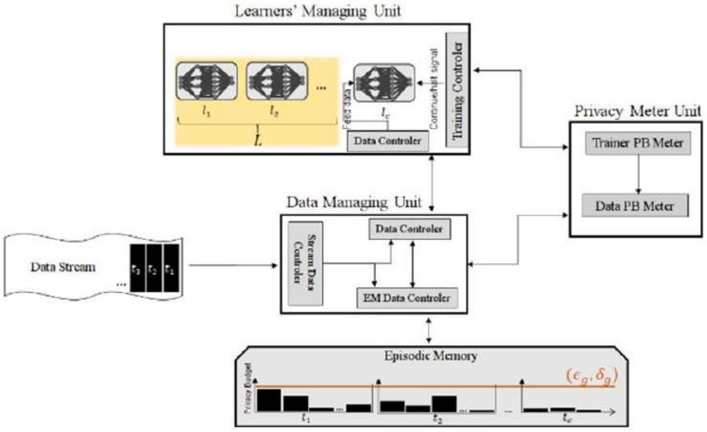


Figure 6.1: The proposed DP-CL architecture.

aging unit (DMU). The detailed procedure of our proposed DP-CL method is shown in Algorithm 1, and is conceptually described in the following.

The LMU is composed of two sub-modules, called Training Controller (TC) and Data Controller (DC). The TC is responsible for adding new learners to the process, adjusting the l_c parameters, saving the l_c 's parameters, and collecting the information about the tasks corresponding to each learner. Moreover, TC also receives the information related to halting time of a learner from the PMU. Besides, the DC also receives the training data from the DMU and feed them to the l_c. Additionally, the DC specifies which samples should be saved in the EM and send them to the DMU.

The PMU is responsible for measuring the spent PB of learners and training samples respectively by two submodules of Trainer PB Meter Unit (TPBMU) and Data PB Meter Unit (DPBMU). For each training iteration of l_c, the spent PB will be calculated by TPBMU so that if it exceeds

the GPB, the TC will be notified to halt the l_c. The DPBMU calculates the spent PB for those samples, used in the current iteration of l_c and send this information to the DMU. It should be noted that, the PB for all samples will be stored in a submodule of DMU namely EM Data Controller (EDC), since we may need to remove some samples from EM and replace them by samples whose spent PB is less than GPB.

The DMU is responsible for managing the data and is composed of three sub-modules of Data Controller (DC), EDC, and Stream Data Controller (SDC). The DC fetches the data from the stream or EM by sending a request to EDC or SDC. It also collects the spent PB of the training samples stored at EM or coming from the stream. SDC also stores the received data from the stream into a temporary database. Upon receiving a request from DC or EDC, the SDC will deliver the requested data to those modules. The EDC is responsible for adding/removing the samples having spent PB more than GPB. When the privacy loss for a sample reaches to GPB, the sample will be removed.

6.4.3 Adding robustness to DP-CL

To combat the CF and mitigate the effect of attacks, we incorporate a robust procedure into the proposed DPCL scheme, including three steps 1) modifying the DL architecture, 2) refining the OF of the A-GEM algorithm, and 3) filling the EM more effectively. In what follows we elaborate each of these steps separately. The first method has the aim of reducing the attacks success rate by making the CL parameters noisy, and the other two methods assist to prevent CF. However, our experiments show that the last two proposed methods can also decrease the attacks success rates to some extent

6.4.3.1 Modifying the DP-CL architecture

To provide a robust DP-CL architecture, we change each learner's architecture by adding a DP noise layer, that provide (ϵ, δ)-DP guarantees, after the first layer of each learner. Adding the DP noisy layer can be considered as a certified defense against p-norm bounded adversarial example attacks proved by [31]. More explicitly, in accordance with the sensitivity (Δ) and size of the first layer $(|h_1|)$, a noise with zero mean and standard deviation $\sigma = \sqrt{2\ln\left(\frac{1.25}{\delta}\right)}\Delta_{p,2}L/\epsilon$ is produced by Gaussian mechanism (noise $(\Delta, L, \epsilon, \delta$), line 6 , Algorithm 2).

6.4.3.2 Refining the objective function of the A-GEM algorithm

Furthermore, to prevent CF, we incorporate a robustness condition into the training stage (called robust-A-GEM hereafter). In this regard, it should be noted that, the expected output of the randomization mechanism A for class j during the training of current task t should be greater or equal to that of the previous task, i.e., $\mathbb{E}_t\left(A_j(x)\right) - \mathbb{E}_{t-1}\left(A_j(x)\right) \geq 0$ where $\mathbb{E}_t\left(A_j(x)\right) = \frac{1}{n}\sum_n a_{j,n}(x)$ and n denotes the number of invocation of $A(x)$ and $a_{j,n}(x)$ demonstrates the n^{th} draw from the distribution of the randomized function A on the j^{th} label. To meet this condition, the computed angle between the gradient of l_c for t_c with respect to label $j\,(\tilde{g}_j)$ and the gradient of l_c for the previous tasks $(\forall k < t)$ for label $j\,(g_{j,k})$ should be greater than zero $(\langle \tilde{g}_j, g_{j,k}\rangle \geq 0)$.

Moreover, instead of n times invoking $A(x)$ for a specific sample x, to calculate $\mathbb{E}_t\left(A_j(x)\right)$, we use n samples belonging to the j^{th} class within the current batch, and for calculating $\mathbb{E}_{t-1}\left(A_j(x)\right), n$ samples having label j

Algorithm 6.1: Proposed DP-CL.

```
Procedure DP-CL:
// Learners' Managing Unit (LMU):
While $b_{i,t_i} \leftarrow$ ask data from $DMU$
If $l_c = \emptyset$ then
Initialize $l_c$
train $l_c$ with $b_{i,t_i}$
$SPB_{l_c} \leftarrow PMU\,(l_c, b_{i,t_i})$
DMU $(b_{i,t_i}, SPB_{l_c})$ // save part of $b_{i,t_i}$ in EM
IfSPB $l_l \geq GPB$ then
$C \leftarrow [l_c, (t_j, \ldots, t_i)]$
$l_c \leftarrow$ Initialize a new learner
//Privacy Meter Unit (PMU):
If $l_c, b_{i,t_i} \leftarrow$ receive request from $LMU$ then
$LMU \leftarrow$ Calculate the spent PB of $l_c$ using MAA
$DMU \leftarrow$ Calculate the spent PB of each training sample in $b_{i,t_i}$
  using BCT
I/Data Managing Unit (DMU):
If receive data from Stream then
Save the data temporary
If receive request from $LMU$ then
Fetch a batch of data from EM/Stream and send to $LMU$
If receive data from $LMU$ then
Store data in EM
Store spent PB for each $b_i$
Update EM by replacing those samples which run out their PB with
  ZPB samples (Procedure UpdateEpsMem, Alg.2)
end procedure
```

are chosen from the EM. This notion assists to incorporate this condition to the training process by changing the constraint A-GEM objective function. Therefore, we modify the optimization function as below:

$$\min_{\tilde{g}} \frac{1}{2} \|g - \tilde{g}_j\|_2^2 \text{ s.t. } \langle \tilde{g}_j, g_{j,k} \rangle \geq 0 \forall \quad k < t \tag{4}$$

where $g_{j,k}$ will be the average gradient from the previous tasks with respect to j th class. By doing that, the new updated rule will be obtained as follows:

$$\tilde{g} \leftarrow g - \frac{g^\top g_{j,ref}}{g_{j,\text{ ref}}^\top g_{j,ref}} g_{j,\text{ ref}} \tag{5}$$

The proof of this update rule is given in Appendix A.

6.4.3.3 Filling the EM efficiently

The easy-to-forget samples (which worsen CF) which classify correctly with small robust boundary during the training process have a chance to enter EM. Therefore, having such samples which are not a good representative of their corresponding classes in EM leads to CF during the learning of next tasks. Particularly this issue happens if the computed angle between the gradient vector of the samples extracted for class j from EM $(\tilde{g}_j)$ and the proposed gradient (g) at current iteration is larger than zero. Here, we propose a robustness condition by which that if a sample meets this condition, then it will be added to the EM (called efficient-EM). For sample x_z located in a batch including n samples, the robustness condition calculated as follow:

$$\begin{aligned} &\mathbb{E}_t^{lb}\left(f_j\left(x_z\right)-\max_{i:i\neq j}\mathbb{E}_t^{ub}\left(f_j\left(x_z\right)\right)\right. \\ &\geq \frac{1}{1+e^{\frac{\sum_{s=1}^{n}\mathbb{E}^{lb}\left(f_j\left(x_s\right)-\max_{i:i\neq j}\mathbb{E}^{\mathrm{ub}}\left(f_j\left(x_s\right)\right)\right)}{n}}} \end{aligned} \tag{6}$$

$\mathbb{E}_t^{lb}\left(f_j\left(x_z\right)\right)$ and $\mathbb{E}_t^{ub}\left(f_j\left(x_z\right)\right)$ are the η-confidence lower and upper bound, respectively. We estimate these bounds using Hoeffding's inequality with probability $\eta, \mathbb{E}^{lb}(f(x)) \triangleq \mathbb{E}(f(x)) - \sqrt{\frac{1}{2n}\ln\left(\frac{2y}{1-\eta}\right)} \leq E(f(x)) \leq \mathbb{E}(f(x)) + \sqrt{\frac{1}{2n}\ln\left(\frac{2y}{1-\eta}\right)} \triangleq \mathbb{E}^{ub}(f(x))$ for y^{th} label (Lines 26, 33, Algorithm 2).

6.4.3.4 The proposed robust dp-cl algorithm

The proposed robust DP-CL algorithm (shown in Algorithm 2) includes three procedures called Train, UpdateEpsMem, and Evaluation. The Train procedure takes the train and test data, as well as the l_c 's parameters. Considering the size of first hidden layer, a generated random Gaussian noise (line 3), is added to the first hidden layer (line 6). By wisely sampling from the EM (considering the notion presented at section 4.3 b; line 7), the gradient for the current batch (line 9) and the sampled batch (line 8) have been calculated. Then, g and g_{ref} are clipped so that its 12-norm is bounded by a predefined gradient clipping bound $\mathcal{C}$ and subsequently, a random Gaussian noise $N\left(0, \sigma^2\mathcal{C}^2 I\right)$ with a predefined noise scale σ is added (Line, 11 and 12). Depending on the computed angle between g' and g'_{ref}, the new gradient will be applied (lines 13-18). After feeding each batch and updating the l_c, the EM will be updated by executing the UpdateEpsMem procedure. During this procedure, we first replace the samples that run out their

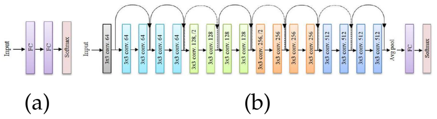


Figure 6.2: a) Fully connected network with two hidden layers used for PMNIST dataset. b) Reduced ResNet18 for SCIFAR dataset.

PB with ZPB ones from $D^{train}_{y_i,t_i}$. Then some samples from the current task which meet the proposed robustness condition (presented at section 4.3c) will be added to the EM. Finally, the Evaluation procedure measures the effectiveness of the training procedure by calculating the accuracy.

6.5 Evaluation

We have carried out extensive experiments on two benchmark datasets (permuted MNIST and split CIFAR) and evaluate our proposed robust DP-CL process by answering the following questions:

Q1: How does the added DP mechanism affect the accuracy of the A-GEM algorithm?

Q2: What is the impact of using several learners on the accuracy of the DP-CL process?

Q3: How can the ES deal with the performance degradation in the training process?

Q4: How the proposed robust DP-CL acts against attacks?

Q5: How much data the DP-CL process will need?

Before answering these questions, we will briefly describe the used datasets description, the used DL architectures, the evaluation metrics, and observe the behavior analysis of DP's parameters in the following subsections.

6.5.1 Dataset description

Two datasets have been considered to train and test the proposed robust DP-enabled CL process. First, Permuted MNIST (PMNIST) [32] is a variant of MNIST dataset including handwritten digits. It consists of 20 tasks each of which is composed of 10 classes, 60,000 training and 10,000 test samples. Each task describes a certain random permutation of the input pixels, applied to the entire images of that task. Split CIFAR (SCIFAR) [33] devides of dividing the original CIFAR-100 dataset [34] into 20 disjoint subsets, each of which is generated through random sampling of 5 classes without replacement from the total number of 100 classes. The whole number of training samples for each task is 2500 whose 20% are allocated for testing. In general, there are two streams of tasks, described by the sequences

Algorithm 6.2: Robust DP-CL.

```
Procedure Train $\left(f_\theta, D^{\text{train}}, D^{\text{test}}\right)$ Input: Datasets $D^{\text{train}}$ and $D^{\text{test}}$,
  batch size $m$, learning rate for each task $\zeta_t$, gradient norm band $C$,
  privacy budget $\epsilon$, broken probability $\delta$, robustness parameters:
  $\sigma_r, \epsilon_r, \delta_r, \Delta_r$, size of first hidden layer $|h_1|$, $f$ includes $z$ hidden
  layers $\{h_1, \ldots, h_z\}$, EM depicted by $M$.
    Initialize $\theta$ randomly
    $\gamma \leftarrow N\left(0, \sigma^2 |h_1|\right)$
    for $t = \{1, \ldots, T\}$ do
        for $(x, y) \in D_t^{\text{train}}$ do
            $h_1 \leftarrow W_1^T x + \gamma$
            $(x_{ref}, y_{ref}) \sim M(y)$
            $g_{\text{ref}} \leftarrow \nabla_\theta l\left(f_\theta\left(x_{\text{ref}}, t\right), y_{\text{ref}}\right)$
            $g \leftarrow \nabla_\theta l\left(f_\theta(x, t), y\right)$
            Clipping gradient and adding noise
            $g'_{\text{ref}} \leftarrow \frac{1}{m}\left(\frac{g_{\text{ref}}}{\max\left(1, \frac{\|r_{\text{ref}}\|}{C}\right)} + N\left(0, \sigma^2 C^2 I\right)\right)$
            $g' \leftarrow \frac{1}{m}\left(\frac{g_{ref}}{\max\left(1, \frac{\|g\|}{C}\right)} + N\left(0, \sigma^2 C^2 I\right)\right)$
            If $g' g'_{\text{ref}} \geq 0$ then
                $\tilde{g} \leftarrow g'$
            else
                $\tilde{g} \leftarrow g' - \frac{g'^{\top} g'_{j,\text{ ref}}}{g'_{j,\text{ ref}} g'_{j,ref}} g_{j,\text{ ref}}$
            end if
            $\theta \leftarrow \theta - \zeta_t \tilde{g}$
        end for
        UpdateEpsMem $\left(M, D_t^{\text{train}}, T\right)$
    end for
end procedure
Procedure UpdateEpsMem $\left(M, D_t^{\text{train}}, T, GPB\right)$
    // remove stored samples in $|M|$ with high PB
    for $i = \{1, \ldots, |M|\}$ do
        if $spent_P B\ B_i > GPB$ do
            remove $(x, y_i, t_i) \leftarrow M_i$
            $(x) \leftarrow D_{y_i, t_i}^{\text{train}}$ which meet robustness condition
            $M_i \leftarrow (x)$
    end for
    // Add a few samples from current task
    $s \leftarrow \frac{|M|}{T}$
    for $i = \{1, \ldots, s\}$ do
        $(x, y) \leftarrow D_t^{\text{train}}$
        If $(x, y)$ meet the robustness condition then
            $M \leftarrow (x, y)$
    end for
    return $M$
end procedure
Procedure Evaluation $\left(f_\theta, D^{\text{test}}\right)$
    $a \leftarrow 0 \in R^T$
    for $t = \{1, \ldots, T$ do
end procedure
```

of datasets $D^{CV} = \{D_1, \ldots, D_{T_{cv}}\}$ and $D^{EV} = \{D_{T_{cv}+1}, \ldots, D_T\}$ where $D_k = \left\{\left(x_i^k, t_i^k, y_i^k\right)_{i=1}^{n_k}\right\}$ is the dataset of k th task. Notably, we have $T_{cv} < T$ and set $T_{cv} = 3$ while $T = 20$ in all our experiments. D^{CV} represents the stream of datasets allocated for cross-validation; this stream allows the learner to replay all samples several times aimed at model hyper-parameters selection as well as system adjustment. By contrast, D^{EV} stands for the actual dataset used for final training and evaluation on the test set. Actually, this means that the model sees the training examples from D^{EV} just one time.

6.5.2 Network architecture

Shallow and a deep DL architectures including a fullyconnected network with two hidden layers of 256 units each (Figure 6.2. a) for PMNIST dataset, a reduced ResNet 18 (Figure 6.2 (b)) for SCIFAR dataset like in [35], will be used in our experiments. While the models are randomly initialized, the stochastic gradient descent (SGD) with mini-batch size 10 is used to optimize the network parameters. Similar to the approach in [25], in order to tune the hyper-parameters, the data of the first three tasks is fed into the first learner several times.

6.5.3 Evaluation metrics

We have used three metrics called Average Accuracy [36], Average Forgetting [36], and Certified Accuracy [31] to evaluate our proposed robust DP-CL model. In the following we briefly define these metrics. The training dataset of each task, D_t, consists of a total B_t mini-batches. After each observation of B_t, the performance of the learner is examined over the whole tasks on the associated test sets. Let $a_{t,i,j} \in [0, 1]$ expresses the accuracy obtained using the test set of task j, after the model has been trained with i th minibatch of task t.

Average Accuracy [36], varying between $[0, 1]$, can be calculated after completing the continually learning procedure of the A-GEM model with all the minibatches corresponding to the t^{th} task and is defined as: $AA_t = \frac{1}{k} \sum_{j=1}^{t} a_{k,B_k,j}$.

Average Forgetting [36], varying between $[-1, 1]$, is computed after the model has been trained for the tasks $1, 2, \ldots, t-1$. This metric is defined as $F_k = \frac{1}{k} \sum_{j=1}^{t-1} f_j^t$ where f_j^t is the forgetting measure on task j after the model is trained for the tasks $1, 2, \ldots, t-1$, obtained as:

$f_j^t = \max_{l \in \{1,\ldots,k-1\}} a_{l,B_l,j} - a_{k,B_t,j}$. AF is crucial to be measured after learning the entire tasks for a two-fold reason. On one hand, AF quantifies the accuracy degradation on the earlier tasks, while on the other hand it specifies how fast a model learns a new task.

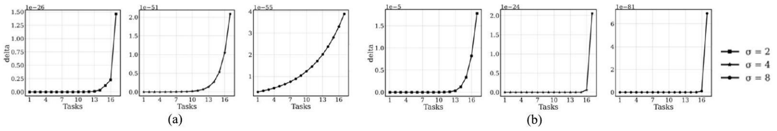


Figure 6.3: Behavior of δ vs. different tasks for $\epsilon = 2$, as well as different level of noise $\sigma \in \{2, 4, 8\}$. (a) PMNIST dataset, (b) SCIFAR dataset.

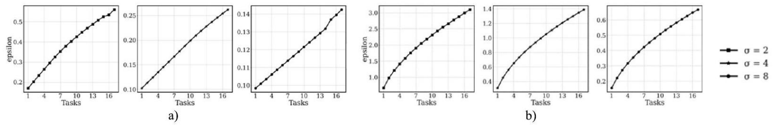


Figure 6.4: Behavior of ϵ vs. different tasks for $\delta = \frac{1}{10000}$, as well as different level of noise $\sigma \in \{2, 4, 8\}$. (a) PMNIST dataset, (b) SCIFAR dataset.

Certified Accuracy [31] measures the prediction robustness, varying between $[0, 1]$ is defined as $CF \triangleq \frac{\sum_{i=1}^{|\text{test}|} \text{isCorrect}(x_i) \ \& \ \text{isRobust}\ (x_i)}{|\text{ test }|}$ where |test| is the size of testing set and isCorrect (x_i) denotes a function returning 1 if the prediction on test sample x_i returns the correct label, and 0 otherwise, and isRobust (x_i) returns 1 if the robustness size is larger than a given attack bound μ_a and 0 otherwise.

The three metrics evaluate our proposed robust DP-CL model, each covering a distinct high-level aspect. Average Accuracy assesses the overall performance across all tasks, indicating how well the model performs after training on multiple tasks. Average Forgetting measures the model's memory retention capability by quantifying accuracy degradation on earlier tasks and the speed of learning new tasks. Certified Accuracy evaluates prediction robustness against adversarial attacks.

6.5.4 Behavior analysis of DP's parameters

In this section, we aim for observing the behavior of DP parameters (ϵ, δ) for two abovementioned DL architectures. To generate the figures, we have exploited the MAA for two various dataset MNIST(a) and CIFAR(b). Figure 6.3 shows six plots where δ is calculated for a given $\epsilon = 2$, while $\sigma \in \{2, 4, 8\}$, and $t \in \{1 \dots 17\}$. As it can be seen from Figure 6.3 (a), for the cases $\sigma = 2$ as well as $\sigma = 4$ the value of δ smoothly increases during the first tasks, while sharply grows for the 4 last tasks. While, in case of $\sigma = 8\delta$ values are smoothly rising for the entire tasks. As it would be expected, the more noise we add to the classifier, the smaller values of δ would be resulted.

Figure 6.4 depicts six other plots where ϵ is calculated for a given $\delta =$

$\frac{1}{10000}$, while σ, and t are opted same as what we used to generate Figure 6.3. As it would be expected, the more noise we add to the classifier, the smaller values of ϵ would be resulted.

6.5.5 Performance evaluation

Now in the following we respond to Q1-Q5 separately.

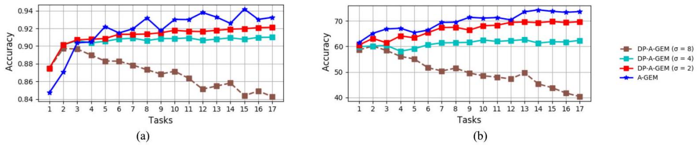
Q1: How does the added DP mechanism affect the accuracy of the A-GEM algorithm?

To answer this question, we compare the accuracy of AGEM with or without adding DP. To do so, we execute the DP-A-GEM with different level of noise $\sigma \in \{2, 4, 8\}$. We further consider the high GPB assumption where $\epsilon = 2$ for PMNIST while for SCIFAR we have $\epsilon = 4$. By doing so, the spent PB will no longer reach to the GPB, and thus no additional learner is needed to be added (i.e., $L = 1$). Figure 6.5 shows the average accuracy after 5 executions for each configuration on PMNIST (Figure 6.5 (a)) and SCIFAR (Figure 6.5 (b)) datasets. As it can be observed, upon increasing the level of noise, the accuracy is reduced so that in case of $\sigma = 8$, a CF phenomenon has been happened, i.e., this can be interpreted from the negative slope of this curve. As another important observation, the results of DP-A-GEM method have less fluctuations and more stable accuracy as compared to the A-GEM method in which DP has been eliminated.

Q2: What is the impact of having several learners on the accuracy of the DP-CL process?

To observe the accuracy of our proposed DP-A-GEM method we need to include several learners in the process. In this regard, aimed at involving two learners, three various small GPB values ($\epsilon = \{0.41, 0.19, 0.12\}$ for PMNIST dataset and $\epsilon = \{2.2, 1.22, 0.5\}$ for SCIFAR dataset) are considered for different levels of noise $\sigma = \{2, 4, 8\}$. As observed in Figure 6.4, the value of σ affects the spent PB for each iteration of each learner. To perceive how these two learners are subsequently involved, we get into one of our experiments shown in Figure 6.6(a). To do so, using PMNIST dataset, three different configurations of (ϵ, δ) including $(0.41, 0.0005)$, $(0.19, 0.0005)$, and $(0.12, 0.0005)$ have been utilized. As it is witnessed, once the PB of first learner reaches to GPB at the end of task 9 , the second learner comes into play to proceed the training process.

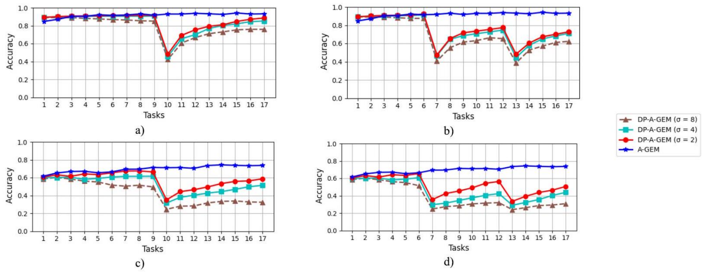
From Figure 6.6, a sudden drop in accuracy can be observed when a new learner starts its learning process. For example, in Figure 6.6(a), the accuracy of three abovementioned configurations is decreased about 40 percent. However, by insisting the training process via the second learner, the accuracy gradually returns to the previous value. There are two main reasons for this issue. First, the noise generated by Gaussian mechanism starts to be added to the weights from the first iteration of second learner. Whilst for the first learner this noise is added after the fine-tuning step, mitigating the im-



(a)

(b)

Figure 6.5: The accuracy of A-GEM method and the proposed DP-A-GEM method for PMNIST (a) and SCIFAR (b) datasets. Various levels of noise $\sigma \in \{2, 4, 8\}$ have been adjusted.



a)

b)

c)

d)

Figure 6.6: The results of DP-CL algorithm for PMNIST and SCIFAR datasets, when two learners are used during the process, show in a and c respectively. For having two learners (a), GPB adjusted such that the first learner will finish its PB at the end of task 9. Since three levels of noise $\sigma = \{2, 4, 8\}$ have been considered during our experiments, three values of (ϵ, δ) including $(0.41, 0.00001), (0.19, 0.00001)$, and $(0.12, 0.\ 00001)$ used for PMNIST dataset, and (2.2, 0. 00001), (1.22, 0. 00001), and (0.5, 0. 00001) used for SCIFAR dataset. Moreover, for having three learners, the GPB values (ϵ, δ) adjusted such that the PB of first learner reaches to GPB after training task 6 , and PB of second learner reaches to GPB after training task $12((0.35, 0.00001), (0.16, 0.00001)$, and $(0.115, 0.00001)$ for PMNIST (b), and $(1.8, 0.00001), (0.81, 0.00001)$, and $(0.4, 0.00001)$ for SCIFAR (d)).

pact of noise on accuracy. Second, the second learner does not exploit D^{CV} for hyper-parameters' fine-tuning. To circumvent this performance degradation, besides of using the D^{CV} for fine-tuning, we start earlier the training process for new learners, i.e., ES.

Q3: How the ES can deal with the performance degradation in the training process?

ES here means that the training process of the new learner commences one task earlier than the task where the spent PB reaches to the GPB. For instance, in case of having two learners, the second learner initiates its training at the beginning of task 9 , whilst the first learner runs out its PB at the end of this task. During this time, both learners are involved in learning task 9 , concurrently. Notably, this will be performed only during the training process where we aimed to fine-tuning the learners. Thus, during the inference time, the entire samples belonging to task 9 are fed to l_1.

Now, we re-execute all our experiments for two various cases. In the first one, namely FT, only D^{CV} is used for finetuning. For the second case, called FT-ES, ES is involved, as well. The curves with transparent colors illustrated in Figure 6.7, correspond to the accuracies corresponding to the FT, dark colors are associated with the accuracies corresponding to the FT-ES. If we have only FT, the accuracy respectively increases $37\%, 18\%$ for PMNIST and SCIFAR datasets, as compared to their counterparts when even FT is not performed. Moreover, in case of FT-ES the accuracy respectively improves to $4\%, 6\%$ for PMNIST and SCIFAR datasets, as compared to their counterparts when only FT is performed. In addition to the accuracy, the forgetting score is evaluated for different levels of noise, when one/two learners are utilized in the process (See Figure 6.8).

6.5.6 The impact of attacks on the robust DP-CL process

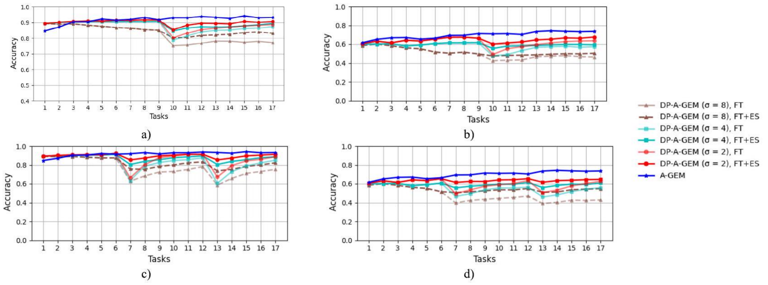
Q4: How the proposed robust DP-CL acts against attacks?

In this regard, we first apply the four white-box attacks mentioned in Section 3.3 on 5 different scenarios, comprised of: 1) A-GEM algorithm, 2) DP-A-GEM, 3) PixelDP-AGEM, 4) RAGEM-PixelDP-A-GEM, and 5) EEM-RAGEMPixelDP-A-GEM. Before that, we applied the attacks on A-GEM and DP-A-GEM algorithms. Figure 6.9 shows the impact of attack on A-GEM algorithm, after learning of each task, the four attack algorithms have been applied on test set and the accuracy has been measured (Figure 6.9, light colors). Compared to A-GEM algorithm, the DP-AGEM algorithms obtained better accuracy by 9.3 percent and 4.6 percent for PMNIST and SCIFAR datasets respectively. Finally, by measuring forgetting average and certified accuracy metrics, we evaluated the effect of the proposed robust solutions (PixelDP, robust-A-GEM (RAGEM), and efficient-EM(EEM)) by applying the white-box attacks on two PMNIST (Figure 6.10 (a)) and SCIFAR (Figure 6.10 (b)) datasets.

6.5.7 Data consumption

Q5: How much data the DP-CL process will need?

The number of replaced samples in EM has been observed during the training process for both datasets, which helps to have a good estimation of number of necessary training data in DP-CL training process. Figure 6.11 shows the number of replaced samples for different levels of noise $\sigma = \{2, 4, 8\}$ for the two PMNIST (Figure 6.11 (a, b)) and SCIFAR (Figure 6.11(c, d)) datasets when one or two learners have been used in the process. As it



a) b) c) d)

Figure 6.7: Using FT and ES, the accuracy of DP-CL algorithm increases when two or three learners have been used. For each experiment, we investigate the effect of ES and start the training process for I_{c+1} one task earlier. At each plot, the light color shows the result without using ES (e.g., $DP - A - GEM\sigma = 2), FT.$) and the dark color shows when ES uses along with FT (e.g., $DP - A - GEM(\sigma = 2), FT + ES$). The DP parameters $((\epsilon, \delta))$ have been adjusted the same as previous step.

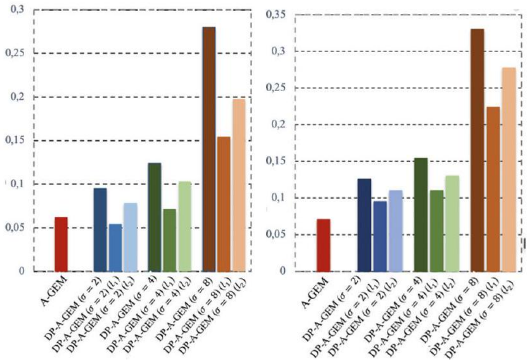


Figure 6.8: The forgetting average has been calculated for two PMNIST (a), and SCIFAR (b) datasets. For each one, when there are one or two learners in the process and $\sigma = \{2, 4, 8\}$, the forgetting measure has been calculated. When two learners have been used, the forgetting score has been measured for each of which (i.e., I_1, I_2).

can be observed from Figure 6.11, when there is one learner, and $\sigma = 2$, the training process needs more training samples. Therefore, the more we add noise, the less data the DP-CL process will need.

6.6 Discussion

There are two DL networks in our experiments, a shallow with 2 hidden layers including $\cong 269,000$ trainable parameters and another one with a deeper architecture including 18 hidden layers including 11 million trainable parameters. By measuring the DP parameters, it can be observed that, the deeper the network, the more noise will be added to the network, and consequently the DP parameters increase more quickly. For instance, at the end of training Resnet 18 , the value of δ is more than 20 times higher than the shallow network for all levels of noise. A similar effect is observed for ϵ such that for different levels of noise $\sigma \in \{2, 4, 8\}$, its value is $5.45, 5.38$, and 5 times larger than shallow network respectively. Notably, although the deeper network has 40 times more parameters than the shallow network, the DP parameters do not increase linearly with respect to number of networks' parameters.

To increase the privacy of both networks, we raised the noise level from 2 to $8(\sigma \in \{2, 4, 8\})$. Although, the accuracy of both networks constantly increases for $\sigma \in \{2, 4\}$, it decreases by about 6% and 20% for FC2 and Resnet 18 networks respectively, when $\sigma = 8$. Interestingly, the results of DP-A-GEM method have less fluctuations and more stable accuracy compared to simple A-GEM method specially for $\sigma \in \{2, 4\}$. In the next step, we decreased the GPB to evaluate the performance of DP-A-GEM when there are several learners. Depending on the noise level, the accuracy of second and third learner has a sudden drop between 35-45% for PMNIST dataset, and between 20-30% for SCIFAR dataset. But by using the fine-tuning and ES strategies, the performance increases about 43% and 23% for FC2 and Resnet18 networks respectively. To accurately measure the degradation, when there are several learners, we calculate the forgetting accuracy. Notably, when there are two learners, the forgetting accuracy of each learner is less than when there is one learner in the process. For instance, the forgetting accuracy for first and second learner are 0.071 and 0.103 respectively (Figure 6.8 (a), $\sigma = 4$) and less than 0.124 which is the forgetting score of when there is just one learner. In other words, the long training process with just one learner leads to high forgetting score and the CF will finally happen.

Furthermore, the proposed three methods to robustize the DP-CL process are effective against the applied four whitebox attacks. First, we applied the attacks on simple A-GEM algorithm and DP-A-GEM algorithm to investigate the effect of adding DP against attacks. As shown in Figure 6.9, almost in all cases the DP-enabled version of A-GEM increases the accuracy

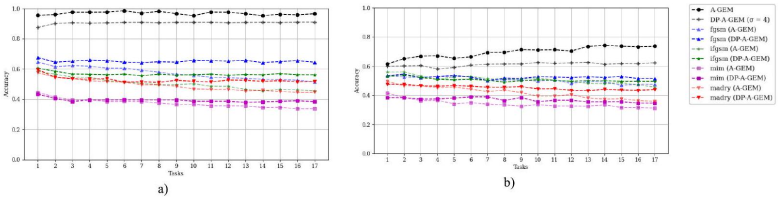


Figure 6.9: The (certified) accuracy of A-GEM algorithm (light colors) and DP-A-GEM algorithm (dark colors) after applying attacks on PMNIST (a) and SCIFAR (b) datasets has been measured (i.e., $I_\infty(\mu = 0.1), \sigma = 4$).

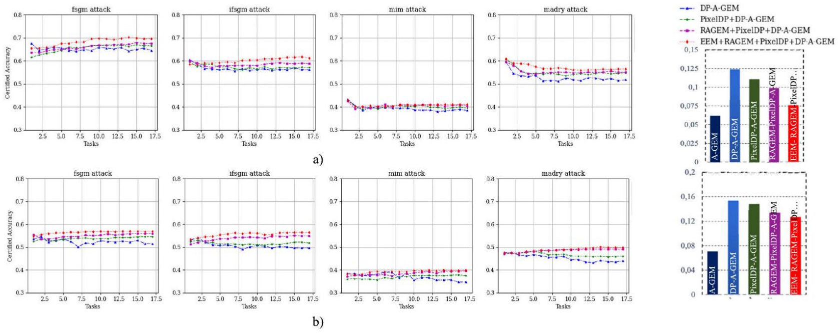


Figure 6.10: The (certified) accuracy of A-GEM algorithm (light colors) and DP-A-GEM algorithm (dark colors) after applying attacks on PMNIST (a) and SCIFAR (b) datasets has been measured (i.e., $I_\infty(\mu = 0.1), \sigma = 4$).

compared to simple A-GEM which is about 7 percent for PMNIST, and 4 percent for SCIFAR on average. Then, by adding the robust methods one-after-another, the attacks have been applied. As shown in Figure 6.10, each of the proposed methods has positive effect on the accuracy of the DP-A-GEM algorithm under attacks. On average, PixelDP improved the accuracy by 3.3 percent for PMNIST and 3.8 percent for SCIFAR dataset. Robust-A-GEM, which is applied after adding PixelDP method, improved the accuracy by 1.65 for PMNIST, and 4.1 percent for SCIFAR dataset. Finally, the efficient-EM increased the accuracy by 3.6 and 2.3 percent for PMNIST and SCIFAR datasets respectively. Therefore, the robustness methods increased the accuracy of DP-A-GEM algorithms by 8.55 and 10.2 percent for PMNIST and SCIFAR datasets respectively. Moreover, by adding the robustness methods, the forgetting average decreased from 0.124 to 0.075 for PMNIST and from 0.155 to 0.123 for SCIFAR dataset.

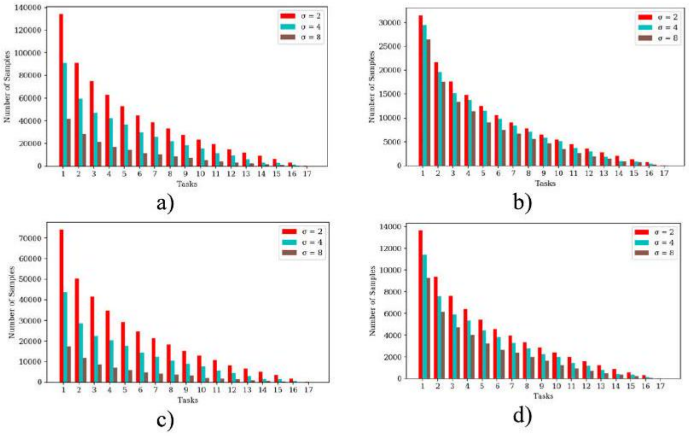


Figure 6.11: The number of replaced training samples have been monitored for the two PMNIST (a and b) and SCIFAR (c and d) datasets when one (a, c) or two (b,d) learners have been used in the training process.

6.7 Conclusion

The major contribution of this paper is adding differential privacy (DP) into continual learning (CL) procedures, aimed at protecting against adversarial examples. In CL processes, the model learns sequentially and endlessly from timevarying data streams which makes the task of adding DP to CL challenging. More explicitly, the added noise due to DP together with the endless learning feature of CL leads to CF which is a serious obstacle. To address this concern, we have proposed an innovative approach by which we cannot only strike a tradeoff between privacy and utility, but also mitigate the CF. We continually control the instantaneous spent PB to not exceed the available GPB. Besides, a threestep robust procedure is also included in our approach to mitigate the negative impact of CF, as much as possible. We also assessed the proposed approach against four wellrecognized adversarial attacks comprised of: 1) FGSM, 2) IFGSM, 3) MIM, and 4) the attack by Madry et al. [29]. Our simulation results validated the effectiveness of the proposed method in facing such strong attacks so that we could improve the criteria of both the certified accuracy and the forgetting measure, simultaneously.

References

[1] Y. LeCun, Y. Bengio, and G. E. Hinton, ”Deep learning,” Nature, vol. 521, no. 7553 , pp. 436-444, Dec. 2015.

[2] C. Su, Z. Xu, J. Pathak, and F. Wang, ”Deep learning in mental health outcome research: A scoping review,” Transl. Psychiatry, vol. 10, no. 1, pp. 1-26, Dec. 2020.

[3] A. I. Karoly, P. Galambos, J. Kuti, and I. J. Rudas, ”Deep learning in robotics: Survey on model structures and training strategies,” IEEE Trans. Syst., Man, Cybern., Syst., vol. 51, no. 1, pp. 266-279, Jan. 2021.

[4] A. Lavecchia, ”Deep learning in drug discovery: Opportunities, challenges and future prospects,” Drug Discovery Today, vol. 24, no. 10, pp. 2017-2032, Oct. 2019.

[5] L. F. Gomez, A. Morales, J. R. Orozco-Arroyave, R. Daza, and J. Fierrez, ”Improving Parkinson detection using dynamic features from evoked expressions in video,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2021, pp. 1562-1570.

[6] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, ”Continual lifelong learning with neural networks: A review,” Neural Netw., vol. 113, pp. 54-71, May 2019.

[7] M. Delange, R. Aljundi, M. Masana, S. Parisot, X. Jia, A. Leonardis, G. Slabaugh, and T. Tuytelaars, ”A continual learning survey: Defying forgetting in classification tasks,” IEEE Trans. Pattern Anal. Mach. Intell., early access, Feb. 5, 2021, doi: 10.1109/TPAMI.2021.3057446.

[8] N. Akhtar and A. Mian, ”Threat of adversarial attacks on deep learning in computer vision: A survey,” IEEE Access, vol. 6, pp. 14410-14430, 2018 .

[9] X. Huang, D. Kroening, W. Ruan, J. Sharp, Y. Sun, E. Thamo, M. Wu, and X. Yi, ”A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability,”, Comput. Sci. Rev., vol. 37, Aug. 2020, Art. no. 100270.

[10] H. Yousuf, M. Lahzi, S. A. Salloum, and K. Shaalan, ”Systematic review on fully homomorphic encryption scheme and its application,” in Recent Advances in Intelligent Systems and Smart Applications. Springer, 2021, pp. 537-551.

[11] J. Li, X. Kuang, S. Lin, X. Ma, and Y. Tang, ”Privacy preservation for machine learning training and classification based on homomorphic encryption schemes,” Inf. Sci., vol. 526, pp. 166-179, Jul. 2020.

[12] L. Sweeney, ”K-anonymity: A model for protecting privacy,” Int. J. Uncertainty, Fuzziness Knowl.-Based Syst., vol. 10, no. 5, pp. 557-570, 2002.

[13] C. C. Aggarwal, ”August. On K-anonymity and the curse of dimensionality,” in Proc. VLDB, vol. 5, 2005, pp. 901-909.

[14] J. Brickell and V. Shmatiko, ”The cost of privacy: Destruction of data-mining utility in anonymized data publishing,” in Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2008, pp. $70 - 78$.

[15] C. Dwork and A. Roth, ”The algorithmic foundations of differential privacy,” Found. Trends Theor. Comput. Sci., vol. 9, nos. 3-4, pp. 211-407, 2014.

[16] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, ”Deep learning with differential privacy,” in Proc. ACM SIGSAC Conf. Comput. Commun. Secur, 2016, pp. 308-318.

[17] J. Zhao, Y. Chen, and W. Zhang, ”Differential privacy preservation in deep learning: Challenges, opportunities and solutions,” IEEE Access, vol. 7, pp. 48901-48911, 2019.

[18] C. Dwork, M. Naor, T. Pitassi, and G. N. Rothblum, ”Differential privacy under continual observation,” in Proc. 42nd ACM Symp. Theory Comput. (STOC), 2010, pp. 715-724.

[19] Z. Bu, J. Dong, Q. Long, and S. Weijie, ”Deep learning with Gaussian differential privacy,” Harvard Data Sci. Rev., vol. 2020, no. 23, Sep. 2020, doi: $10.1162/99608$ f92.cfc5dd25

[20] P. C. Mahawaga Arachchige, P. Bertok, I. Khalil, D. Liu, S. Camtepe, and M. Atiquzzaman, ”Local differential privacy for deep learning,” IEEE Internet Things J., vol. 7, no. 7, pp. 5827-5842, Jul. 2020.

[21] N. Phan, X. Wu, H. Hu, and D. Dou, ”Adaptive Laplace mechanism: Differential privacy preservation in deep learning,” in Proc. IEEE Int. Conf. Data Mining (ICDM), Nov. 2017, pp. 385-394.

[22] M. Lécuyer, R. Spahn, K. Vodrahalli, R. Geambasu, and D. Hsu, ”Privacy accounting and quality control in the sage differentially private ML platform,” in Proc. 27th ACM Symp. Operating Syst. Princ., Oct. 2019, pp. $181 - 195$

[23] R. Cummings, S. Krehbiel, K. A. Lai, and U. Tantipongpipat, ”Differential privacy for growing databases,” in Proc. Adv. Neural Inf. Process. Syst., 2018, pp. 8864-8873.

[24] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, Q. S. T. Quek, and H. V. Poor, ”Federated learning with differential privacy: Algorithms and performance analysis,” IEEE Trans. Inf. Forensics Security, vol. 15, pp. 3454-3469, 2020.

[25] A. Chaudhry, M. A. Ranzato, M. Rohrbach, and M. Elhoseiny, ”Efficient lifelong learning with a-gem,” in Proc. Int. Conf. Learn. Represent., 2019.

[26] C. Dwork, F. McSherry, K. Nissim, and A. Smith, ”Calibrating noise to sensitivity in private data analysis,” in Proc. Theory Cryptogr. Conf., 2006, pp. $265 - 284$

[27] I. J. Goodfellow, J. Shlens, and C. Szegedy, ”Explaining and harnessing adversarial examples,” 2014, arXiv:1412.6572.

[28] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li, ”Boosting adversarial attacks with momentum,” 2017, arXiv:1710.06081.

[29] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, ”Towards deep learning models resistant to adversarial attacks,” 2017.

[30] A. Kurakin, I. Goodfellow, and S. Bengio, ”Adversarial examples in the physical world,” 2016, arXiv: 1607.02533.

[31] M. Lecuyer, V. Atlidakis, R. Geambasu, D. Hsu, and S. Jana, ”Certified robustness to adversarial examples with differential privacy,” in Proc. IEEE Symp. Secur. Privacy (SP), May 2019, pp. 656-672.

[32] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, and D. Hassabis, ”Overcoming catastrophic forgetting in neural networks,” Proc. Nat. Acad. Sci. USA, vol. 114, no. 13, pp. 3521-3526, 2017.

[33] F. Zenke, B. Poole, and S. Ganguli, ”Continual learning through synaptic intelligence,” in Proc. 34th Int. Conf. Mach. Learn., 2017, pp. 3987-3995.

[34] A. Krizhevsky and G. Hinton, ”Learning multiple layers of features from tiny images,” Univ. Toronto, Toronto, ON, Canada, Tech. Rep., 2009.

[35] D. Lopez-Paz and M. A. Ranzato, ”Gradient episodic memory for continual learning,” in Proc. Adv. Neural Inf. Process. Syst., vol. 30, 2017, pp. 6467-6476.

[36] A. Chaudhry, P. K. Dokania, T. Ajanthan, and P. H. Torr, ”Riemannian walk for incremental learning: Understanding forgetting and intransigence,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 532-547.

[37] M. Gomez-Barrero, E. Maiorana, J. Galbally, P. Campisi, and J. Fierrez, ”Multi-biometric template protection based on homomorphic encryption,” Pattern Recognit., vol. 67, pp. 149-163, Jul. 2017.

[38] B. Jiang, J. Li, G. Yue, and H. Song, ”Differential privacy for industrial Internet of Things: Opportunities, applications, and challenges,” IEEE Internet Things J., vol. 8, no. 13, pp. 10430-10451, Jul. 2021.

6.8 APPENDIX A: Modifying a-gem update rule (EQ. (5))

Here we provide the proof DP-A-GEM' update rule, stated in Section IV (C2), Eq. 5. To proof, we first invoke the DPA-GEM problem in Eq. 4 as follows:

$$\begin{aligned} &\min_{\tilde{g}_j} \frac{1}{2} \left\| g - \tilde{g}_j \right\|_2^2 \\ &\text{s.t. } \langle \tilde{g}_j, g_{j,k} \rangle \geq 0 \quad \forall k < t \end{aligned} \tag{A.1}$$

Replacing $\tilde{g}_j$ with z and rewriting Eq. A. 1 yields:

$$\min_z \frac{1}{2} z^\top z - g^\top z \text{ s.t. } - z^\top g_{j,ref} \leq 0 \tag{A.2}$$

Note that we removed the term $g^\top > g$ from the OF and change the direction of the inequality constraint. The Lagrangian function can be acquired as:

$$L(z, \alpha) = \frac{1}{2} z^\top z - g^\top z - \alpha z^\top g_{j, \text{ ref}} \tag{A.3}$$

Now, we pose the dual of Eq. A. 3 as:

$$\theta_D(\alpha) = \min_{\text{z}} L(z, \alpha) \tag{A.4}$$

Lets find the value z^* that minimizes the $L(z, \alpha)$ by setting the derivatives of $L(z, \alpha)$ w.r.t. to z to zero:

$$\begin{aligned} \nabla_z L(z, \alpha) = 0 \\ z^* = g + \alpha g_{j,ref} \end{aligned} \tag{A.5}$$

The simplified dual after putting the value of z^* in Eq. A. 4 can be written as:

$$\begin{aligned} \theta_D(\alpha) =& \frac{1}{2} \left(g^\top g + 2\alpha g^\top g_{j,ref} + \alpha^2 g_{j,ref}^\top g_{j,ref} \right) \\ &- g^\top g - 2\alpha g^\top g_{j,ref} - \alpha^2 g_{j,ref}^\top g_{j,ref} \\ =& - \frac{1}{2} g^\top g - \alpha g^\top g_{j,ref} - \frac{1}{2} \alpha^2 g_{j,ref}^\top g_{j, \text{ ref}} \end{aligned} \tag{A.6}$$

This solution $\alpha^* = \max_{\alpha; \alpha > 0} \theta_D(\alpha)$ to dual is given by:

$$\begin{aligned} \nabla_\alpha \theta_D(\alpha) = 0 \\ \alpha^* = - \frac{g^\top g_{ref}}{g_{ref}^\top g_{ref}} \end{aligned} \tag{A.7}$$

By putting α^* in Eq. A.5, we recover the A - GEM update rule:

$$z^* = g - \frac{g^\top g_{j,ref}}{g_{j, \text{ ref}}^\top g_{j,ref}} g_{j, \text{ ref}} = \tilde{g} \tag{A.8}$$

Chapter 7

The Impact of Generalization Techniques on the Interplay Among Privacy, Utility, and Fairness in Image Classification

Ahmad Hassanpour, Amir Zarei, Khawla Mallat, AS de Oliveira, Bian Yang

ISBN 978-82-326-8286-7 (printed ver.)
ISBN 978-82-326-8285-0 (electronic ver.)
ISSN 1503-8181 (printed ver.)
ISSN 2703-8084 (online ver.)

NTNU
Norwegian University of
Science and Technology