A Deep Learning Paradigm for Fingerprint Recognition: Harnessing U-Net Architecture for Fingerprint Enhancement and Representation Learning

Thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science in Computer Science and Engineering by Research

by

EKTA GAVAS
2019701005
ekta.gavas@research.iiit.ac.in

International Institute of Information Technology, Hyderabad
(Deemed to be University)
Hyderabad - 500 032, INDIA
February 2024

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled "A Deep Learning Paradigm for Fingerprint Recognition: Harnessing U-Net Architecture for Fingerprint Enhancement and Representation Learning" by Ekta Gavas, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date	Advisor: Dr. Anoop Namboodiri

To my Family

Acknowledgments

I want to sincerely thank the following individuals who have played a significant role in the completion of my thesis toward the fulfillment of my master's degree.

Firstly, I would like to express my heartfelt gratitude to my advisor, Dr. Anoop Namboodiri, for his unwavering support and mentorship throughout this research. His invaluable advice and insights have been instrumental in shaping this work. Your guidance has transformed my experience in academia, enriching both my thesis work and my general approach to scientific inquiry.

I am also thankful to this institute IIIT Hyderabad for providing me the opportunity to pursue my graduation and providing a research-centric environment that fosters learning and research. I am grateful to the BASIL and CVIT Lab, for providing me with the necessary resources and support, without which this thesis would not have been possible. I would like to express my appreciation to the faculty and staff at the institute for their administrative assistance, which has enabled me to conduct my research seamlessly.

I would like to acknowledge and thank my good friend, Kaustubh Olpadkar, for his help and support. The engaging and fruitful discussions we had helped a lot in my research journey so far and will continue to do so.

To my friends, peers, and fellow researchers, your support and companionship have meant more than I can express. Thanks for the much-needed breaks, and shared struggles and triumphs. You made this journey enjoyable and kept my spirits high.

Last but certainly not least, I extend my deepest thanks to my family and my fiancé. Your unconditional love and unwavering confidence in me have been the foundation upon which this work is built. To my parents, thank you for your constant encouragement and for always being there when I needed you the most.

Abstract

Biometric technology has long relied on fingerprint recognition as a trusted means of identity verification. While fingerprint enhancement shares similarities with general image denoising tasks, the unique properties of biometric data make it crucial to treat fingerprints differently from typical real-world images. Deep learning methods have the capacity to capture and improve complex patterns and subtle details in fingerprint data, offering a comprehensive solution. This thesis proposes a deep learning-based method, specifically leveraging the U-Net architecture to achieve superior fingerprint enhancement. At its core, the approach employs a multi-task deep network with domain knowledge from minutia and orientation fields for the enhancement of rolled and plain fingerprint images. We investigate different configurations of the U-Net architecture, examine the effects of various architectural elements, and present extensive experimental evidence to support the effectiveness of our proposed approach.

Subsequently, we explore a self-supervised learning paradigm with fingerprint biometrics for robust feature representation learning. In this direction, we introduce a pre-training technique by utilizing the learned information from the enhancement task. We adopt the enhancement pre-trained encoder for learning fixed-length fingerprint embeddings. We evaluate the performance of the learned embeddings in the verification task compared to standard self-supervised techniques without the explicit need for enhanced fingerprints.

By merging the capabilities of deep learning with the specificity of fingerprint features, the proposed paradigm offers significant improvements in the robustness and accuracy of fingerprint verification. Experimental evaluations on standard benchmarks such as the NIST and FVC datasets demonstrate the efficacy of this approach. We present compelling evidence that our methodology not only improves the quality of fingerprint images but also facilitates a more effective embedding extraction, ultimately leading to enhanced recognition performance.

This work highlights the promising role that deep learning, tailored to biometrics, can play in advancing the field of fingerprint recognition. In addition, it opens up avenues for future research in biometrics, particularly in optimizing computational efficiency and in tackling challenges related to partial and distorted fingerprints.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Biometric Systems

For decades, biometrics have been used as a means for the identification of people for various purposes related to security and defense. From facial recognition of passengers at airports to fingerprint verification in office authentication systems to using iris and fingerprint identification in security systems, biometrics, and biometric systems have been of utmost importance. Over the years, the applications and usage of biometric systems have increased with the increase in the efficiency and reliability of biometric systems. A biometric system is a system that allows the verification and recognition of the identity of an individual based on a physiological or behavioral trait or characteristic [4, 5]. These traits are not related to personal attributes of people like name, age, gender, etc, and hence do not involve any identity-related information. Thus, biometric systems as such allow anonymous recognition [4]. Since ages, pins and passwords or security numbers have been used to authorize a person for making use of any service, for eg. using PIN with ATM card to withdraw cash from ATM or logging into email with password. But these security passwords do not validate the identity of person in particular as anyone bearing the password can be authorized successfully. Hence, biometric becomes important as it is more difficult to forge/steal biometric identity than pins and passwords. This is one of the reasons that biometric systems find applications in varied areas. Various modalities of biometrics are being used by law enforcement agencies and government to keep a track and maintain the record of its citizens. Other than being used for important documents like passport etc, facial recognition, fingerprints etc are used at airports and border security systems. Recently, healthcare has also seen use of biometrics to secure and maintain health records of patients to be accessed by only concerned doctors and patients. Apart from these large-scale applications, biometric systems are being used in user-oriented applications as well like banks or mobile phones, in finance-related applications or in basic security and authentication systems at offices and homes. These applications suggest the crux of having a highly robust and accurate biometric system, which is safe from attacks or security breaches as much as possible.

Biometric systems require the enrollment of users in the system. In this enrollment phase, based on the use case, the biometric traits of the user are captured using specific sensors and entered in the database. After enrollment, the system can be used for identification purposes or verification/authentication purposes. For identification, the test or query entity, hereafter referred to as a template, of a given biometric modality, is compared with each saved template in the database to identify the person (1:N). Whereas, for authentication or verification purposes, the test template of the user is compared with the template of the user in the database (1:1) to validate the person's identity. Based on the modality of biometrics captured, the matching algorithms vary and hence the accuracy of the systems also.

1.2 Characterics of biometric traits

In early days, different kinds of quantitative measures were used for identifying individuals, like arm and foot length, body measurements, etc. Such kind of measures was not very efficient and reliable as these may change over the lifetime of an individual or may get affected in one or the way. The biometric modality should possess a few standard characteristics to be used in a biometric system that is robust enough and can successfully recognize/validate the identity of any individual at all times. Currently, biometric modalities that are used prominently are the face, fingerprints, iris, palm prints, voice, signature, gait, etc. [6] listed seven characteristics for any trait to be suitable for use in biometric systems:

1. Universality: Every individual should possess the trait.

2. Uniqueness: The trait should be sufficiently unique among all individuals in the population.

3. Permanence: The trait should be invariant with time with respect to a given matching algorithm, i.e, it should not change significantly over time.

4. Measurability: It should be possible to acquire the biometric trait using suitable devices without causing much inconvenience to subjects and to process the raw data to extract representative features.

5. Performance: The accuracy obtained with the trait and the resources required to achieve this should meet the application requirements.

6. Acceptability: The individuals in the target population should agree to let their traits be acquired for biometric systems.

7. Circumvention: The biometric trait should not be easily forged and stolen.

Table 1.1 shows the comparison of different biometric techniques in the context of the above biometric characteristics. Moreover, Table 1.2 represents the comparison of biometric systems of various modalities based on additional features like security, accuracy etc.

Table 1.1: Comparison of Biometric Traits based on trait characteristics

Biometrics	Universality	Uniqueness	Permanence	Collectability	Performance	Acceptability	Circumvention
Fingerprint	Medium	High	High	Medium	High	Medium	High
Face	High	Low	Medium	High	Low	High	Low
Iris	High	High	High	Medium	High	Low	High
Voice	Medium	Low	Low	Medium	Low	High	Low
Hand Geometry	Medium	Medium	Medium	High	Medium	Medium	Medium
DNA	High	High	High	Low	High	Low	Low
Signature	Low	Low	Low	High	Low	High	Low
Retina	High	High	Medium	Low	High	Low	High

Table 1.2: Comparison of biometric systems of varying modality based on additional features

Modality	Cost	Equipment	Expertise Req.	Accuracy	Security	Stability	Error Incidence
Fingerprint	Low to moderate	Fingerprint scanner	Moderate	High	High	High	Dryness, dirt, finger injury etc.
Face	Low	Camera (commonly available)	Low to moderate	Varies (can be high with advanced systems)	Moderate	Moderate	Lighting, age, glasses, hair etc.
Iris	High	Specialized iris scanner	High	Very high	High	High	Lighting
Voice	Low	Microphone (commonly available)	Low	Moderate	Low to moderate	Moderate	Age, emotion, noise, colds etc.
Hand Geometry	Moderate	Specialized scanner	Moderate	Moderate to high	High	Moderate	Age, weight changes, hand injury etc.
Signature	Low to moderate	Pen tablet or paper and scanner	Low to moderate	Moderate	Low to moderate	Low	Mood, age, health etc.
Retina	High	Specialized retina scanner	High	Very high	Very high	High	Glasses, eye diseases

1.3 Fingerprint as biometric

The science and application of fingerprint biometrics boast a storied history that spans centuries. Existing archaeological evidence suggests that as far back as 6000 BC, fingerprints were used as a means of identification [7]. These early instances of usage ignited interest in fingerprint biometrics, prompting extensive exploration and research into the technology.

As technology advanced, commercial entities began to incorporate biometrics for regulating physical access to premises. Today, as the drive to mitigate fraudulent activities intensifies, along with the escalating need for securing access to physical and digital assets, fingerprint biometrics have soared in popularity, making it an extensively utilized technology worldwide.

The uniqueness of an individual's fingerprints lies in the minute details known as minutiae points. These are predominantly ridge endings and ridge bifurcations. A ridge ending refers to the point where a ridge ends and a ridge bifurcation is the point where a ridge splits into two. These minutiae points are unique for each individual and provide the basis for the high reliability of fingerprint recognition systems. Each individual, including identical twins, possesses distinct fingerprints characterized by spe-

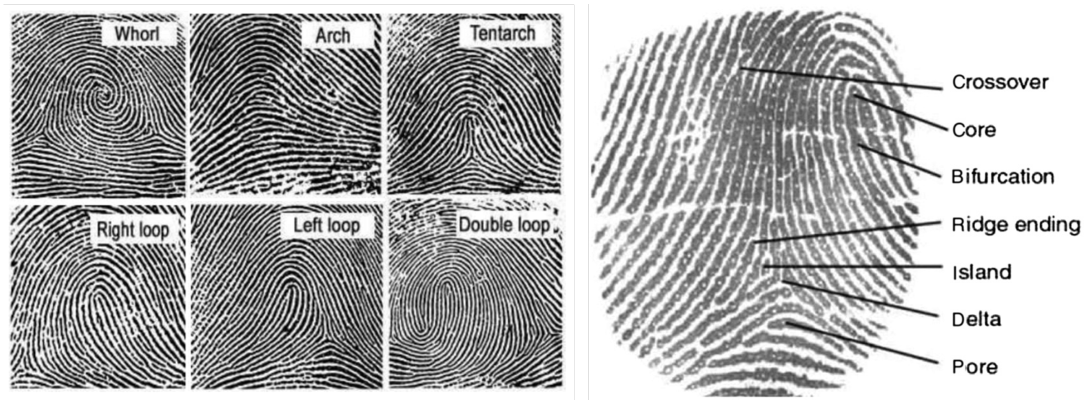

Figure 1.1: Type of singularity points (minutia)

cific patterns of whorls, arches, loops, ridges, and valleys [8, 9]. Figure 1.1 These patterns remain stable and unchanged throughout an individual's lifetime, cementing the reliability of fingerprint biometrics.

Several advantages accompany the use of fingerprints as biometrics:

1. Fingerprints offer simplicity and user-friendliness, making them a convenient choice for biometric identification.

2. They provide cost-effective solutions that are easily deployable, enhancing their appeal.

3. Unlike passwords and tokens, fingerprints cannot be lost, guessed, or transferred, thereby offering enhanced security and authentication.

4. The chances of fingerprints being spoofed or stolen are significantly lower, further strengthening their security value.

The combination of these unique properties, along with high recognition accuracy, makes fingerprints a preferred choice for numerous biometric applications, often superseding other biometric traits such as iris, face, or voice recognition.

Fingerprint biometrics serve a variety of functions across numerous applications [4, 10], a few of which include:

- **Border Control and Travel:** In an effort to maintain stringent security across borders and regulate international movement, fingerprint biometrics play an integral role. For instance, all immigrants and visitors are required to submit their biometric details as part of visa procedures before entering a country.

- **Law Enforcement:** Fingerprints collected from crime scenes or suspects are cross-referenced with those in databases for identifying culprits. In countries like India, fingerprints have played a pivotal role in registering citizens into the national database (Aadhar), facilitating a multitude of authentication requirements.

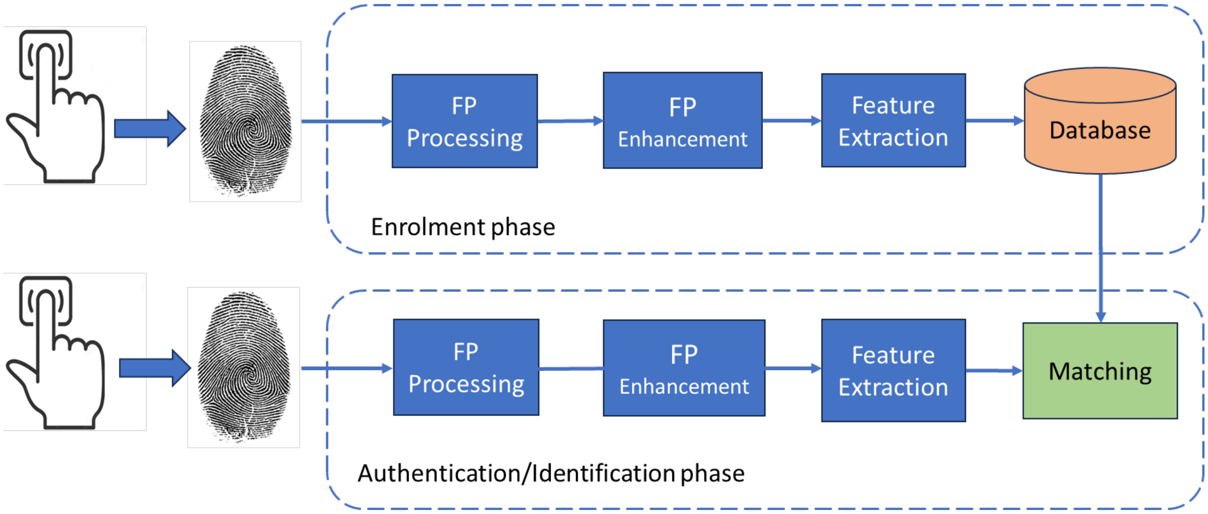

Figure 1.2: General architecture of Fingerprint Biometric authentication system

- **Commercial Applications:** The current commercial landscape is dotted with the usage of fingerprint biometrics. From door lock systems and office access controls to surveillance systems and the maintenance of confidential patient records in healthcare, fingerprints serve various purposes. Personal devices such as mobiles and tablets, and the financial services sector, including banks, also leverage fingerprint technology for secure customer transactions.

1.4 Fingerprint Recognition

Within the realm of biometric recognition technologies, fingerprint recognition stands tall as an exceptionally reliable and widely adopted modality. Its application has permeated various sectors, including law enforcement, border control, commercial security, and personal device authentication, testifying to the unique advantages it brings.

The process of fingerprint recognition is essentially two-staged: enrollment and matching 1.2. The enrollment phase captures an individual's fingerprint image, processes it, and distills it into a unique set of features. These features are then stored as a reference template. The matching phase follows, where a newly acquired fingerprint image is compared against all the stored templates in a search for a match (identification) or compared only to the stored template of the user (authentication).

A typical fingerprint recognition pipeline encompasses majorly four major stages as follows:

1. Acquisition: In this stage, the fingerprint image is acquired from the user with the help of specialized fingerprint sensors. The acquisition can be a slap or individual fingerprints.

2. Preprocessing: Image quality, for instance, can significantly impact the accuracy of fingerprint recognition systems. Poor quality or degraded images can lead to incorrect feature extraction and, therefore, inaccurate matches. Hence, once the image is acquired, preprocessing steps are

employed to enhance the image quality. These steps may include normalization, orientation estimation, region of interest extraction, thinning, and other image enhancement techniques. If the image captured is a slap fingerprint, this process can include a segmentation step to segment out individual fingerprint images. Preprocessing steps are generally performed on individual fingerprint images.

3. Feature Extraction: This step in the pipeline involves feature extraction, where distinguishing characteristics such as minutiae points or ridge patterns are identified and extracted. The type of features extracted, local or global, governs the choice of the algorithm being used for matching at next-stage. The extracted features can be post-processed in a standard template or a more useful format for further processing (to store in a database or for matching).

4. Matching: The final stage involves matching the extracted features from the input fingerprint image against the stored templates in the database. Different techniques are used for matching in fingerprint recognition systems, including pattern matching, minutiae-based matching, ridge feature-based matching, and image-based matching. Advances in deep learning and artificial intelligence have led to the emergence of new methods that improve the accuracy and efficiency of fingerprint recognition. These algorithms can broadly be divided into two categories: minutiae-based and image-based. Minutiae-based techniques compare minutiae points, such as ridge bifurcations and endings, between the input image and the stored template. Image-based techniques, on the other hand, use global pattern geometry and ridge flow to compare the images. Most of these methods output a matching score which determines whether the fingerprint is recognized (in identification tasks) or whether it matches a previously enrolled fingerprint (in verification tasks)

Each stage in the fingerprint recognition pipeline plays a crucial role in the overall performance of the system, influencing its accuracy, speed, and robustness.

1.5 Motivation

While fingerprint recognition has proven its merit, it also presents certain challenges that motivate this thesis. A crucial factor is the quality of the fingerprint image. Poor or degraded images can hinder accurate feature extraction due to the presence of spurious or missing minutia, leading to potential misidentification. This realization has spawned research into image enhancement techniques to improve the quality of fingerprint images, one of the key motivations for this thesis.

Beyond enhancing image quality, there is a compelling need to make the fingerprint feature extraction process efficient in order to increase recognition performance. Deep learning offers a promising avenue to accomplish this, with its ability to learn complex patterns and extract meaningful representations. Consequently, the development of deep learning models for learning robust and accurate fingerprint representations is one of the focus areas of this thesis. To achieve this, we explore the self-supervised learning paradigm along with reusing the knowledge from the fingerprint enhancement task.

By confronting these challenges and delving into the potential of deep learning, this research aims to contribute substantially to the development of more accurate, efficient, and secure fingerprint recognition systems, revolutionizing the use of this important biometric modality.

1.6 Contributions

We make the following contributions in this work:

1. We propose a U-Net-based deep learning architecture specifically designed for fingerprint enhancement task. We propose several refinements to vanilla U-Net along with exploring wavelet transform in this architecture.

2. In order to improve the enhancement performance, we explore the usage of domain knowledge in the context of minutia extraction and orientation estimation to optimize the fingerprint enhancement task further.

3. We suggest a pre-training technique in a self-supervised paradigm with the U-Net encoder on fingerprint enhancement task and demonstrate the usefulness of this approach in robust representation learning.

4. We evaluate our methods with various evaluation metrics demonstrating its effectiveness in fingerprint enhancement and verification tasks and also provide a comparison with previous state-of-the-art methods.

1.7 Thesis Organization

This thesis report contains 5 chapters. Chapter 2 discusses in detail existing fingerprint enhancement and representation learning techniques. Chapter 3 gives the proposed algorithm for fingerprint enhancement using the deep learning paradigm. Chapter 4 gives the proposed approach for fingerprint representation learning in self-supervised paradigm. Chapter 5 concludes and discusses the future work.

Chapter 2

Literature Review: Fingerprint Enhancement and Feature Extraction

The application of fingerprint-based authentication systems has gained traction in recent years, not only in general-purpose devices like laptops, mouse, smartphones, and pen drives but also at the industrial or defense level, making fingerprints stand out as a prevalent and widely adopted modality. The word *fingerprint* is usually associated with the term *individuality*, but the uniqueness is not formally established, it is just an empirical observation over millions of acquired ones. In this chapter, we present a literature review of the fingerprint as a biometric modality and its applications in building authentication and recognition systems. We cover a variety of topics starting from a historical overview and acquisition methods to the application of machine learning to acquire fingerprint features and perform fingerprint enhancement. We also discuss various datasets and tools available for fingerprint processing and recent developments in synthetic fingerprint generation.

2.1 Historical Overview

Fingerprint recognition as a form of identification has a history that spans centuries, its origins steeped in the earliest civilizations and evolving over time to become a cornerstone of modern biometric technology. The roots of fingerprint recognition can be traced back to the ancient Babylonians, who used fingerprints as seals on contracts indicating the awareness of the unique and personal nature of the raised patterns on the fingers, around 1955–1913 BCE [11]. Similarly, Chinese documents from the Qin Dynasty (221-206 BCE) depict the use of handprints as evidence during burglary investigations, marking an early instance of fingerprints used for law enforcement [12].

The uniqueness and permanence of fingerprints were scientifically established in the 19th century. Sir William Herschel, a British officer in India, is often credited as one of the first people to recognize the value of fingerprints for identification [13, 12, 7, 14]. He started using fingerprints in 1858 for contracts and prisoners' records to prevent fraudulent practices [11].

Around the same time, across the globe in Argentina, Police Official Juan Vucetich started building the first fingerprint database for criminal cases. One notable success came in 1892 when he successfully used fingerprints to solve a murder case, marking one of the earliest uses of fingerprint evidence in a criminal investigation [15, 14, 7].

In the scientific community, the systematic study of fingerprints began with the work of Francis Galton in the late 19th century. Galton, a cousin of Charles Darwin, was the first to prove the uniqueness and permanence of fingerprints, laying the groundwork for their use in identification. He introduced a fingerprint classification system based on 'Galton pattern types' and also identified minutia points, referred to as 'Galton details' [16, 14, 7].

The early 20th century saw the adoption of fingerprinting techniques by law enforcement agencies worldwide. Edward Henry, an Inspector General of Police in Bengal, India, developed a systematic method of classifying fingerprints based on Galton's observations, which was adopted by the Scotland Yard in 1901 [17, 7, 14]. This method, with some modifications, is still in use today.

As fingerprint recognition technology swiftly gained traction within the realm of forensics, operational fingerprint databases grew exponentially. This surge in data volume soon rendered manual fingerprint identification unfeasible. With the advent of computer technology, Automated Fingerprint Identification Systems (AFIS) began to emerge in the 1970s [14]. These systems enabled digital scanning, storage, and automatic matching of fingerprints, dramatically increasing the speed and efficiency of fingerprint recognition.

In the 21st century, fingerprint recognition has been further revolutionized by the application of machine learning and deep learning techniques, enabling even more accurate and efficient recognition processes. These advanced algorithms can automatically learn and extract relevant features from fingerprint images, paving the way for a new era in biometric recognition.

Today, fingerprint recognition continues to evolve, fueled by ongoing technological advancements. It remains a vital tool for identification and security, widely used across sectors from law enforcement and border control to personal device authentication, demonstrating the lasting impact of this historical biometric modality.

2.2 Acquisition Methods

Fingerprint image acquisition, the initial phase in fingerprint recognition, has been extensively researched and has seen considerable technological advancements over the years. It occurs in two broad categories: online and offline [14]. Offline acquisition involves a process where an individual's fingertip is inked and then imprinted onto a piece of paper. This physical fingerprint is then digitized via optical scanning or high-resolution photography. Such fingerprints are frequently referred to as "rolled fingerprints." An essential subset of offline fingerprint images is latent fingerprints, partial prints retrieved from crime scenes by forensic professionals. Latent fingerprints, however, often present challenges due to their generally inferior quality compared to rolled fingerprints, making them more difficult to process. On the other hand, online acquisition involves digitally capturing the fingerprint in real-time, bypassing the need for physical impressions and subsequent digitization. This method typically leverages technol-

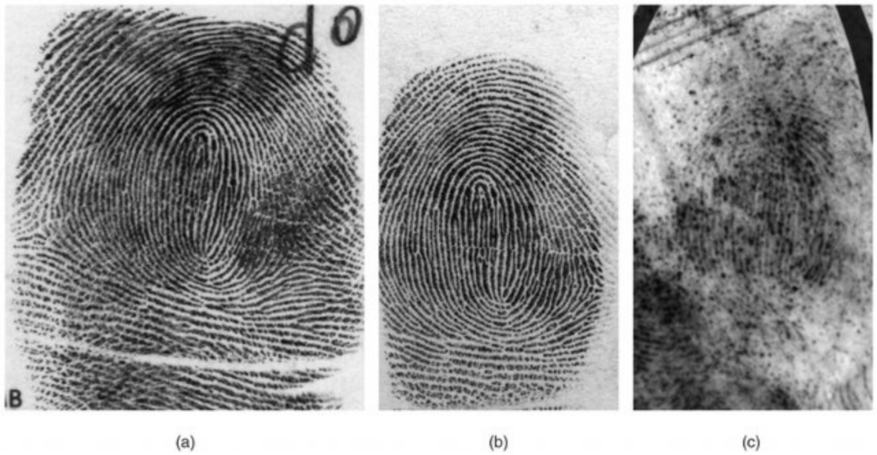

Figure 2.1: Three types of fingerprint images: (a) rolled, (b) plain, and (c) latent fingerprints from the same finger in NIST SD27 [1]

ogy such as capacitive, optical, or ultrasonic sensors to capture the fingerprint image directly, resulting in higher-quality images that are easier to process.

Various acquisition methods have been devised, each with its unique set of advantages and limitations.

1. **Optical sensors** are one of the oldest and most prevalent methods for fingerprint acquisition [18]. They use a glass prism/sheet on which the finger is placed and illuminated on one side of the prism. They operate by capturing the light reflected back from the ridges and valleys of the finger surface in different proportions. While generally accurate, they are susceptible to contaminants such as dust, dirt, and oil that can interfere with image quality [14]. Moreover, optical fibers are also used for optical sensing where the prism and lens are substituted by a fiber-optic platen [19, 20]. This type of optical sensor reduces the packaging size considerably but result in high production cost for large-area sensors [14].

2. **Solid-state Sensors** These are silicon-based sensors that contain an array of elements where each element is a tiny sensor. The advent of mobile technology led to the widespread use of capacitive sensors. They were classified into four types viz. capacitive, thermal, electric, and piezoelectric. Capacitive sensors contain a 2D array of capacitive plates that generate different capacitance values for ridges and valleys of the finger [21, 22, 23, 24, 14]. Thermal sensors work on a similar concept but instead of capacitance, the sensor material generates current based on temperature differences [25, 14, 26, 27]. Electric field sensors, on the other hand, contain a drive ring that generates a radio frequency signal (RF) and contains a matrix of antennas. The signal emitted by the drive ring gets modulated by the finger surface and recorded as the analog response [28, 14]. Piezoelectric sensors are pressure-sensitive sensors that generate electric current when mechanical stress is applied. As ridges and valleys generate different amounts of stress, the fingerprint image is obtained [14, 29].

3. **Ultrasonic Sensors** employ echo and time-delay principles to map the unique pattern of a fingerprint's ridges and valleys. A sound signal is sent to the finger surface and the echo signal is captured to estimate the depth. They can capture high-quality images despite unfavorable skin conditions or environmental contaminants, making them suitable for harsh conditions. However, their high cost and size currently limit their widespread adoption [30, 31, 14, 32]. Qualcomm's ultrasonic in-display fingerprint scanner, used in some smartphones, is an example of this technology.

4. **Emerging Technologies** Emerging technologies, such as multispectral imaging sensors, are revolutionizing the process of fingerprint acquisition. These advanced sensors utilize multiple wavelengths to capture data from both the surface and subsurface layers of skin. As a result, they exhibit a high degree of resistance to conditions that might compromise surface features, such as dirt, wear, or transient alterations in skin state. These sensors are recognized for their reliability, as they can successfully capture fingerprints that other sensors may fail to read [33, 34, 35, 14]. Moreover, the concerns over hygiene and the potential distortion of fingerprints with traditional contact-based methods are driving the ascent of contactless scanners. By leveraging high-resolution cameras or capacitive technologies, these devices offer a non-intrusive and hygienic approach to fingerprint acquisition [36, 37]. Another standout innovation in this space has been the rise of in-display fingerprint sensors. Syncing with the trend of bezel-less smartphones, these sensors—typically optical or ultrasonic— are embedded right within the phone's display [38, 32, 39], heralding the seamless integration of biometric security into everyday technology.

The choice of acquisition method significantly impacts the overall performance and practicality of a fingerprint recognition system. Thus, it remains an active area of research to develop acquisition methods that strike an optimal balance among accuracy, robustness, cost, and user-friendliness.

2.3 Fingerprint Enhancement

Fingerprint enhancement refers to the process of improving the quality of a fingerprint image, which is pivotal in ensuring the accuracy and reliability of a fingerprint recognition system. In practice, due to various external conditions such as dirt, moisture, cuts, or varying pressure during fingerprint acquisition, fingerprint images may not always be clear or distinct. This may result in the following errors: 1) the loss of too many correct minutiae, 2) the presence of too many spurious(false) minutiae, and 3) errors in the locations and orientations of minutiae [7, 40, 14]. These anomalies can greatly affect the fingerprint recognition pipeline. Enhancing these images to increase the clarity of minutiae and accentuate their unique ridge and valley patterns improves the efficiency of the feature extraction and matching processes, thereby improving the overall performance of the system. A fingerprint expert exploits visual cues like local ridge orientation, ridge continuity, etc. to determine the location of minutiae in the

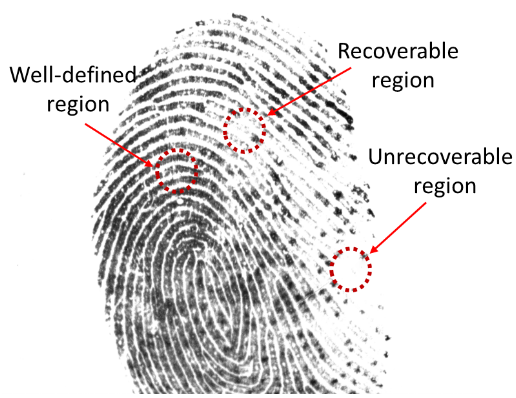

Figure 2.2: Regions of different quality in a fingerprint image from FVC 2000 dataset

fingerprint. In order to automate the process, the algorithm should be able to consider these cues from fingerprint images. [40, 14] divided fingerprint regions into three main categories:

1. Well-defined region: The ridges and valleys are visible clearly and minutiae can be extracted reasonably well.
2. Recoverable region: The ridges and valleys are not very well differentiated as they might be corrupted by creases, cuts, gaps, etc. but are still visible. Neighboring regions provide enough information to reconstruct true patterns of ridges and valleys.
3. Unrecoverable region: The region is highly corrupted such that no pattern is clearly visible and neighboring regions do not provide sufficient information about the true ridge valley pattern.

The first two categories can be grouped into recoverable regions. The aim of the fingerprint enhancement algorithm, therefore, is to enhance the clarity of ridge-valley structure in recoverable regions and remove the unrecoverable regions or at least mark them as too noisy for further processing. Moreover, spurious minutiae or ridges should not be introduced in the process.

Fingerprint enhancement algorithms can be broadly grouped into the following four categories viz filtering, pixel-wise, multiresolution, and deep learning approaches.

2.3.1 Filter-based methods

One of the earliest and most fundamental methods for fingerprint enhancement has been the employment of filter-based techniques. These techniques primarily operate by convolving a filter matrix with the fingerprint image to modify its characteristics, thereby enhancing the important features.

Gabor Filter-Based Technique: A popular choice among filter-based techniques is the Gabor filter, first proposed for fingerprint enhancement by Bazen and Gerez [41]. Gabor filters work by convolving the input fingerprint image with a set of Gabor kernels, which resemble the ridge and valley patterns in a fingerprint. This filter is defined as a sinusoidal plane wave tapered by a Gaussian to capture the periodic and non-stationary nature of fingerprint regions. This characteristic makes them particularly effective at capturing and enhancing local ridge structures [40]. Though it has limitations. It fails to restore the ridge structure influenced by unstructured noise precisely [42].

Gaussian Filter-Based Technique: Gaussian filters have also been employed in fingerprint enhancement for their property of preserving edges while reducing noise and smoothing the image, making it easier to extract meaningful features from the fingerprint [43, 44, 45]. Gaussian filtering helps in reducing high-frequency noise, which can interfere with subsequent processing steps like ridge detection and orientation estimation [46].

Contextual Filtering Technique: Contextual filtering methods consider the context of each pixel (its neighboring pixels) for the enhancement process [47]. Hence, the filter characteristics change according to the local region. In a fingerprint, the sinusoidal-shaped wave of ridges and valleys is defined by a local orientation and frequency that varies slowly across the fingerprint area. This approach, taking into account the context of local orientation and frequency helps to adaptively improve the fingerprint image. Generally, a contextual filter provides a low-pass (averaging) effect along the ridge direction with the goal of removing small gaps, pores, or noise and performs a bandpass (differentiating) effect in the direction orthogonal to the ridges to better differentiate between ridges and valleys and to separate parallel linked ridges [14]. [40] proposed a fast fingerprint enhancement algorithm that adaptively improves ridge clarity by estimating local ridge orientation and frequency. Moreover, based on the fact that convolution in the spatial domain translates to multiplication in the Fourier domain, [48] performed contextual filtering in the Fourier domain.

Anisotropic Filtering Technique: Introduced by Perona and Malik [49], Anisotropic filtering is a sophisticated method that reduces noise while preserving structural edges by considering the image's gradients [50]. This method is highly effective at dealing with poor-quality images with varying types and levels of noise. [51] adapted the algorithm in [40] using a unique anisotropic filter along with using only local orientation. [52, 53] used diffusion techniques with structure-adaptive anisotropic filtering in local feature estimation whereas [54] used local intensity orientation and an anisotropic measure to control the shape of the filter, instead of using the local gradients.

2.3.2 Pixel-wise enhancement

Pixel-wise enhancement techniques modify the pixel values depending on previous values or global parameters without considering the neighboring pixels. These techniques include viz. histogram manipulation, contrast stretching, and normalization. Histogram Equalization (HE) is one of the popular image processing methods for contrast adjustment and it is also commonly used to enhance the visibility and clarity of fingerprints. It flattens out the histogram of the image by stretching out the intensity range by applying the cumulative density operation. This method performs the enhancement considering the global context so highlights the edges and borders in the image, at the cost of local details. So, Adaptive Histogram Equalization (AHE) was introduced which computes histograms for different sections in the image such as to enhance local contrast along with improving edge definitions in the image [55]. How-

Figure 2.3: Histogram equalization techniques applied on FVC2000 fingerprint image. First row corresponds to a) Original image b) Histogram equalization (HE) of the image c) Adaptive Histogram Equalization (AHE) of the original image d) Contrast-Limiting AHE (CLAHE) of the original image. Second row corresponds to output from Otsu thresholding of the corresponding image in the first row.

ever, AHE tends to over-amplify noise in homogeneous regions of the image. To overcome this issue, Contrast Limited AHE (CLAHE) was introduced which operates on small regions (tiles) of the image and combines neighboring ones with bilinear interpolation to remove artificial edges [56]. Figure 2.3 shows the application of all these three variants of HE on a sample fingerprint image from the FVC 2000 dataset [57]. As seen from the figure, CLAHE performs better than the rest. However, these techniques are not effective for noisy images with backgrounds or with extreme degradations like scratches, etc. Moreover, the thresholds for these techniques need to be adjusted for different databases and various levels of noise present. Therefore, these techniques are not very effective for fingerprint enhancement but are used in the initial stages of the fingerprint enhancement algorithm [14].

2.3.3 Multiresolution-based methods

Fingerprint enhancement using multiresolution techniques typically involves the application of algorithms that perform processing at different scales or resolutions. This concept is borrowed from the field of image processing and computer vision, where multiresolution methods have been successfully used to analyze images at different scales, often providing more robust and discriminating features [58]. For fingerprint images, the goal of multiresolution enhancement is to emphasize the relevant fingerprint features such as ridges and valleys at various scales. This often improves the performance of subsequent stages in a fingerprint recognition system, such as minutiae extraction or matching.

Wavelet transform is one of the well-known multiresolution techniques. It is a mathematical tool that decomposes an image into a set of scaled and translated versions of a wavelet function, thereby providing a multiresolution representation [59]. For fingerprint images, wavelet transform techniques can be used to emphasize the ridge structures and suppress the noise at various scales. In [60], Lee et al. used wavelet transform for decomposing fingerprints into smaller directional images and estimated coherence and orientation for fingerprint recognition. Later, Hsieh et al. [61] used wavelet transform to develop fingerprint enhancement by exploiting multiresolution analysis of global texture and local orientation of fingerprints. In their work, Almansa et al. [52] present a dual-technique system for fingerprint image processing that combines shape-adapted smoothing with automatic scale selection over multiple scales, enabling detailed resolution of fingerprint features while joining up the interrupted ridges as well. [62] proposed a dyadic-scale space technique where the image is decomposed into a series of smaller images at different scales to analyze coarse and finer details and combine creditable information for enhancement. Fronthaler et al. used a Laplacian-like image pyramid for image decomposition and contextual filtering on smaller images [63].

2.3.4 Deep learning approaches

Convolutional Neural Networks (CNNs) have significantly transformed the landscape of computer vision and artificial intelligence by introducing a robust and efficient means to learn from and interpret images. Inspired by the human visual system, CNNs possess an intrinsic ability to automatically and adaptively learn spatial hierarchies of features from the data they are provided. Furthermore, CNNs have been extensively used in various enhancement and restoration tasks, such as super-resolution, denoising, and inpainting, where they have demonstrated their ability to recover and enhance fine details in images. In the realm of fingerprint enhancement, the advent of deep learning techniques has brought forth significant improvements in the quality of enhanced fingerprint images and in the robustness of the enhancement process itself. In the previous works, [64] used CNNs to estimate orientation fields for fingerprint classification. Later, [65] proposed a deep network architecture called FingerNet with a multi-task learning approach to perform fingerprint enhancement along with orientation estimation.

In their work, Qian et al. [66] introduced a fingerprint enhancement approach using a deep network called DenseUNET to improve image quality in a pixel-to-pixel and end-to-end manner. More recently, researchers have been exploring the utility of generative models, particularly generative adversarial networks (GANs) [67], for fingerprint enhancement. [68] incorporated adversarial training using GANs for fingerprint enhancement. In another interesting work, [69] posed the fingerprint denoising problem as a segmentation (task) using M-net-based architecture. Despite the advances achieved so far, deep learning-based fingerprint enhancement is still a rapidly evolving field. Current research efforts are focused on overcoming the limitations of existing models, such as their high computational requirements and their occasional tendency to over-enhance certain features, creating artifacts that can hinder the subsequent recognition process.

2.4 Fingerprint Features

Feature extraction is a crucial stage in the process of fingerprint recognition. This step involves identifying distinctive attributes from fingerprint images that can be utilized to distinguish between different individuals. Over time, two broad categories of features, global and local, have been employed.

2.4.1 Global Features

Global features capture the comprehensive characteristics of a fingerprint, including elements such as ridge flow and ridge curvature. These features serve a pivotal role in fingerprint classification, which categorizes fingerprints into various categories, namely, arch, loop, and whorl [70, 6, 71, 72]. This classification can substantially decrease the search space and time in vast databases, thus streamlining the matching process to a specific group.

Historically, the initial approaches of feature extraction concentrated predominantly on these global features. Significant contributions in the field of fingerprint classification, particularly focusing on global features, were made by Maio and Maltoni [73], as well as Sherlock et al. [48]. Their innovative work served as a foundation for the evolution of more sophisticated fingerprint recognition systems, by underscoring the importance of global feature-based classification.

2.4.2 Local Features

Local features capture more granular, detailed aspects of the fingerprint, primarily in the form of local ridge orientation, frequency, and minutiae.

Ridge Orientation and Frequency: Ridge orientation refers to the directional flow of the ridges, and ridge frequency denotes the number of ridges present in a local region of the fingerprint [73]. Together, these two attributes form the basic texture of the fingerprint and are useful for both global pattern flow analysis and local minutiae extraction. Algorithms like Gabor filters and Fourier analysis have been commonly used to estimate these attributes [48].

Minutiae: These are the distinct point features observed where the ridges terminate or bifurcate. Minutiae, particularly ridge endings and bifurcations, form the most discriminative feature set used in fingerprint recognition [71, 72]. Each minutia can be defined by its position (x, y coordinates), type (ending or bifurcation), and orientation (the direction of the ridge on which it is located) [40, 14].

Traditional minutiae-based methods represent fingerprints as a set of minutiae points, making them efficient for matching purposes. However, they can be susceptible to issues when the fingerprint quality is poor or the number of minutiae points is low [74, 14].

To overcome these challenges, researchers have expanded the local feature set to include ridge shape information, sweat pores, and third-level minutiae such as ridge contours and edge shapes. The inclusion of these additional features has been shown to increase the discriminatory power of the feature set, especially when minutiae information is scarce [75, 76, 40, 77].

2.4.3 Deep Learning-Based Feature Extraction

Deep learning has shown to be remarkably effective in automated feature extraction, and fingerprint analysis has greatly benefited from this approach. The traditional technique of defining and extracting global and local features is challenging, error-prone, and often does not capture all the intricate patterns of fingerprints. In contrast, deep learning models are capable of automatically learning features directly from raw data, leading to more robust and discriminative features for tasks such as fingerprint matching and recognition.

Among the early applications of deep learning for fingerprint feature extraction is the work [78] where the authors designed a CNN-based two-stage architecture named MinutiaeNet that first detects coarse minutiae points followed by refining their localizations. Their work exhibited superior performance on latent fingerprints over traditional approaches, especially in handling poor-quality fingerprint images. Following this, [79] proposed a Convolutional Neural Network (CNN) based model that used a set of 'n' local nearest neighbor minutiae features to generate rotation-scale invariant feature vectors. They used hash indexing to reduce the number of retrievals in matching and compared their work against various conventional approaches on several benchmark datasets. Subsequently, several works also explored the use of deep learning for extracting higher-level fingerprint features. For instance, Darlow et al. [80] employed a Fully Convolutional Network (FCN) for robust minutiae extraction. They devised a network called MENet that detects minutia as probabilities and uses post-processing to refine them further. [81] proposed a unified network based on CNNs to extract various fingerprint representations including minutia and orientation from fingerprints along with segmentation and enhancement.

These advancements highlight the promising potential of deep learning for fingerprint feature extraction. However, there is still room for further research to improve the robustness of these models, especially in the presence of noisy or incomplete fingerprint data.

2.5 Fingerprint Matching

Fingerprint matching is a pivotal process in fingerprint recognition, fundamentally designed to compare and measure the similarity between two fingerprint images. Over the years, this process has evolved, encompassing a range of techniques, each with its advantages and challenges.

2.5.1 Minutiae-Based Matching

Early approaches to fingerprint matching relied heavily on minutiae-based methods. Minutiae, primarily ridge endings, and bifurcations, are extracted from the fingerprint and compared between prints. These methods leverage the uniqueness and permanence of minutiae points to achieve effective identification. A large number of previous works have been published that have [82, 83, 14, 84, 85, 86] significantly contributed to the development of these methods, formulating robust algorithms for minutiae extraction and matching. Jain et al. provided a methodical approach to online fingerprint verification, elucidating both the minutiae extraction and the subsequent matching algorithm [87].

Despite its robustness, minutiae-based matching isn't devoid of challenges. Scenarios with low-quality fingerprints, partial prints, or those that are rotated or scaled require refined algorithms for accurate matching. Furthermore, precise alignment of prints for minutiae comparison is complex, often requiring significant computational resources. Noteworthy in this context is the alignment-based approach by Jiang et al., which incorporates both local and global minutiae structures to enhance matching accuracy [75].

2.5.2 Correlation-Based Matching

To address some limitations of minutiae-based methods, researchers introduced correlation-based matching techniques. These methods compare whole regions of fingerprints, calculating the degree of similarity or correlation [88, 89, 90]. These techniques can tolerate some distortion or noise, offering more robustness against low-quality prints.

Despite their robustness, correlation-based methods have their own limitations. They are sensitive to the alignment and scale of the fingerprint images, and their computational efficiency can be challenged when dealing with large databases.

2.5.3 Texture and Image-Based Matching

As an alternative to minutiae and correlation-based methods, researchers have explored texture and image-based matching techniques. These methods leverage texture patterns, ridge flow, or frequency within the fingerprint image to generate match scores. [91, 86, 92, 93] have made significant advancements in this area.

2.6 Fingerprint Datasets and Tools

The availability of large public-domain fingerprint datasets is crucial for the development of fingerprint biometrics as it enables training, evaluation, and benchmarking of different techniques. The National Institute of Standards and Technology (NIST) introduced the first large-scale public domain dataset which constituted an excellent benchmark for automatic fingerprint identification systems development and fingerprint classification studies [94, 14].

NIST introduced several databases for fingerprints including Special Databases viz. SD 4 [95], SD 14 [96], SD 27 [97] etc over the years which contain rolled impressions and latent prints as well as other data like handprinted forms. NIST Special Databases 300, 301 [98], and 302 [99] stand out for their significance in fingerprint-related research. NIST SD 300 was aimed at fostering research in the domain of fingerprint segmentation. It consists of a set of fingerprint images that vary in quality, with the intention that researchers would develop and evaluate algorithms to effectively segment these images. NIST SD 301 was designed to promote advancements in minutiae extraction from fingerprints. NIST SD 302 was collected as part of Nail to Nail (N2N) Fingerprint Challenge which is a collection that caters to the larger fingerprint recognition community by providing a diverse set of fingerprint samples. It contains various types of fingerprint impressions, like plain, rolled, and touch-free, which have been captured from an array of devices. This database contains thousands of fingerprint images acquired from different auxiliary devices. This variety is especially valuable for researchers aiming to develop algorithms robust to varying acquisition conditions. The database is extensive and is one of the standard datasets used for evaluating fingerprint recognition systems. The fingerprint images are offered in different resolutions and capture conditions, emphasizing the diversity of real-world scenarios. Currently, only the SD 302 database is available in the public domain which is suitable for our work. So, we utilize this dataset for training and evaluation purposes in our thesis.

The Fingerprint Verification Competition (FVC) datasets serve as a benchmark in the biometric research community, encompassing a diverse collection of fingerprint samples gathered under various conditions to evaluate and compare fingerprint recognition algorithms. FVC campaigns (2000, 2002, 2004, and 2006) were organized with the aim of providing fingerprint databases to any interested researcher and tracking the performance of the state-of-the-art fingerprint matching algorithms. Fortunately, most of the authors now report the results of their experiments on one or more of these databases according to the proposed protocol, thus producing results that can be compared across the whole scientific community. We have included the complete FVC 2000 [57], FVC 2002 [100], and FVC 2004 [101] databases in our evaluation experiments. Each FVC database has 110 identities with eight impressions each, taken from four different sensors.

Apart from the databases, NIST provides a comprehensive set of tools designed for biometric research and fingerprint matching [94, 102, 103]. One of these tools is MINDTCT [102] which is used for minutiae detection in fingerprint images. It processes the input fingerprint image to extract relevant features, specifically, minutiae points, which play a crucial role in fingerprint recognition. BOZORTH3 [102] is a minutiae-based fingerprint-matching algorithm. After the extraction of minutiae points from fingerprint images using MINDTCT or any other extraction method, BOZORTH3 is employed to match these points between different fingerprint samples and produce a similarity score. The higher the score, the more likely the two fingerprints are from the same individual. Moreover, NFIQ2 [103] is another utility that scores fingerprint images based on fingerprint quality. Overall, NBIS offers a rich set of utilities that can be integrated into various biometric systems or used in research projects to analyze and process biometric data, especially fingerprints.

2.7 Synthetic Fingerprint Generation

The generation of synthetic fingerprints has emerged as a valuable tool in the field of fingerprint recognition where fingerprint data is scarce. It involves the use of computational algorithms to create artificial fingerprint images that bear the same fundamental characteristics as their natural counterparts.

One of the primary motivations for generating synthetic fingerprints is to supplement the training of machine learning and deep learning models. Given the necessity of extensive datasets for these models to learn effectively, synthetic fingerprints can help expand and diversify these datasets without the need for additional real-world data collection. Synthetic fingerprints can also assist in stress-testing the robustness of recognition algorithms. By generating fingerprints with specific types of distortions or abnormalities, researchers can gain insights into how well their recognition systems cope under various challenging conditions.

Historically, synthetic fingerprint generation methods were based on procedural algorithms that constructed fingerprints from basic ridge elements [104]. SFinGe, or Synthetic Fingerprint Generator [105], is a widely-used software for the generation of synthetic fingerprint images. It employs a procedural approach to create large databases of synthetic fingerprints along with associated ground truth data. SFinGe has been designed to replicate the global statistical features found in collections of real-world fingerprints. It is especially useful for algorithm testing and benchmarking, and it has been adopted in numerous studies for generating databases of synthetic fingerprints. These methods, while effective in creating structurally coherent fingerprints, often failed to replicate all the complex and varied patterns found in real-world data [101].

Anguli [106] is another software for synthetic fingerprint generation that is deployed at the Database Systems Lab, Indian Institute of Science. It is a freely available tool that generates synthetic fingerprints using SFinGe algorithms. In our experiments, we have made use of SFinGe and Anguli tools to add data for training.

Recent years have witnessed the application of deep learning for synthetic fingerprint generation, with Generative Adversarial Networks (GANs) [67] being a particularly promising avenue. Various papers have proposed different architectures and techniques to improve the realism and quality of the generated fingerprints. PrintsGAN [107] and Finger-GAN [108] papers further expanded on this theme by devising specialized GAN architectures optimized for fingerprint generation, demonstrating impressive results in terms of realism and diversity. A recent work introduced a lightweight GAN network focusing on the development of a streamlined architecture that can generate high-quality fingerprints with less computational requirements [109]. The recent works including HQ-finGAN [110] applied the CycleGAN [111] based generative architecture for hyper-realistic fingerprint synthesis [112]. SynFi [113] utilizes GAN for synthetic fingerprint generation followed by the Super-Resolution [114, 115] to synthesize fine-grained textures. Lastly, the recent work [116] presented a comprehensive study on generating high-fidelity fingerprints while maintaining the uniqueness of the generated prints and preserving privacy.

Chapter 3

Fingerprint Enhancement with U-Net-based architecture

3.1 Introduction

Fingerprint recognition stands as one of the most widely used and researched biometric modalities, owing to its unique characteristics that differentiate individuals and its longevity over time. Its applications span a wide range of sectors, from criminal justice systems to smartphone security. Despite its ubiquity and historical roots, the task of enhancing and reliably recognizing fingerprints from diverse and often challenging environments remains a forefront challenge.

A predominant portion of these challenges arises from the intrinsic variability of fingerprint impressions, including factors like different skin conditions, pressure variations during acquisition, and the presence of artifacts such as dirt or smudges. Furthermore, many real-world scenarios produce low-quality or partial prints, underscoring the essential nature of the enhancement step for achieving dependable recognition in subsequent stages.

The pursuit of advanced fingerprint enhancement techniques is motivated by two primary objectives: enhancing the recognition accuracy and maximizing the extraction of discriminative features from fingerprints of sub-optimal quality. Traditional methods involving filtering techniques and other image processing methods [117, 51, 40, 118, 119], while effective to an extent, often fall short when dealing with the diverse range of imperfections present in real-world datasets. This realization has fueled the pursuit of more adaptive and data-driven approaches, specifically harnessing the power of deep learning.

Deep learning, with its demonstrated prowess in various image processing tasks, offers an appealing avenue for fingerprint enhancement [65, 66, 68]. By modeling intricate and complex relationships, deep learning methods possess the potential to recognize and refine intricate patterns and nuances in fingerprint data, thus providing a holistic solution. The motivation behind this chapter, therefore, is to delve deep into a neural fingerprint enhancement technique, exploring its intricacies, strengths, and areas of improvement.

In the following sections, we will delve into the intricacies of our proposed model architecture, showcasing its modified elements and design principles in order to support the fingerprint enhancement task. We will investigate variations of the U-Net structure, study the impacts of different architectural components, and provide comprehensive experimental results to substantiate the efficacy of the proposed method. Through rigorous experimentation, we aim to shed light on the potential of deep learning in revolutionizing the domain of fingerprint enhancement.

3.2 Finger-UNet - Proposed Approach for Fingerprint Enhancement

The challenge of fingerprint enhancement bears a resemblance to the broader domain of image denoising. However, the intrinsic characteristics of biometric data set it apart, emphasizing that fingerprints must not be approached in the exact same manner as conventional real-world images. While autoencoder-style architectures have garnered significant acclaim for image-denoising endeavors, the uniqueness of fingerprints necessitates a specialized approach.

We proposed Finger-UNet—a multi-task architecture, drawing inspiration from the popular deep architecture U-Net [2]. Originally envisioned for biomedical image segmentation, U-Net's versatility has been underscored by its impressive performance in fingerprint enhancement, as evidenced by studies such as [66] and [120]. This chapter introduces a tailored rendition of U-Net, specifically honed to manage fingerprint data adeptly, ensuring its optimal enhancement.

In the following topics, we will discuss the components in the originally proposed U-Net architecture and slowly build up our specific elements on top of it.

3.2.1 Vanilla-UNet

U-Net is a convolutional neural network that was originally developed for biomedical image segmentation [2] and its architecture is characterized by its symmetric shape, which gave it the 'U' name.

1. Downsampling (Contracting) Path: The architecture starts with an input layer where the initial image (or feature map) is fed. This is followed by a series of convolutional layers, each typically followed by a rectified linear unit (ReLU) activation function and a max-pooling operation. As you progress down this path, the spatial dimensions of the feature maps reduce while the depth (number of channels or feature maps) increases. This downsampling process allows the network to abstract the features of the image, encoding the most important information.

2. Bottleneck: After the contracting path, the architecture has a few more convolutional layers, allowing the network to store representations of the input image.

3. Upsampling (Expansive) Path: Post bottleneck, the network starts the upsampling process, where the spatial dimensions of the feature maps are increased. This is achieved using transposed convolutions or upsampling operations. After each upsampling step, the feature map from the corresponding step in the contracting path is concatenated. This skip-connection architecture ensures that the localization information from the downsampling path is combined with the abstracted features of the upsampling path.

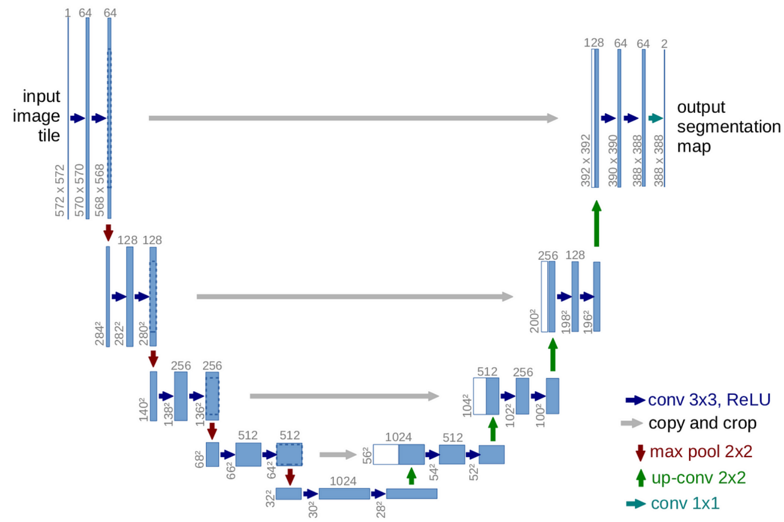

Figure 3.1: The U-Net architecture [2]

4. Final Layer: At the end of the expansive path, a final convolutional layer is used to map the output to the desired number of classes (in case of classification) or desired output size.

One of the key features of U-Net is its ability to combine the location information from the down-sampling path to precisely localize and segment features in the upsampling path. This is especially crucial in biomedical image segmentation where precise boundaries are needed.

The advantage of U-Net comes from its efficiency. The skip connections allow it to use fewer parameters than a fully convolutional network, making it faster and more memory-efficient, while still producing a high segmentation accuracy. This architecture, while originally designed for biomedical images, has since found applications in a wide range of image segmentation problems across different domains.

3.2.2 Wavelet Transform as Pooling

Generally, CNNs work with images in the spatial domain, considering the image as a matrix of values called pixels. In contrast, the frequency domain deals with how these pixel values change in the spatial domain. Several mathematical transforms exist in the frequency domain, including Fourier, Laplace, Z, and wavelet transform.

Wavelet transform can be said as a transformation that maps the signal to a multi-resolution representation. It is widely used in signal processing applications [121, 122], image denoising [123, 124, 125, 126], and compression [127, 128, 129]. It decomposes the image information into signal details and approximations, commonly known as high-frequency and low-frequency components. Previous studies show several attempts have been made to incorporate Discrete Wavelet Transform (DWT) into CNNs [130, 131, 132, 3]. [130] showed that CNNs could benefit from learning on wavelet subbands and proposed a wavelet residual network (WavResNet). [133] applied dual-tree complex wavelet transform (DT-CWT) and designed a Convolutional-Wavelet Neural Network (CWNN) to suppress noise and extract features robustly from SAR images. In [131], the authors proposed a multi-level wavelet CNN (MWCNN) model that integrates wavelet transform into CNN to reduce feature map resolution and increase receptive field. In another work, [132] designed DWT/IDWT layer for integration into deep networks. Later, [3] modified these layers to include a wavelet attention module to retain detailed information in high-frequency components in DWT. In our approach, we build on the works of [132] and [3] to use wavelet transform in our U-Net architecture. In the domain of fingerprint biometrics also, research has been carried out to show the effectiveness of wavelet transform for fingerprint enhancement task [134, 61].

In [132], Li et al. proposed DWT layers that decompose a 2D image into its frequency components. Here, the pooling is performed with a down-sampling operation instead of max pooling or average pooling to avoid information loss and aliasing effect. It also increases the noise robustness of CNNs.

Given 2D data X, the DWT usually does 1D DWT on every row and column, resulting in four frequency components X_{LL}, X_{LH}, X_{HL}, and X_{HH}. X_{LL} is the low-frequency component of input X, representing the main information, including the basic structure in the image; X_{LH}, X_{HL}, and X_{HH} are three high-frequency components that save the horizontal, vertical, and diagonal details of X, respectively. [132] designed these DWT/IDWT layers in Pytorch and made DWT/IDWT operations differentiable and compatible with CNNs.

$$\begin{gathered} X_{LL} = LXL^T, X_{LH} = HXL^T, \\ X_{HL} = HXL^T, X_{HH} = HXH^T, \end{gathered} \tag{3.1}$$

where matrix L and H are the cyclic matrix composed of wavelet low-pass filter $\{l_k\}_{k\in Z}$ and high-pass filter $\{h_k\}_{k\in Z}$ respectively. Here L and H are as below [132]:

$$L = \begin{pmatrix} \cdots & \cdots & \cdots & & \\ \cdots & l_0 & l_1 & \cdots & \\ & \cdots & l_0 & l_1 & \cdots \\ & & & \cdots & \cdots \end{pmatrix}; H = \begin{pmatrix} \cdots & \cdots & \cdots & & \\ \cdots & h_0 & h_1 & \cdots & \\ & \cdots & h_0 & h_1 & \cdots \\ & & & \cdots & \cdots \end{pmatrix} \tag{3.2}$$

In Finger-UNet, we use the wavelet attention block (WA Block) proposed in [3] built by modifying the above DWT layer. The wavelet attention block can be defined as

$$x_g = \sigma(f(X_{LH}, X_{HL})) \tag{3.3}$$

$$A_m = X_{LL} * x_g \tag{3.4}$$

$$Z = f(X_{LL}, A_m) \tag{3.5}$$

where f represents the feature aggregation function, σ denotes the softmax function. Once the input is decomposed into four frequency components using DWT, the WA block takes the horizontal feature

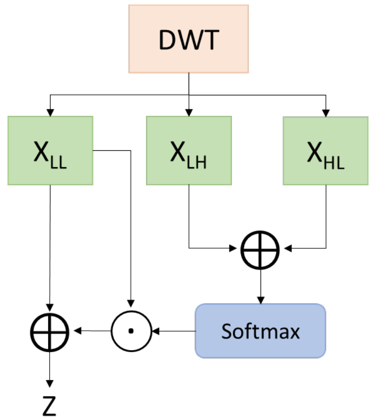

Figure 3.2: Wavelet Attention Block [3]

X_{LH} and vertical feature X_{HL} and aggregates them by element-wise addition as a global detail feature. Then this feature is normalized using the softmax function. The normalized feature x_g and the low-frequency component X_{LL} are used to generate the attention map A_m through element-wise multiplication. Finally, the original low-frequency component X_{LL} is added to the attention map A_m by element-wise operation and given as output Z. As the X_{LL}, X_{LH}, X_{HL} components account for most of the information and the X_{HH} component does not contain any additional information for fingerprint ridges, X_{HH} is not used in WA block.

Figure 3.2 shows the diagram of the WA block. We replace the max-pooling layers in vanilla U-Net with this WA block and notice an improvement in the fingerprint enhancement task. IDWT layer is used to reconstruct the output back to spatial domain [132].

3.2.3 Reconstruction Loss

Loss functions play a pivotal role in training deep learning models, guiding the optimization process to converge to a solution that minimizes the error. Many earlier works [135, 136] used mean squared error (MSE) or L_2 loss for reconstruction in image enhancement tasks. While L_2 loss has been a conventional choice for regression problems due to its sensitivity to outliers and its smooth gradient, it might not always be the optimal choice. [137] pointed out several limitations to using L_2 for image restoration tasks. L_2 does not correlate well with human perception of image quality [138], due to the assumption that the impact of noise is independent of the local characteristics of the image. In contrast, the human visual system (HVS) is more sensitive to luminance and color variations in texture-less regions [137, 139]. Moreover, L_2 tends to penalize large deviations more heavily than small ones, potentially causing a model to under-emphasize minor but important details in the image. [137] suggested using L_1 loss instead of L_2 as it does not over-penalize large errors as it treats deviations in linear fashion and it also helps reduce artifacts introduced by L_2. We utilize the effectiveness of using L_1 as a reconstruction loss for fingerprint enhancement as it might be more suitable for preserving and enhancing intricate

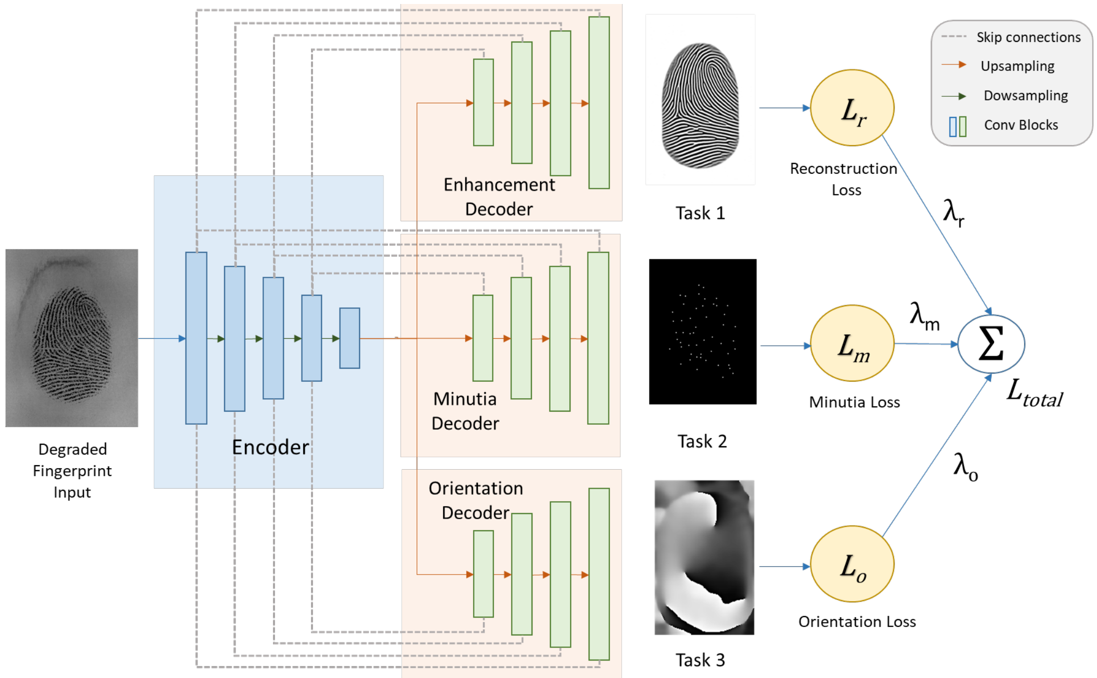

Figure 3.3: Architecture of our proposed approach with enhancement, minutia detection, and orientation estimation branches in a multi-task learning setting

details present in fingerprint patterns. Even though the images are grayscale, we witness a performance improvement compared to the L_2 counterpart, as demonstrated in later sections.

$$L_2(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3.6}$$

$$L_1(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{3.7}$$

where y is the true value, $\hat{y}$ is the predicted value, n is the number of samples or data points.

3.2.4 Domain Knowledge Incorporation with Minutia Detection and Orientation Estimation Tasks

Fingerprints are unique in their nature. Beyond their distinguishing patterns like whorls, loops, and arches, they possess an intricate ridge-valley structure accompanied by a specific frequency. This uniqueness is underscored by specific features such as ridge endings and bifurcations or minutiae, which play a pivotal role in fingerprint matching. Contrasting sharply with conventional real-world images, fingerprints require a differently tailored approach. It is important that their intrinsic properties are not neglected during the learning process. Infusing domain knowledge into our network's training aids in capturing the nuances of fingerprint data more effectively. A simplistic image enhancement

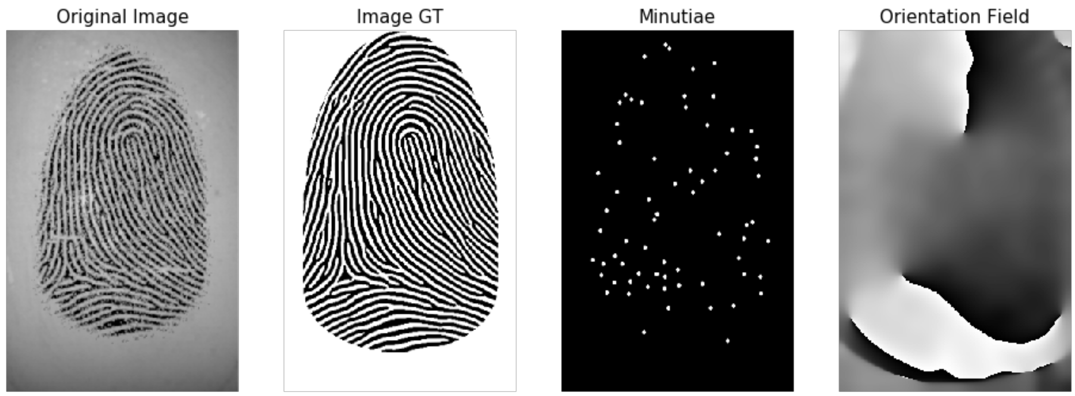

Figure 3.4: a) Degraded image from SFinGe dataset b) Enhancement GT c) Minutia branch GT created from the image GT d) Orientation branch GT created from image GT

method might overlook these subtleties, potentially leading to suboptimal fingerprint reconstructions. By weaving in domain knowledge, specifically through minutia prediction and orientation estimation, we aspire to bridge this gap. We propose to optimize these tasks along with fingerprint enhancement that enables the model to not only recognize but also prioritize the enhancement of these defining features.

When the model is conditioned to recognize and reconstruct the ridge-valley structure, combined with minutiae prediction, it becomes attuned to the very nuances that make fingerprints uniquely identifiable. This strategic integration ensures that the enhancement process isn't just about improving image quality, but about preserving and accentuating the essential features that are critical for subsequent fingerprint recognition tasks.

Moreover, in scenarios where fingerprint images are of subpar quality, the model, equipped with this added domain knowledge, can act with informed intuition. Instead of merely trying to enhance what's visibly present, it can infer and reconstruct based on the inherent properties and structures typical of fingerprints. This is particularly beneficial for fingerprints that might have been compromised due to smudging, partial prints, or other forms of distortions, ensuring that the output is not just clearer but also more representative of an authentic fingerprint

Orientation Estimation: Historically, the importance of ridge orientations in fingerprint enhancement has been underscored in several studies [40, 118, 119, 65]. The orientation estimation essentially indicates the gradient direction of a fingerprint segment, proving instrumental in the enhancement process. For instance, [65] demonstrated the potential of harnessing orientation knowledge by training enhancement and orientation estimation branches simultaneously. In our work, we evolve this idea further. With a shared U-Net encoder, we introduce an orientation estimation branch that predicts vectorized orientation fields—specifically, sine and cosine values (Figure 3.4 d)). This method sidesteps the constraints of categorizing orientation patches into fixed sets, as seen in [65], offering a more nuanced estimation of ridge orientation.

Minutiae Prediction: Further enriching our approach, we assimilate a minutia detection branch to amplify the domain knowledge. Drawing inspiration from [80], which introduced the minutia extraction network (MENet) that generated a minutiae probability map later refined for accurate minutia location detection, we adopt a similar strategy. Our approach translates this knowledge into a grayscale image, where minutiae locations are distinctly marked by white dots against a contrasting black background, as depicted in the minutia branch (Task 2) image in Figure 3.3.

Capitalizing on the strength of multi-task learning, our methodology concurrently optimizes fingerprint enhancement, minutia detection, and orientation tasks. The architectural foundation consists of a shared encoder extracting features from the noisy fingerprint inputs and individual decoders tailored for each specific task, as illustrated in Figure 3.3. By adopting this approach, we show that the model assimilates the intrinsic ridge-valley structure of fingerprints while concurrently honing its ability to predict minutiae. As a result, the model's emphasis extends beyond simple image enhancement to a more nuanced refinement of the fingerprint structure within the image.

We use the L_1 loss (Equation 3.7) for the enhancement task denoted by L_r. As we are solving orientation estimation as a regression problem, we use L_2 loss or MSE (Equation 3.6) for this branch. We denote this loss by L_o. For the minutia detection branch, the values in the ground-truth map are zero or one based on the presence of minutia locations (Figure 3.4 c)), hence we use binary cross entropy as the loss function and it is denoted by L_m. The final loss L is the summation of L_r, L_m, and L_o. To balance the different losses of the three branches, we weigh them by scalars λ_r, λ_m, and λ_o respectively.

$$L_{total} = \lambda_r L_r + \lambda_m L_m + \lambda_o L_o \tag{3.8}$$

3.2.5 Depthwise Separable Convolution for Efficient Operation Approximation

Conventional convolution operations engage both spatial and channel-wise interactions by executing multiplicative operations across various spatial pixels and the entirety of the channels. Depthwise separable convolution, introduced by Chollet in 2017 [140], aims to decompose these interactions. It achieves this by ensuring each filter channel operates exclusively on a singular input channel. Following this depthwise operation, a 1x1 convolution, often termed as pointwise convolution, is applied to manage the depth dimension. This approach, although an approximation of the traditional convolution mechanism, typically maintains competitive performance levels. Notably, by adopting depthwise separable convolutions in our experiments, we observed a reduction of approximately 30% in the model parameters. This substantial decrease not only simplifies the model but also provides a countermeasure against overfitting. As a result, our model architecture integrates depthwise separable convolutions in lieu of standard convolution layers.

3.3 Experimental Setup

3.3.1 Dataset

While there exists a number of publicly available fingerprint datasets, they often suffer from two limitations. Firstly, their volume is insufficient for training the vast parameters of deep neural networks, and secondly, they might lack enhanced or clean impressions to be used as ground-truth. Addressing this challenge, we resort to synthetically generated fingerprints produced by SFinGe [141] for our experiments. This tool offers the flexibility to generate fingerprints with diverse ridge structures, patterns, and a customizable number of impressions. Furthermore, it's adept at simulating various forms of noise and degradations — from skin elasticity, external noise, and applied pressure to physical imperfections such as scratches. For this work, we created a dataset comprising 10,000 training fingerprint pairs (each consisting of a degraded image and its corresponding ground-truth), 1,000 pairs for validation, and 3,000 pairs earmarked for testing. The curated data encapsulates a gamut of noises and is representative of different backgrounds (spanning optical, and capacitive sensors, to neutral backgrounds). A visual snapshot of this dataset, displaying sample input and ground-truth pairs, can be found in Figure 3.5, showcased in the initial two rows.

For the evaluation phase, we harness the strengths of two well-known fingerprint datasets: FVC 2002 [100] and NIST Special Database (SD) 302 [99]. The FVC 2002 dataset encompasses fingerprints sourced from optical and capacitive sensors and even includes synthetic fingerprints, distributed across four sets. Each set makes 80 images publicly accessible. NIST SD302, on the other hand, provides a rich collection of plain, rolled, and touch-free impressions, sourced from a multitude of devices. For our purposes, we utilize subset 302d, which boasts 5141 fingerprint images procured from four distinct auxiliary devices.

3.3.2 Training

We use the PyTorch framework for all experiments in this work. We apply various data augmentations like random rotation, and translation, for the network to generalize well. We apply the same degree of augmentations to each input-ground truth pair for consistency. The images are resized to fixed dimensions of 400x256. The hyperparameters are chosen using grid search. The model is trained with Adam optimizer with a learning rate of 0.001. The batch size is set as 32. The loss weights λ_r, λ_m, and λ_o are 0.8, 0.1, 0.1 respectively. The network is trained on four GPUs in data parallel mode, and each GPU is NVIDIA GeForce TITAN X with 16 GB RAM. As the training is end-to-end, the losses from minutia and orientation guide the enhancement branch for better fingerprint output.

Figure 3.5: Enhanced images (bottom row) with corresponding degraded input (top row) and ground-truth (middle row) with our approach on the SFinGe synthetic test set. SSIM is reported between each enhanced and ground-truth pair

(a) FVC 2002 dataset

(b) NIST SD302 dataset

Figure 3.6: Illustration of enhancement results with our approach on samples from test datasets

3.4 Results and Analysis

3.4.1 Performance Evaluation

For evaluation, we make use of standard metrics like SSIM, RMSE, and PSNR [69]. Additionally, we utilize the NFIQ2 [103] package from NIST's NBIS [102] to measure the quality of fingerprint images. Quality scores can range from 1 to 100. Moreover, we present the average matching scores of genuine pairs on all four subsets of FVC2002 using BOZORTH3 [102]. SD302 does not contain multiple impressions of a finger, so matching performance cannot be obtained.

3.4.1.1 Ridge Structure Preservation

In Figure 3.5, we show a few sample images from our SFinge test set with corresponding ground-truth and enhanced images. We see the SSIM values are higher which suggests that our approach tries to preserve the ridge structure while performing enhancement on degraded input. In addition to this, Table 3.1 reports SSIM values for various combinations of techniques suggested in this work. In Figure 3.6, we show the results of our approach on both datasets.

3.4.1.2 Fingerprint Quality Analysis

We report the average NFIQ2 scores on the test sets FVC 2002 and NIST SD302 in Table 3.2. From the results, we say that the fingerprint quality improved by a significant amount of 58% in the case of FVC 2002, whereas it improved by 23% in the SD302 dataset after enhancing the raw images with our approach. Further, our approach gives comparable results with the previous works. Moreover, we also present the NFIQ2 scores on the SFinGe test set in Table 3.1 which supports our approach to use WA Block, depthwise separable convolutions, and domain knowledge.

3.4.1.3 Matching Performance

We report and compare the average matching scores of genuine pairs of our approach with raw images and previous works in Table 3.3. The results suggest our approach with the inclusion of domain knowledge from the minutia and orientation branch is able to retain the minutiae from the degraded images which increases the matching score and performs well in comparison to earlier works. This suggests the effectiveness of our approach in the pipeline of fingerprint matching.

3.4.1.4 Ablation Study

We report the evaluation metrics for the techniques discussed in the chapter as a part of the ablation study in Table 3.1. We show how each discussed method helps to improve the enhancement further. The depthwise separable convolution does not significantly impact the performance but helps reduce the

Figure 3.7: Illustration of failure scenarios observed with our approach. Top row represents degraded input and the bottom row denotes the enhanced images with our model.

model parameters to a large extent. Multi-task learning with minutia detection and orientation branches gives a clear performance improvement. Moreover, the use of wavelet attention block further improves the SSIM and NFIQ2. Overall, the combination of these techniques results in the best-performing model.

During the early efforts developing the approach, we experimented with dataset provided as part of challenge 'Fingerprint inpainting and denoising (WCCI'18, ECCV'18)' This dataset contains synthetic fingerprints generated with Anguli-Synthetic Fingerprint Generator and added random artifacts(blur, brightness, contrast, elastic transformation, occlusion, scratch, resolution, rotation) and backgrounds to the ground-truth fingerprint images. We noticed significant results with our approach on this dataset as shown in Figure 3.8. Even with highly degraded images, the model showed impressive results. As the dataset is quite unrealistic to real-world images, we did not consider using it further in our experiments, so we do not report the evaluation results on the same. Nevertheless, the results show that our approach can prove useful for fingerprint inpainting task as well.

3.4.2 Observed Challenges

Our Finger-UNet model, as evidenced by our experiments, demonstrates promising results in many scenarios. Yet, there are specific instances where it falters, as depicted in Figure 3.7. Notably, when confronted with inputs that exhibit extreme artifacts, the model occasionally misconstrues these as integral parts of the fingerprint, attempting to enhance them instead. Similarly, when presented with inputs that are excessively dark or light, obscuring the ridge structure, the model's predictions fall short in those regions. Another notable challenge arises when the input features a fingernail; the model struggles to

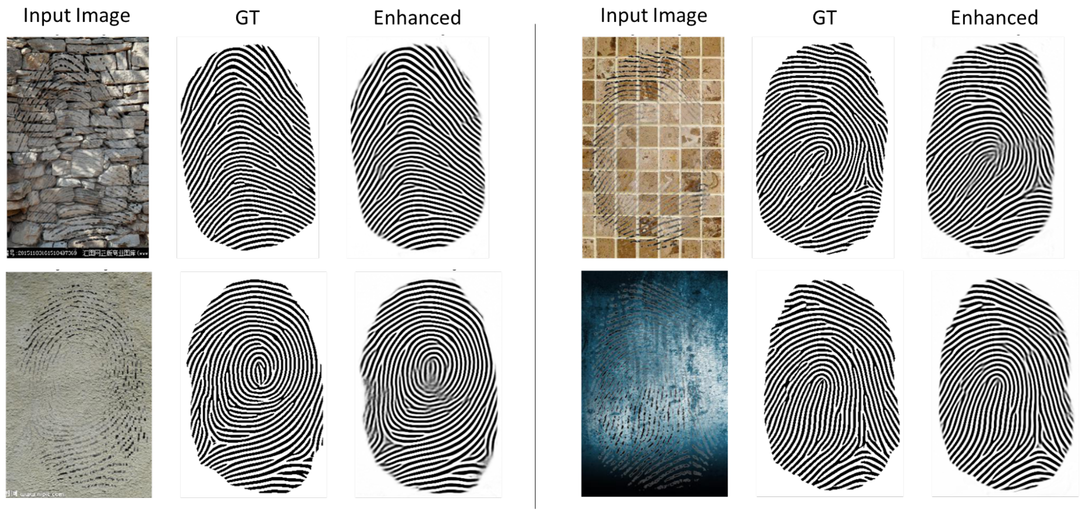

Figure 3.8: Fingerprint enhancement results on Chalearn data

distinguish between the nail and the fingerprint texture. A plausible explanation for these shortcomings could be the nature of our training dataset. The SFinge dataset, to our knowledge, does not present instances of fingerprints with nails or significant artifacts, potentially leaving our model unprepared for such variations.

3.5 Summary

In this chapter, we discussed the fingerprint enhancement approach by making modifications to vanilla U-Net, to improve fingerprint quality. We also incorporated domain knowledge in the learning with minutia and orientation fields to make the enhancement task robust. We evaluated our model on two public fingerprint datasets FVC 2002 and NIST SD302. The network is robust enough to recover fingerprints even with various degrees of degradation. From the experimental results, we say that L_1 loss performed well for this task, along with domain knowledge from minutia and orientation branches, which improved performance above baselines. We also discussed the challenging cases in our experiments and the possible solutions. Moreover, using the wavelet attention block helped improve the performance.

Table 3.1: Ablation study: Evaluation performance on SFinGe test dataset with SSIM, MSE, MAE and PSNR metrics with different modifications to U-Net suggested in this chapter.

Approach	**SSIM**	**RMSE**	**PSNR**	**NFIQ2**
Raw Images	0.605	117.31	6.89	36.42
U-Net (L_2 loss)	0.863	43.67	12.78	47.72
U-Net (L_1 loss)	0.883	41.83	14.12	49.45
U-Net + Depthwise Sep.	0.890	41.25	14.17	49.78
U-Net + WA Block	0.919	40.31	14.43	51.01
U-Net + Minutia	0.934	37.92	17.15	52.86
U-Net + Orientation	0.928	38.58	16.81	51.32
U-Net + Minutia + Orientation	0.943	37.43	17.62	53.11
U-Net + WA Block + Minutia + Orientation	0.954	36.91	18.47	55.32
U-Net + WA Block + Minutia + Orientation + Depthwise Sep.	0.955	36.78	18.81	55.34

Table 3.2: Average NFIQ2 scores of the images from FVC 2002 and NIST SD302 datasets. Higher scores represent higher fingerprint image quality.

Dataset	**FVC 2002**	**NIST SD302**
Raw Images	35.10	46.97
[68]	54.11	56.84
[40]	56.01	58.23
Ours	56.26	58.49

Table 3.3: Average matching scores from BOZORTH3 on different subsets (DB1, DB2, DB3 and DB4) of FVC 2002 dataset. The higher the scores, the better the approach. Feature extraction was performed using MINDTCT.

Database	DB1	DB2	DB3	DB4
Raw Images	52.77	48.62	45.21	50.26
[68]	71.34	71.06	67.08	68.20
[40]	73.52	72.31	69.12	70.29
Ours	74.01	72.61	69.10	71.08

Chapter 4

Self-supervised Fingerprint Representation Learning

4.1 Introduction

Fingerprints, with their distinctive and unique patterns, serve as a vital biometric identifier in a range of applications, from security systems to crime scene analysis. The ability to harness their potential lies in the acquisition of high-quality fingerprint representations. Traditionally, generating these representations relies heavily on supervised learning techniques, which demand extensive, multi-impression fingerprint datasets. The acquisition of such datasets often poses significant financial and logistical challenges, thereby necessitating a more efficient alternative.

Recently, the field of machine learning has witnessed a surge in the popularity of self-supervised learning techniques [142, 143, 144]. This paradigm has proven particularly effective in domains where labeled data is scarce or expensive to obtain. Unlike supervised learning, self-supervised learning techniques do not rely on labeled data. Instead, they learn representations by predicting certain aspects of the input data based on other parts or by solving auxiliary tasks. This ability to learn from unlabeled data makes self-supervised learning a compelling alternative for domains marked by data scarcity or the challenges of data acquisition.

In the context of fingerprint biometrics, self-supervised learning promises a solution to the problems inherent to data acquisition for supervised learning. It offers a pathway to learn meaningful representations from abundant unlabeled fingerprint data, bypassing the need for time-consuming and labor-intensive acquisition and labeling processes. Furthermore, self-supervised learning can potentially capture more intricate, data-specific patterns, leading to richer, more robust fingerprint representations.

The application of self-supervised learning techniques to fingerprint representation learning constitutes a significant research opportunity. This chapter aims to systematically investigate the efficacy of self-supervised learning in fingerprint biometrics. The ultimate goal is to establish a more data-efficient, cost-effective, and robust framework for fingerprint identification systems.

Our journey into self-supervised learning begins with an examination of the techniques that have been successful in the realm of computer vision. With the aim of transferring these techniques to the domain of fingerprint biometrics, we will explore how these self-supervised learning methods can be adapted to function efficiently with fingerprints, a domain with its own unique challenges and intricacies.

Following this exploration, we introduce a novel pre-training technique, developed specifically for Fingerprint Representation Learning. This technique leverages the pre-trained encoder from our U-Net-based fingerprint enhancement model, which we hypothesize may already encode valuable information about the fingerprints. This hypothesis stands at the core of our research in this chapter. This aims to build upon our study in the preceding chapter, where we explored the potential of the U-Net architecture for fingerprint enhancement.

Our objectives in this chapter are threefold. First, we aim to transplant existing self-supervised techniques, established in the computer vision domain, into the realm of fingerprint biometrics. By testing their performance in this novel context, we seek not only to bridge the gap between these two fields but also to establish solid benchmarks for future studies.

Second, we plan to investigate the effectiveness of using the training of a fingerprint enhancement model as a pre-training strategy. Specifically, we are interested in the capacity of the pre-trained encoder from our U-Net based enhancement model to generate useful fingerprint representations. This exploration seeks to evaluate whether the complex task of fingerprint enhancement can provide a rich, initial encoding of fingerprints, which can serve as a strong starting point for further representation learning.

Finally, our third objective is to examine the potential of these pre-trained networks with the probing experiments. Here, we add a few linear layers on top of the encoder, keep the encoder frozen and train the model for the fingerprint verification task using the siamese architecture. This allows the network to adapt the learned representations using only a few linear layers and with the limited amount of data. The trained network then can be used for verification and recognition tasks. This strategy leverages a large amount of unlabeled data to learn initial representations and limited amount of labeled data for supervised adaptation for specific tasks reducing overall cost of fingerprint annotation and training model from scratch.

In accomplishing these objectives, we aim to lay the foundation for an efficient, data-savvy learning framework in the field of fingerprint biometrics. Our focus on self-supervised learning, combined with the efficient use of pre-training and fine-tuning strategies, promises a novel, robust, and economical approach to fingerprint verification and identification, thus contributing to the advancement of the broader field of biometric identification.

4.2 Self-supervised Learning Techniques

Self-supervised learning is a machine learning paradigm that leverages unlabeled data to train models, obviating the need for manually annotated labels. In this approach, the data itself provides supervision, often through the formulation of pretext tasks designed to capture the underlying structure of the data. The potential of self-supervised learning is particularly significant in fields like computer vision and natural language processing, where acquiring large amounts of labeled data can be both costly and time-consuming.

4.2.1 Contrastive Learning

Contrastive learning, based on the principle of learning by comparison, stands as a cornerstone of self-supervised learning. It revolves around the idea of differentiating between similar (positive) and dissimilar (negative) instances [145]. Various techniques that illustrate this concept have achieved significant success. The SimCLR [146] framework, for instance, produces augmented versions of an image, treats them as distinct instances, and then trains the model to identify the original image pair among a set of negative samples. Another example is MoCo (Momentum Contrast) [147] which maintains a dynamic dictionary of data samples in a queue and a moving-averaged encoder to tackle the challenge of large-scale instance discrimination. More recent advancements include methods like BYOL (Bootstrap Your Own Latent) [148] which deviate from the traditional contrastive learning paradigm by not using negative samples and instead focusing on bringing representations of different views of the same image closer in the embedding space. Similarly, SwAV [149] uses a unique approach of swapping cluster assignments to maximize consistency between differently augmented views of the same image. Another notable technique is the Noise Contrastive Estimation (NCE) [150], which contrasts a true data sample against noise samples and has been extensively used in the field of natural language processing for word embedding learning. These techniques provide a comprehensive insight into contrastive learning, a critical approach within self-supervised learning that has potential applications in fingerprint biometrics.

4.2.2 Generative Modeling

Generative models offer a unique approach to learning from unlabeled data by understanding data distribution and generating new data points, with Variational Autoencoders (VAEs) [151] and Generative Adversarial Networks (GANs) [67] representing traditional methodologies. Recent advances have extended these concepts to transformer-based architectures, such as Masked Auto Encoders (MAE) [152] that learn to predict masked image patches, BERT-like pre-training methods applied to image data as in BEiT [153], and frameworks like SimMIM [154] that further explore the potential of masked image modeling. These techniques represent cutting-edge progress in using generative modeling for the self-supervised learning, promising new opportunities in image domains like fingerprint biometrics.

4.2.3 Self-supervision in Natural Language Processing (NLP)

Self-supervised learning has significantly advanced Natural Language Processing (NLP), using the inherent structure of language to learn powerful representations. Techniques such as language modeling and masked language modeling, used in models like GPT [155] and BERT [156], involve predicting words or masked words in a sentence, fostering understanding of word meanings, grammar rules, and sentence structure. Moreover, contrastive learning, often used for image data, has been adapted for NLP, as exemplified by SimCSE [157] - simple contrastive learning of sentence embeddings. SimCSE creates pairs of sentence embeddings from the same sentence with different noise transformations and uses contrastive learning to bring these pairs closer together in the embedding space, thus generating meaningful sentence representations. These self-supervised methods have enriched NLP tasks like text classification and question answering. While these techniques are specifically designed for language, their underlying principles offer valuable insights for applying self-supervised learning in other domains, including fingerprint biometrics.

4.2.4 Other Techniques

Numerous self-supervised learning techniques have been explored recently, underscoring the field's dynamism and potential. Traditional methods include predicting the rotation angle of an image [158], colorization of grayscale images [159], solving jigsaw puzzles [160], and temporal prediction tasks in videos [161, 162]. Newer strategies have emerged such as DINO [163] for Vision Transformers [164], redundancy reduction with Barlow Twins [165], and contrasting cluster assignments with SwAV [149]. Adding to these innovative techniques, recent works have proposed Position Prediction as an effective pretraining strategy [166] and Masked Feature Prediction for visual pre-training [167]. These advancements not only illustrate the versatility of self-supervised learning across diverse domains but also provide valuable insights for its application in fingerprint biometrics.

While the self-supervised learning techniques have been extensively applied in fields like computer vision and NLP, their adaptation for the domain of fingerprint biometrics presents a largely unexplored yet promising avenue. The following sections delve into this unique exploration, aiming to bridge this gap and uncover the potential of self-supervised learning for fingerprint representation.

4.3 Methodology

The methodology for our research is constructed around a two-stage framework to probe the potential of self-supervised learning in fingerprint representation learning. A broad overview of the process is as follows:

- **Stage 1: Self-Supervised Pre-training:** This is the initial stage of our methodology, in which we perform pre-training of our models in a self-supervised manner. It includes the application of both existing self-supervised learning techniques as well as our novel approach for this task. The intent of this stage is to leverage the power of unlabeled data to learn meaningful representations that can serve as a starting point for subsequent stages. Notably, for all methods, we keep the encoder architecture the same. While other self-supervised methods traditionally use encoders

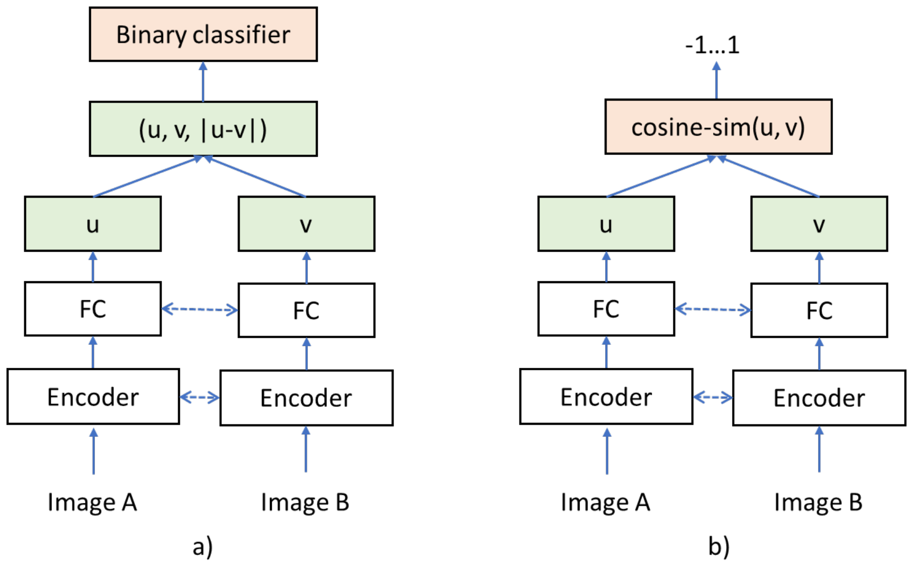

Figure 4.1: a) Architecture with verification objective i.e. with binary classifier (at training and inference) b) Architecture to compute similarity scores (at inference). The dotted arrows indicate networks having tied weights (siamese network structure).

like ResNet or Vision Transformers, in our framework we use the encoder of our U-Net-based enhancement model to ensure a fair comparison.

- **Stage 2: Probing Experiments:** Upon completion of the pre-training phase, we progress to the second stage where a few linear layers are added on top of the frozen pre-trained encoder. We then perform probing experiments using this newly formed model with a limited amount of labeled data. By keeping the encoder part frozen, we ensure that the model adapts the existing representations for the verification task without altering the learned patterns from the self-supervised pre-training phase.

Following this framework, we navigate through the process of adapting and implementing self-supervised learning techniques, exploring a U-Net-based pre-training strategy, and conducting probing experiments with pre-trained networks. The sections below provide a detailed overview of the procedures involved in each stage.

4.3.1 Adaptation and Implementation of Self-Supervised Learning Techniques

Our exploration into the potential of self-supervised learning for fingerprint biometrics begins with an examination of the existing methods. Our selected methods include SimCLR v2 [146], MoCo v2 [147], BYOL (Bootstrap Your Own Latent) [148], SimSiam [168]. These methods, although primarily developed and proven in the domain of computer vision, offer promising prospects for adaptation to fingerprint data. It is required to make necessary modifications to these methods in order to fit the unique characteristics of fingerprints, such as their high dimensionality and complex patterns.

The data augmentation is one of the key elements for the success of the self-supervised methods. SimCLR uses a variety of augmentations to form the positive pairs and it showed that the composition of data augmentations plays a critical role in defining effective predictive tasks [169]. MoCo v2 showed that simple modifications like using the Multi-Layer Perceptron (MLP) projection head and more data augmentation establish stronger baselines that outperform SimCLR and do not require large training batches [147]. They also show that the extra augmentation alone significantly improves the MoCo [170] baseline on ImageNet [147]. Starting from an augmented view of a given image, BYOL trains its online network to predict the target network's representation of another augmented view of the same image [148]. SimSiam shows that a simple siamese network can learn meaningful representations without using negative sample pairs, large batches, and momentum encoders but it still requires the use of extensive data augmentation [168]. We apply a random combination of transformations like rotation, color jitter, resize, crop, and Gaussian blur for the data augmentation. Moreover, the training datasets also contain data acquired with various devices, with varied degrees of distortion and scratches, providing enough data diversity for the specific use case of fingerprint biometrics.

The techniques used in this section attach an additional projection head on top of the encoder during the training. Once the training is completed, we discard the projection head to obtain learned representations. This encoder is used further for probing experiments for downstream tasks. By setting the stage with a comprehensive examination of existing self-supervised techniques, this phase of our research offers the necessary foundation and benchmarks for the introduction and evaluation of our novel self-supervised technique for fingerprint representation learning.

4.3.2 Fingerprint Enhancement as Pre-training Strategy

While the adaptation and application of existing self-supervised methods to fingerprint data provide a valuable starting point, we believe that the uniqueness of fingerprint data could benefit from a self-supervised learning method specifically tailored for it. Building upon our findings in the previous chapter, we propose a novel self-supervised learning technique for fingerprint representation, which leverages the training of a fingerprint enhancement model as a form of self-supervision.

We explore the use of a U-Net-based fingerprint enhancement as a pre-training strategy. We hypothesize that the encoder of our U-Net model, already trained on the task of fingerprint enhancement, might contain valuable fingerprint representations. The process of enhancing a fingerprint image can serve as an effective self-supervised task, encouraging the model to learn useful, fingerprint-specific representations. Specifically, we suggest that the pre-trained encoder from our U-Net-based fingerprint enhancement model already encapsulates valuable information about the fingerprint, which can be used as a foundation for further representation learning.

It's crucial to note that the quality of these initial representations heavily relies on the effectiveness of the U-Net-based enhancement model. Hence, the importance of the enhancement model's design and training, which we discussed extensively in the preceding chapter, cannot be overstated.

4.3.3 Probing Experiments with Pre-trained Networks

After the self-supervised pre-training, we conduct the probing experiments using the pre-trained networks. The aim of these experiments is to assess the usefulness of the learned representations for the task of fingerprint verification. For this, we add a 3-layer MLP on top of the frozen encoder part of the pre-trained network, making the feature representations 512-d. We then train this model using a Sentence-BERT-like [171] siamese architecture, with a small amount of labeled data for the fingerprint verification task. We concatenate the fingerprint representations u and v with the element-wise difference $|u - v|$ and then pass it through the linear layers and train it for binary-classification objective as illustrated in Figure 4.1. By keeping the encoder part frozen, the model learns to adapt the existing representations for the verification task, without changing the underlying learned patterns. Note that the supervised fine-tuning, allowing modifications in the encoder weights can lead to higher performance on the end task. This approach allows us to leverage a large amount of unlabeled data to learn initial representations and a limited amount of labeled data for supervised adaptation. As our goal here is to examine the robustness of the learned representations by different pre-training techniques, we keep the encoder frozen. In summary, the combination of self-supervised pre-training with supervised fine-tuning offers a promising learning framework for fingerprint biometrics. Our methodology aims to leverage the strengths of both self-supervised and supervised learning, offering a pathway towards robust, efficient, and data-efficient fingerprint biometric systems.

4.4 Experiments

In this section, we discuss the experiments performed to evaluate our proposed framework's efficacy. We cover the specifics of our experimental setup, including the datasets used, the training details, and the evaluation metrics employed.

4.4.1 Datasets and Preprocessing

The datasets used in this study consist of both synthetic and real-world fingerprint images, originating from the Synthetic Fingerprint Generator (SFinGe) tool, the Fingerprint Verification Competition (FVC) database, and the NIST SD-302 database.

The synthetic SFinGe dataset was crafted to reflect real-world challenges in fingerprint recognition, such as the presence of various kinds of distortions like backgrounds, rotation/translation, and the presence of scratches and other noise. The real-world FVC datasets and NIST SD-302 database provide large-scale, realistic fingerprint data, ensuring the generalizability and robustness of the findings. This

Dataset	Source	Identities	Images	Purpose
SFinGe	Synthetic	3,700	15000	Training
FVC-2000	Real-world	440	3520	Training
FVC-2002	Real-world	440	3520	Training
NIST SD-302	Real-world	2000	8000	Training
SFinGe	Synthetic	1584	6336	Testing
FVC-2004	Real-world	440	3520	Testing

Table 4.1: Summary of Datasets

combination of synthetic and real-world datasets allows the training and evaluation of the model under a diverse set of conditions. The synthetic data offers scalability and control over experimental parameters, while the real-world data ensures applicability in realistic conditions. Table 4.1 provides a summary statistics of the datasets.

For the self-supervised pre-training phase, we used only the fingerprint images from the training datasets, discarding any associated ground truth labels or identities, in accordance with the principles of self-supervised learning. In the probing experiments, however, the training process shifted to a binary classification task for fingerprint verification. Positive and negative fingerprint pairs are generated taking into account their associated identities. The model was trained to classify if a given pair of fingerprints belong to the same identity.

4.4.2 Implementation Details

We conduct all our experiments using the PyTorch [172] deep learning framework with NVIDIA GeForce TITAN X GPUs for training.

Our proposed enhancement-based pre-training technique utilizes the U-Net architecture, and its encoder part is used as a feature extractor to generate fingerprint representations. The same U-Net encoder architecture is used for pre-training with other self-supervised learning methods for a fair comparison with our technique. During self-supervised pre-training, all the models are trained from scratch. Our U-Net architecture has a depth of 5 layers, wherein each layer consists of 2 convolutions. It expects gray-scale fingerprint images with dimensions of 512 x 512 pixels as encoder input and decoder output. Therefore, all input images were resized and padded as needed to match this input size. The encoder produces a 4096-dimensional vector bottleneck which is used as fingerprint representation. The model uses depth-wise convolutions to reduce the overall number of parameters.

For the pre-training with the existing self-supervised methods, we performed a grid search to identify the best hyperparameters, such as learning rate, batch size, learning rate schedule, and momentum value, and pre-train each model for 50 epochs with early stopping applied to prevent overfitting. The enhancement model is trained as described in the previous chapter to achieve the best results. Note that we ignore the minutia and orientation estimation branches from the enhancement approach from the previous chapter in order to compare the results only for the enhancement branch without supervision from the minutia and orientation branches.

For probing experiments, we add MLP layers on top of the trained encoders and adapt the weights for the verification task while keeping the encoder weights fixed. We generate training and testing verification sets from FVC and SFinge datasets in a 1:3 ratio of positive and negative pairs. We used SD302 only in training and not in evaluation due to an insufficient number of impressions in the dataset. After training, we evaluate the models on the test sets for the verification task. We report verification accuracy, precision, recall, and F1 scores as metrics to evaluate different techniques. We report the results in two ways: 1) using the entire model as a binary classifier and 2) only utilizing the embeddings and applying thresholds on the cosine similarity between pairs (Refer Equation 4.1). The first way simply evaluates the trained model as an end-to-end verification network. The second approach shows us the potential to use the learned representations in the context of similarity search and in turn for recognition tasks.

$$\text{cosine similarity } (x_1, x_2) = \frac{x_1 \cdot x_2}{\max(||x_1||_2 \cdot ||x_2||_2, \epsilon)} \tag{4.1}$$

where x_1 and x_2 are representation vectors and ϵ is very small value $1e^{-8}$ to avoid division by zero.

4.5 Results and Discussion

The model is first pre-trained to learn fingerprint representations using various self-supervised learning strategies. Because these representations are not explicitly trained for fingerprint verification or identification, using them directly for evaluation is inappropriate. To gauge the stability and usefulness of these learned representations, we add linear layers to the frozen pre-trained encoders and then train the models for fingerprint verification tasks. The encoders remain frozen, allowing only the weights of the MLP to adjust to the task, keeping the original representations unchanged. This setup aids in comparing the efficacy of different self-supervised learning techniques against our method. The accuracy, F1 score, precision, and recall metrics of our probing experiments are presented in the following tables on the SFinGe dataset and FVC datasets.

Our approach is compared with methods like SimCLR v2, SimSiam, MoCo v2, and BYOL on the SFinGe and FVC test sets for fingerprint verification. As the ratio of positive to negative pairs in the test data for fingerprint verification is 1:3, the random guess accuracy becomes 75%. The whole model, a pre-trained encoder with the attached MLP, is treated as a binary classifier, and the resulting performance is reported. This is presented under the 'Classification' column in the following tables. In the second method, the embeddings of the input fingerprints are extracted from the initial MLP layer, their cosine similarity is calculated, and a threshold is applied to determine if the fingerprints are from the same identity. The optimal threshold value providing the best performance numbers is chosen for all methods.

Table 4.2: Verification accuracy on SFinGe test dataset with genuine and imposter pairs

SFinGe - Accuracy						
Method	**Classification**			**Similarity**		
	Imposter	**Genuine**	**Entire Data**	**Imposter**	**Genuine**	**Entire Data**
SimCLR	0.968	0.881	0.96	0.982	0.749	0.961
SimSiam	0.972	0.362	0.916	0.888	0.648	0.866
MoCo	0.963	0.881	0.956	0.955	0.845	0.945
BYOL	0.96	0.825	0.947	0.963	0.718	0.941
Ours	**0.982**	**0.886**	**0.973**	**0.975**	**0.847**	**0.963**

Table 4.3: Precision on SFinGe test dataset with genuine and imposter pairs

SFinGe - Precision						
Method	**Classification**			**Similarity**		
	Imposter	**Genuine**	**Entire Data**	**Imposter**	**Genuine**	**Entire Data**
SimCLR	0.99	0.74	0.737	0.98	0.82	0.815
SimSiam	0.94	0.57	0.567	0.96	0.37	0.367
MoCo	0.99	0.71	0.708	0.98	0.65	0.654
BYOL	0.98	0.67	0.674	0.97	0.66	0.662
Ours	**0.99**	**0.83**	**0.832**	**0.98**	**0.78**	**0.776**

This is represented under the ‘Similarity’ column in the tables. Additionally, we report ROC curves on both datasets based on the similarity scores.

As demonstrated by the results, our pre-training method consistently outperforms other self-supervised pre-training strategies across both test datasets. SimCLR v2 also consistently performs well. SimSiam and BYOL methods show comparatively poorer performance. It is noteworthy that all models perform better on the SFinGe test set than the FVC test set. We believe this is due to two primary factors: the training sets contain more data from SFinGe than FVC, potentially resulting in a bias towards the former, and SFinGe is a synthetic dataset while FVC consists of real fingerprints, making the latter more challenging. Importantly, our method also provides superior performance when verification is based on the similarity of the embeddings, suggesting that the learned representations are also useful for fingerprint recognition.

4.5.1 Limitations and Future Directions

Despite our promising findings, there exist certain limitations that need to be acknowledged. Primarily, the current model is more effective on the SFinGe dataset, which is synthetic, than on the FVC dataset that encompasses real-world fingerprints. This discrepancy might be due to the underrepresentation of FVC data in the training sets, leading to potential bias, and also due to inherent complexities

Table 4.4: Recall on SFinGe test dataset with genuine and imposter pairs

SFinGe - Recall						
Method	**Classification**			**Similarity**		
	Imposter	**Genuine**	**Entire Data**	**Imposter**	**Genuine**	**Entire Data**
SimCLR	0.97	0.88	0.881	0.98	0.75	0.749
SimSiam	0.97	0.36	0.362	0.89	0.65	0.648
MoCo	0.96	0.88	0.881	0.96	0.85	0.845
BYOL	0.96	0.83	0.825	0.96	0.72	0.718
Ours	**0.98**	**0.89**	**0.886**	**0.98**	**0.85**	**0.847**

Table 4.5: F1 score on SFinGe test dataset with genuine and imposter pairs

SFinGe - F1 score						
Method	**Classification**			**Similarity**		
	Imposter	**Genuine**	**Entire Data**	**Imposter**	**Genuine**	**Entire Data**
SimCLR	0.98	0.8	0.803	0.98	0.78	0.781
SimSiam	0.96	0.44	0.442	0.92	0.47	0.469
MoCo	0.98	0.79	0.785	0.97	0.74	0.737
BYOL	0.97	0.74	0.742	0.97	0.69	0.689
Ours	**0.99**	**0.86**	**0.858**	**0.98**	**0.81**	**0.821**

Table 4.6: Verification accuracy on FVC test dataset with genuine and imposter pairs

FVC - Accuracy						
Method	**Classification**			**Similarity**		
	Imposter	**Genuine**	**Entire Data**	**Imposter**	**Genuine**	**Entire Data**
SimCLR	0.915	0.619	0.888	0.943	0.537	0.906
SimSiam	0.956	0.122	0.88	0.387	0.733	0.419
MoCo	0.902	0.522	0.867	0.896	0.56	0.865
BYOL	0.886	0.568	0.857	0.926	0.477	0.886
Ours	**0.957**	**0.73**	**0.937**	**0.933**	**0.818**	**0.923**

Table 4.7: Precision on FVC test dataset with genuine and imposter pairs

FVC - Precision						
Method	**Classification**			**Similarity**		
	Imposter	**Genuine**	**Entire Data**	**Imposter**	**Genuine**	**Entire Data**
SimCLR	0.96	0.42	0.422	0.95	0.49	0.486
SimSiam	0.92	0.22	0.218	0.94	0.11	0.107
MoCo	0.95	0.35	0.347	0.95	0.35	0.35
BYOL	0.95	0.33	0.334	0.93	0.48	0.394
Ours	**0.97**	**0.63**	**0.634**	**0.98**	**0.55**	**0.553**

Table 4.8: Recall on FVC test dataset with genuine and imposter pairs

FVC - Recall						
Method	**Classification**			**Similarity**		
	Imposter	**Genuine**	**Entire Data**	**Imposter**	**Genuine**	**Entire Data**
SimCLR	0.92	0.62	0.619	0.94	0.54	0.537
SimSiam	0.96	0.12	0.122	0.39	0.73	0.733
MoCo	0.9	0.52	0.522	0.9	0.56	0.56
BYOL	0.89	0.57	0.568	0.93	0.48	0.477
Ours	**0.96**	**0.73**	**0.73**	**0.93**	**0.82**	**0.818**

Table 4.9: F1 score on FVC test dataset with genuine and imposter pairs

FVC - F1 score						
Method	**Classification**			**Similarity**		
	Imposter	**Genuine**	**Entire Data**	**Imposter**	**Genuine**	**Entire Data**
SimCLR	0.94	0.5	0.502	0.95	0.51	0.51
SimSiam	0.94	0.16	0.156	0.55	0.19	0.186
MoCo	0.93	0.42	0.417	0.92	0.43	0.431
BYOL	0.92	0.42	0.421	0.94	0.43	0.432
Ours	**0.97**	**0.68**	**0.679**	**0.96**	**0.66**	**0.659**

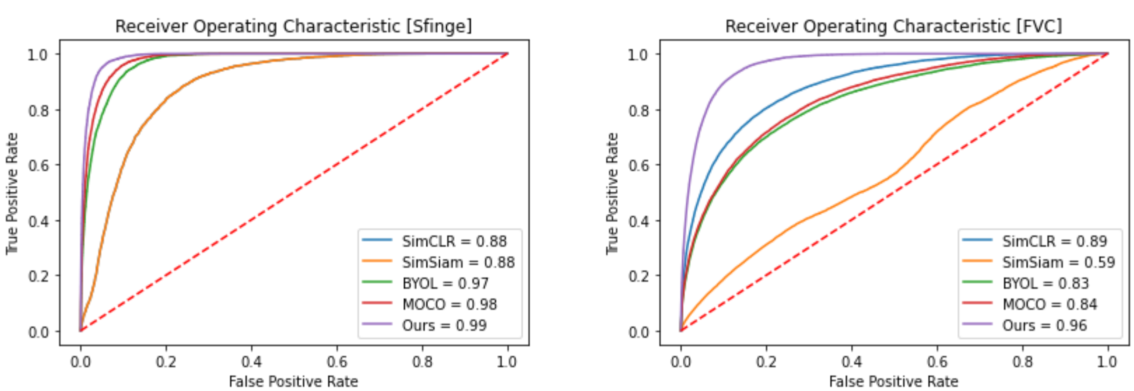

Figure 4.2: ROC curve based on similarity scores on SFinGe dataset(left) and FVC dataset (right)

in real-life fingerprint data, which are harder to handle. Another limitation is that our study did not specifically train or evaluate for the fingerprint recognition task. While our model has shown potential for this task, a specific and thorough evaluation is needed to fully understand its performance in this regard. Furthermore, the efficacy of self-supervised learning methodologies is inherently reliant on the quality and diversity of the training data. As the current training data does not encompass all the complexities found in real-world applications, there might be limitations in generalizing the findings beyond the datasets used. In addition, we have utilized a specific form of linear probing for our study. There exist alternative approaches, such as softmax or ArcFace based classification, which may help in learning better representations.

In terms of future work, it would be beneficial to address these limitations by including a broader, more diverse range of real-world fingerprint datasets in the training phase. This would enhance the model's generalizability and make it more robust against a variety of fingerprint data. Specific training and evaluation for the recognition task, as well as exploring alternative linear probing techniques, would also be valuable directions to pursue. Additionally, other forms of self-supervised learning methods could be explored to continually optimize the model performance. Exploring ways to refine the model to function more effectively with challenging real-world fingerprint data would also be a beneficial direction for future research.

4.6 Summary

In this chapter, we explored the application of various self-supervised learning techniques for pre-training a model to learn good fingerprint representations which can be useful to recognize and verify fingerprints. We proposed a novel approach to use fingerprint enhancement as a self-supervised pre-training method. We performed probing experiments that proved beneficial in evaluating the effectiveness of learned fingerprint representations across different pre-training strategies. The verification performance of our method was compared against SimCLR v2, SimSiam, MoCo v2, and BYOL methods using two test sets, SFinGe and FVC. Our method consistently outperforms other techniques across both test datasets, thereby demonstrating the robustness and effectiveness of our model. Our method also surpasses other methods when evaluated in terms of similarity-based verification accuracy indicating the effectiveness of the learned representations for fingerprint recognition task. However, it was observed that all models performed better on the synthetic SFinGe dataset compared to the real-world FVC dataset, indicating potential limitations related to bias in the training set and complexities of real-world fingerprint data. In the future, we intend to extend our research to encompass more diverse and complex real-world fingerprint datasets, thus enhancing the generalizability of our model. We also plan to investigate other self-supervised learning methods and strategies to enhance the model's performance and adaptability to real-world fingerprint data complexities. In conclusion, this chapter illuminates the potential of self-supervised learning methods in the domain of fingerprint biometrics, while highlighting areas for further exploration and improvement.

Chapter 5

Conclusions and Future Work

This thesis aimed to address two critical aspects of fingerprint recognition: enhancement and verification. Our primary objective was to improve fingerprint image quality using a U-Net-based model and to incorporate domain-specific knowledge by adding minutiae and orientation branches. The second goal was to employ the pre-trained encoder from our enhancement model for fingerprint verification in a self-supervised manner.

Through rigorous experimentation, we demonstrated that the incorporation of domain knowledge into a U-Net variant improved fingerprint image quality, as validated by various metrics such as SSIM, PSNR, and RMSE. Additionally, our self-supervised approach for fingerprint verification yielded substantial improvements in recognition performance when compared to traditional methods.

Our research has several implications for the field of biometric recognition. First, it provides a way to include domain-specific knowledge in deep learning models for fingerprint enhancement with a simpler architecture like U-Net. Second, it demonstrates the utility of enhancement-pretrained models in self-supervised tasks, specifically for fingerprint verification.

While our research offers promising results, there are limitations that must be acknowledged. The most notable being that our work majorly relies on synthetically generated data, which may not capture all the nuances and challenges of real-world fingerprint images. This is due to the unavailability of suitable fingerprint datasets in the public domain.

Further research can be directed towards expanding the model to adapt to real-world scenarios better and handle additional forms of degradation like latent fingerprints and text on fingerprint degradations. Testing the model on larger and more diverse datasets may also yield more comprehensive insights into its efficacy.

This thesis not only contributes to the current body of knowledge in the realm of fingerprint recognition but also opens avenues for cost-effective, efficient, and robust biometric systems. It is our hope that our findings serve as a foundation for future studies aiming to push the boundaries of what is achievable in fingerprint biometrics.

Related Publications

- **Ekta Gavas** and Anoop Namboodiri, “Finger-UNet: A U-Net Based Multi-Task Architecture for Deep Fingerprint Enhancement”, Proceedings of the 18th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - VISAPP, February 2023

- **Ekta Gavas**, Kaustubh Olpadkar, Anoop Namboodiri, “Enhancement-Driven Pretraining for Robust Fingerprint Representation Learning”, Proceedings of the 19th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - VISAPP, February 2024

Bibliography

[1] Anil K Jain and Jianjiang Feng. Latent fingerprint matching. *IEEE Transactions on pattern analysis and machine intelligence*, 33(1):88–100, 2010.

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

[3] Xiangyu Zhao, Peng Huang, and Xiangbo Shu. Wavelet-attention cnn for image classification. *Multimedia Systems*, 28(3):915–924, 2022.

[4] James Wayman, Anil Jain, Davide Maltoni, and Dario Maio. *An Introduction to Biometric Authentication Systems*, pages 1–20. Springer London, London, 2005.

[5] Erik Bowman. Everything you need to know about biometrics. *Identix Corporation*, 2000.

[6] Anil Jain, Ruud Bolle, and Sharath Pankanti. *Biometrics: personal identification in networked society*, volume 479. Springer Science & Business Media, 1999.

[7] Robert Allen, Pat Sankar, and Salil Prabhakar. *Fingerprint Identification Technology*, pages 22–61. Springer London, London, 2005.

[8] Adhiyaman Manickam, Rishin Haldar, Muhammad Saqlain, and Veerappan Sellam. Fingerprint image classification using local diagonal and directional extrema patterns. *Journal of Electronic Imaging*, 28:1, 06 2019.

[9] Nurettin Topaloglu. Revised: Fingerprint classification based on gray-level fuzzy clustering co-occurrence matrix. *Energy Education Science and Technology Part A: Energy Science and Research*, 31:1307–1316, 01 2013.

[10] Wencheng Yang, Song Wang, Jiankun Hu, Guanglou Zheng, and Craig Valli. Security and accuracy of fingerprint-based biometrics: A review. *Symmetry*, 11(2):141, 2019.

[11] Brian E. Dalrymple. *Fingerprints*, pages 117–141. Humana Press, Totowa, NJ, 2006.

[12] U.S.D. Justice. *The Fingerprint: Sourcebook*. CreateSpace Independent Publishing Platform, 2014.

[13] William J Herschel. Skin furrows of the hand. *Nature*, 23(578), 1880.

[14] Davide Maltoni, Dario Maio, Anil K. Jain, and Jianjiang Feng. *Fingerprint Sensing*, pages 63–114. Springer International Publishing, Cham, 2022.

[15] John Berry and David A Stoney. The history and development of fingerprinting. *Advances in fingerprint Technology*, 2:13–52, 2001.

[16] Francis Galton. Personal identification and description. *Journal of anthropological institute of Great Britain and Ireland*, pages 177–191, 1889.

[17] Edward Richard Henry. *Classification and uses of finger prints*. HM Stationery office, 1922.

[18] Xiongwu Xia and Lawrence O'Gorman. Innovations in fingerprint capture devices. *Pattern Recognition*, 36(2):361–369, 2003. Biometrics.

[19] Ichiro Fujieda, Yuzo Ono, and Seijin Sugama. Fingerprint image input device having an image sensor with openings, August 29 1995. US Patent 5,446,290.

[20] Robert F Dowling Jr and Keith L Knowlton. Fingerprint acquisition system with a fiber optic block, November 15 1988. US Patent 4,785,171.

[21] Constantine Tsikos. Capacitive fingerprint sensor, October 5 1982. US Patent 4,353,056.

[22] Jeong-Woo Lee, Dong-Jin Min, Jiyoun Kim, and Wonchan Kim. A 600-dpi capacitive fingerprint sensor chip and image-synthesis technique. *IEEE Journal of Solid-State Circuits*, 34(4):469–475, 1999.

[23] Alexander G Dickinson, Ross McPherson, Sunetra Mendis, and Paul C Ross. Capacitive fingerprint sensor with adjustable gain, April 11 2000. US Patent 6,049,620.

[24] Marco Tartagni and Roberto Guerrieri. A fingerprint sensor based on the feedback capacitive sensing scheme. *IEEE Journal of Solid-State Circuits*, 33(1):133–142, 1998.

[25] David G Edwards. Fingerprint sensor, January 31 1984. US Patent 4,429,413.

[26] Hiro Han and Yasuhiro Koshimoto. Characteristics of thermal-type fingerprint sensor. In *Biometric Technology for Human Identification V*, volume 6944, pages 214–225. SPIE, 2008.

[27] JF Mainguet, M Pegulu, and JB Harris. Fingerchip: thermal imaging and finger sweeping in a silicon fingerprint sensor. In *Proceedings of AutoID*, volume 99, pages 91–94, 1999.

[28] Dale R Setlak. Electric field fingerprint sensor apparatus and related methods, October 5 1999. US Patent 5,963,679.

[29] Chang-Jung Kim, Young-soo Park, and June-key Lee. Fingerprint sensor using a piezoelectric membrane, March 14 2006. US Patent 7,013,031.

[30] John K Schneider and Darold C Wobschall. Live scan fingerprint imagery using high resolution c-scan ultrasonography. In *Proceedings. 25th Annual 1991 IEEE International Carnahan Conference on Security Technology*, pages 88–95. IEEE, 1991.

[31] Wieslaw Bicz, Dariusz Banasiak, Pawel Bruciak, Zbigniew Gumienny, Stanislaw Gumuliński, Dariusz Kosz, Agnieszka Krysiak, Wladysław Kuczyński, Mieczystaw Pluta, and Grzegorz Rabiej. Fingerprint structure imaging based on an ultrasound camera. *Instrumentation science & technology*, 27(4):295–303, 1999.

[32] Hao-Yen Tang, Yipeng Lu, Xiaoyue Jiang, Eldwin J Ng, Julius M Tsai, David A Horsley, and Bernhard E Boser. 3-d ultrasonic fingerprint sensor-on-a-chip. *IEEE Journal of Solid-State Circuits*, 51(11):2522–2533, 2016.

[33] Robert K Rowe and Kristin A Nixon. Fingerprint enhancement using a multispectral sensor. In *Biometric Technology for Human Identification II*, volume 5779, pages 81–93. SPIE, 2005.

[34] Robert K Rowe, Stephen P Corcoran, Kristin A Nixon, and Robert E Ostrom. Multispectral imaging for biometrics. In *Spectral Imaging: Instrumentation, Applications, and Analysis III*, volume 5694, pages 90–99. SPIE, 2005.

[35] Robert K Rowe, Kristin Adair Nixon, and Paul W Butler. Multispectral fingerprint image acquisition. In *Advances in biometrics: Sensors, algorithms and systems*, pages 3–23. Springer, 2008.

[36] Ajay Kumar and Cyril Kwong. Towards contactless, low-cost and accurate 3d fingerprint identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3438–3443, 2013.

[37] John Libert, John Grantham, Bruce Bandini, Stephen Wood, Michael Garris, Kenneth Ko, Fred Byers, and Craig Watson. Guidance for evaluating contactless fingerprint acquisition devices. *NIST Spec. Publ*, 500:305, 2018.

[38] Hyun-Joon Kim-Lee, Seog Woo Hong, Dong Kyun Kim, Jinmyoung Kim, Hong Suk Kim, Seok-Whan Chung, Eun-Hyoung Cho, Hae-Sung Kim, and Byung-Kyu Lee. On-screen fingerprint sensor with optically and electrically tailored transparent electrode patterns for use on high-resolution mobile displays. *Microsystems & Nanoengineering*, 6(1):98, 2020.

[39] Hyunggun Ma, Zhenyi Liu, Sanghyun Heo, Jeeyoon Lee, Kyungmin Na, Han Byul Jin, Sungchul Jung, Kibog Park, Jae Joon Kim, and Franklin Bien. On-display transparent half-diamond pattern capacitive fingerprint sensor compatible with amoled display. *IEEE Sensors Journal*, 16(22):8124–8131, 2016.

[40] Lin Hong, Yifei Wan, and Anil Jain. Fingerprint image enhancement: Algorithm and performance evaluation. *IEEE transactions on pattern analysis and machine intelligence*, 20(8):777–789, 1998.

[41] Asker M Bazen and Sabih H Gerez. Fingerprint matching by thin-plate spline modelling of elastic deformations. *Pattern recognition*, 36(8):1859–1867, 2003.

[42] Xijie Huang, Peng Qian, and Manhua Liu. Latent fingerprint image enhancement based on progressive generative adversarial network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 800–801, 2020.

[43] Tariq Mahmood Khan, Donald G Bailey, Mohammad AU Khan, and Yinan Kong. Efficient hardware implementation for fingerprint image enhancement using anisotropic gaussian filter. *IEEE Transactions on Image processing*, 26(5):2116–2126, 2017.

[44] Somsak Choomchuay and Keokanlaya Sihalath. An application of second derivative of gaussian filters in fingerprint image enhancement. In *2010 4th International Conference on Bioinformatics and Biomedical Engineering*, pages 1–4. IEEE, 2010.

[45] Lawrence O'Gorman. Binarization and multithresholding of document images using connectivity. *CVGIP: Graphical Models and Image Processing*, 56(6):494–506, 1994.

[46] Sharat Chikkerur, Alexander N Cartwright, and Venu Govindaraju. Fingerprint enhancement using stft analysis. *Pattern recognition*, 40(1):198–211, 2007.

[47] Babu M Mehtre, Narayanabhatla Narasimha Murthy, Surendra Kapoor, and B Chatterjee. Segmentation of fingerprint images using the directional image. *Pattern recognition*, 20(4):429–435, 1987.

[48] Barry G Sherlock, DM Monro, and K Millard. Fingerprint enhancement by directional fourier filtering. *IEE Proceedings-Vision, Image and Signal Processing*, 141(2):87–94, 1994.

[49] Pietro Perona and Jitendra Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on pattern analysis and machine intelligence*, 12(7):629–639, 1990.

[50] Michael Kass and Andrew Witkin. Analyzing oriented patterns. *Computer vision, graphics, and image processing*, 37(3):362–385, 1987.

[51] Shlomo Greenberg, Mayer Aladjem, and Daniel Kogan. Fingerprint image enhancement using filtering techniques. *Real-Time Imaging*, 8(3):227–236, 2002.

[52] Andrés Almansa and Tony Lindeberg. Fingerprint enhancement by shape adaptation of scale-space operators with automatic scale selection. *IEEE Transactions on Image Processing*, 9(12):2027–2042, 2000.

[53] Joachim Weickert. Applications of nonlinear diffusion in image processing and computer vision. *None*, 2000.

[54] Guang-Zhong Yang, Peter Burger, David N Firmin, and SR Underwood. Structure adaptive anisotropic image filtering. *Image and Vision Computing*, 14(2):135–145, 1996.

[55] Stephen M Pizer, E Philip Amburn, John D Austin, Robert Cromartie, Ari Geselowitz, Trey Greer, Bart ter Haar Romeny, John B Zimmerman, and Karel Zuiderveld. Adaptive histogram equalization and its variations. *Computer vision, graphics, and image processing*, 39(3):355–368, 1987.

[56] Ali M Reza. Realization of the contrast limited adaptive histogram equalization (clahe) for real-time image enhancement. *Journal of VLSI signal processing systems for signal, image and video technology*, 38:35–44, 2004.

[57] Dario Maio, Davide Maltoni, Raffaele Cappelli, James L. Wayman, and Anil K. Jain. Fvc2000: Fingerprint verification competition. *IEEE transactions on pattern analysis and machine intelligence*, 24(3):402–412, 2002.

[58] Ramana L Rao. *Multiresolution techniques in image processing*. Louisiana State University and Agricultural & Mechanical College, 1995.

[59] Nick Kingsbury and Julian Magarey. Wavelet transforms in image processing, 1998.

[60] Woo Kyu Lee and Jae Ho Chung. Automatic real-time identification of fingerprint images using wavelet transform and gradient of gaussian. In *Proceedings of APCCAS'96-Asia Pacific Conference on Circuits and Systems*, pages 508–511. IEEE, 1996.

[61] Ching-Tang Hsieh, Eugene Lai, and You-Chuang Wang. An effective algorithm for fingerprint image enhancement based on wavelet transform. *Pattern Recognition*, 36(2):303–312, 2003.

[62] Jiangang Cheng and Jie Tian. Fingerprint enhancement with dyadic scale-space. *Pattern Recognition Letters*, 25(11):1273–1284, 2004.

[63] Hartwig Fronthaler, Klaus Kollreider, and Josef Bigun. Pyramid-based image enhancement of fingerprints. In *2007 IEEE Workshop on Automatic Identification Advanced Technologies*, pages 45–50. IEEE, 2007.

[64] Kai Cao and Anil K Jain. Latent orientation field estimation via convolutional neural network. In *2015 International Conference on Biometrics (ICB)*, pages 349–356. IEEE, 2015.

[65] Jian Li, Jianjiang Feng, and C-C Jay Kuo. Deep convolutional neural network for latent fingerprint enhancement. *Signal Processing: Image Communication*, 2018.

[66] Peng Qian, Aojie Li, and Manhua Liu. Latent fingerprint enhancement based on denseunet. In *2019 International Conference on Biometrics (ICB)*, pages 1–6. IEEE, 2019.

[67] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[68] Indu Joshi, Adithya Anand, Mayank Vatsa, Richa Singh, Sumantra Dutta Roy, and Prem Kalra. Latent fingerprint enhancement using generative adversarial networks. In *2019 IEEE winter conference on applications of computer vision (WACV)*, pages 895–903. IEEE, 2019.

[69] Sukesh Adiga V and Jayanthi Sivaswamy. Fpd-m-net: Fingerprint image denoising and inpainting using m-net based convolutional neural networks. *CoRR*, abs/1812.10191, 2018.

[70] Kalle Karu and Anil K Jain. Fingerprint classification. *Pattern recognition*, 29(3):389–404, 1996.

[71] Anil Jain, Lin Hong, and S. Pankanti. Biometric identification. *Commun. ACM*, 43:90–98, 02 2000.

[72] Anil K Jain, Arun Ross, and Salil Prabhakar. An introduction to biometric recognition. *IEEE Transactions on circuits and systems for video technology*, 14(1):4–20, 2004.

[73] Dario Maio and Davide Maltoni. Direct gray-scale minutiae detection in fingerprints. *IEEE transactions on pattern analysis and machine intelligence*, 19(1):27–40, 1997.

[74] Raffaele Cappelli, Dario Maio, Alessandra Lumini, and Davide Maltoni. Fingerprint image reconstruction from standard templates. *IEEE transactions on pattern analysis and machine intelligence*, 29(9):1489–1503, 2007.

[75] Xudong Jiang and Wei-Yun Yau. Fingerprint minutiae matching based on the local and global structures. In *Proceedings 15th international conference on pattern recognition. ICPR-2000*, volume 2, pages 1038–1041. IEEE, 2000.

[76] Tsai-Yang Jea and Venu Govindaraju. A minutia-based partial fingerprint recognition system. *Pattern recognition*, 38(10):1672–1684, 2005.

[77] Soweon Yoon, Jianjiang Feng, and Anil K Jain. Altered fingerprints: Analysis and detection. *IEEE transactions on pattern analysis and machine intelligence*, 34(3):451–464, 2012.

[78] Dinh-Luan Nguyen, Kai Cao, and Anil K Jain. Robust minutiae extractor: Integrating deep networks and fingerprint domain knowledge. In *2018 International Conference on Biometrics (ICB)*, pages 9–16. IEEE, 2018.

[79] Uttam U Deshpande, VS Malemath, Shivanand M Patil, and Sushma V Chaugule. Cnnai: a convolution neural network-based latent fingerprint matching using the combination of nearest neighbor arrangement indexing. *Frontiers in Robotics and AI*, 7:113, 2020.

[80] Luke Nicholas Darlow and Benjamin Rosman. Fingerprint minutiae extraction using deep learning. In *2017 IEEE International Joint Conference on Biometrics (IJCB)*. IEEE, 2017.

[81] Yao Tang, Fei Gao, Jufu Feng, and Yuhang Liu. Fingernet: An unified deep network for fingerprint minutiae extraction. In *2017 IEEE International Joint Conference on Biometrics (IJCB)*, pages 108–116. IEEE, 2017.

[82] Nalini K Ratha, Kalle Karu, Shaoyun Chen, and Anil K Jain. A real-time matching system for large fingerprint databases. *IEEE transactions on pattern analysis and machine intelligence*, 18(8):799–813, 1996.

[83] Shih-Hsu Chang, Fang-Hsuan Cheng, Wen-Hsing Hsu, and Guo-Zua Wu. Fast algorithm for point pattern matching: invariant to translations, rotations and scale changes. *Pattern recognition*, 30(2):311–320, 1997.

[84] Raffaele Cappelli, Matteo Ferrara, Davide Maltoni, and Massimo Tistarelli. Mcc: A baseline algorithm for fingerprint verification in fvc-ongoing. In *2010 11th International Conference on Control Automation Robotics & Vision*, pages 19–23. IEEE, 2010.

[85] Raffaele Cappelli, Matteo Ferrara, and Davide Maltoni. Minutia cylinder-code: A new representation and matching technique for fingerprint recognition. *IEEE transactions on pattern analysis and machine intelligence*, 32(12):2128–2141, 2010.

[86] Anil Jain, Arun Ross, and Salil Prabhakar. Fingerprint matching using minutiae and texture features. In *Proceedings 2001 International Conference on Image Processing (Cat. No. 01CH37205)*, volume 3, pages 282–285. IEEE, 2001.

[87] Anil Jain, Lin Hong, and Ruud Bolle. On-line fingerprint verification. *IEEE transactions on pattern analysis and machine intelligence*, 19(4):302–314, 1997.

[88] Asker M Bazen, Gerben TB Verwaaijen, Sabih H Gerez, Leo PJ Veelenturf, and Berend Jan Van Der Zwaag. A correlation-based fingerprint verification system. In *Proceedings of the ProRISC2000 workshop on circuits, systems and signal processing*, pages 205–213. Citeseer, 2000.

[89] Koichi Ito, Ayumi Morita, Takafumi Aoki, Tatsuo Higuchi, Hiroshi Nakajima, and Koji Kobayashi. A fingerprint recognition algorithm using phase-based image matching for low-quality fingerprints. In *IEEE International Conference on Image Processing 2005*, volume 2, pages II–33. IEEE, 2005.

[90] Koichi Ito, Ayumi Morita, Takafumi Aoki, Hiroshi Nakajima, Koji Kobayashi, and Tatsuo Higuchi. A fingerprint recognition algorithm combining phase-based image matching and feature-based matching. In *Advances in Biometrics: International Conference, ICB 2006, Hong Kong, China, January 5-7, 2006. Proceedings*, pages 316–325. Springer, 2005.

[91] Anil K Jain, Salil Prabhakar, Lin Hong, and Sharath Pankanti. Filterbank-based fingerprint matching. *IEEE transactions on Image Processing*, 9(5):846–859, 2000.

[92] Arun Ross, Anil Jain, and James Reisman. A hybrid fingerprint matcher. *Pattern Recognition*, 36(7):1661–1673, 2003.

[93] Arun Ross, James Reisman, and Anil Jain. Fingerprint matching using feature space correlation. In *Biometric Authentication: International ECCV 2002 Workshop Copenhagen, Denmark, June 1, 2002 Proceedings 1*, pages 48–57. Springer, 2002.

[94] Nist-nbis biometric image software.

[95] Craig I Watson and Charles L Wilson. Nist special database 4. *Fingerprint Database, National Institute of Standards and Technology*, 17(77):5, 1992.

[96] Craig I Watson. Nist special database 14. *Fingerprint Database, US National Institute of Standards and Technology*, 1993.

[97] Michael D Garris and Michael D Garris. *NIST special database 27: Fingerprint minutiae from latent and matching tenprint images*. US Department of Commerce, National Institute of Standards and Technology ..., 2000.

[98] Gregory P Fiumara, Patricia A Flanagan, John D Grantham, Bruce Bandini, Kenneth Ko, and John M Libert. Nist special database 300: uncompressed plain and rolled images from fingerprint cards. 2018.

[99] Gregory P Fiumara, Patricia A Flanagan, John D Grantham, Kenneth Ko, Karen Marshall, Matthew Schwarz, Elham Tabassi, Bryan Woodgate, and Christopher Boehnen. Nist special database 302: Nail to nail fingerprint challenge. 2019.

[100] Dario Maio, Davide Maltoni, Raffaele Cappelli, James L Wayman, and Anil K Jain. Fvc2002: Second fingerprint verification competition. In *2002 International Conference on Pattern Recognition*, volume 3. IEEE, 2002.

[101] Dario Maio, Davide Maltoni, Raffaele Cappelli, Jim L. Wayman, and Anil K. Jain. Fvc2004: Third fingerprint verification competition. In David Zhang and Anil K. Jain, editors, *Biometric Authentication*, pages 1–7, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[102] Kenneth Ko. User's guide to nist biometric image software (nbis), 2007-01-21 2007.

[103] Elham Tabassi, Martin Olsen, Oliver Bausinger, Christoph Busch, Andrew Figlarz, Gregory Fiumara, Olaf Henniger, Johannes Merkle, Timo Ruhland, Christopher Schiel, and Michael Schwaiger. Nist fingerprint image quality 2, 2021-07-13 04:07:00 2021.

[104] Arun Ross, Jidnya Shah, and Anil Jain. From template to image: Reconstructing fingerprints from minutiae points. *IEEE transactions on pattern analysis and machine intelligence*, 29:544–60, 05 2007.

[105] Raffaele Cappelli. Sfinge : an approach to synthetic fingerprint generation. 2004.

[106] Jadhav Sachin Namdeo. Generating, classifying and indexing large scale fingerprints. 2012.

[107] Joshua J. Engelsma, Steven A. Grosz, and Anil K. Jain. Printsgan: Synthetic fingerprint generator. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, page 1–14, 2022.

[108] Shervin Minaee and Amirali Abdolrashidi. Finger-gan: Generating realistic fingerprint images using connectivity imposed gan, 2018.

[109] Masud An-Nur Islam Fahim and Ho Yub Jung. A lightweight gan network for large scale fingerprint generation. *IEEE Access*, 8:92918–92928, 2020.

[110] Ataher Sams, Homaira Huda Shomee, and S. M. Mahbubur Rahman. Hq-fingan: High-quality synthetic fingerprint generation using gans. *Circuits Syst. Signal Process.*, 41(11):6354–6369, nov 2022.

[111] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[112] Andre Brasil Vieira Wyzykowski, Mauricio Pamplona Segundo, and Rubisley de Paula Lemes. Level three synthetic fingerprint generation. *2020 25th International Conference on Pattern Recognition (ICPR)*, Jan 2021.

[113] M. Sadegh Riazi, Seyed M. Chavoshian, and Farinaz Koushanfar. Synfi: Automatic synthetic fingerprint generation, 2020.

[114] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution, 2016.

[115] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network, 2017.

[116] Keivan Bahmani, Richard Plesh, Peter Johnson, Stephanie Schuckers, and Timothy Swyka. High fidelity fingerprint generation: Quality, uniqueness, and privacy. *2021 IEEE International Conference on Image Processing (ICIP)*, Sep 2021.

[117] Sharat Chikkerur, Venu Govindaraju, and Alexander N Cartwright. Fingerprint image enhancement using stft analysis. In *International Conference on Pattern Recognition and Image Analysis*, pages 20–29. Springer, 2005.

[118] Byung-Gyu Kim, Han-Ju Kim, and Dong-Jo Park. New enhancement algorithm for fingerprint images. In *2002 International Conference on Pattern Recognition*, volume 3, pages 879–882. IEEE, 2002.

[119] Jianwei Yang, Lifeng Liu, and Tianzi Jiang. Improved method for extraction of fingerprint features. In *Second International Conference on Image and Graphics*, volume 4875, pages 552–558. SPIE, 2002.

[120] Manhua Liu and Peng Qian. Automatic segmentation and enhancement of latent fingerprints using deep nested unets. *IEEE Transactions on Information Forensics and Security*, 16:1709–1719, 2020.

[121] Qinghua Zhang and Albert Benveniste. Wavelet networks. *IEEE transactions on Neural Networks*, 3(6), 1992.

[122] Harold H Szu, Brian A Telfer, and Shubha L Kadambe. Neural network adaptive wavelets for signal representation and classification. *Optical Engineering*, 31(9):1907–1916, 1992.

[123] Markus Lang, Haitao Guo, Jan Erik Odegard, C Sidney Burrus, and Raymond O Wells Jr. Nonlinear processing of a shift-invariant discrete wavelet transform (dwt) for noise reduction. In *Wavelet Applications II*, volume 2491, pages 640–651. SPIE, 1995.

[124] Maxim Kimlyk and Sergei Umnyashkin. Image denoising using discrete wavelet transform and edge information. In *2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*. IEEE, 2018.

[125] Kashif Rajpoot, Nasir Rajpoot, and J Alison Noble. Discrete wavelet diffusion for image denoising. In *International Conference on Image and Signal Processing*. Springer, 2008.

[126] Sami Hussien Ismael, Firas Mahmood Mustafa, and Ibrahim Taner Okümüs. A new approach of image denoising based on discrete wavelet transform. In *2016 World Symposium on Computer Applications & Research (WSCAR)*, pages 36–40. IEEE, 2016.

[127] M Mozammel Hoque Chowdhury and Amina Khatun. Image compression using discrete wavelet transform. *International Journal of Computer Science Issues (IJCSI)*, 9(4):327, 2012.

[128] Abdullah Al Muhit, Md Shabiul Islam, and Masuri Othman. Vlsi implementation of discrete wavelet transform (dwt) for image compression. In *2nd international conference on autonomous robots and agents*, volume 4, 2004.

[129] Hemalatha Kanagaraj and V Muneeswaran. Image compression using haar discrete wavelet transform. In *2020 5th International Conference on Devices, Circuits and Systems (ICDCS)*. IEEE, 2020.

[130] Woong Bae, Jaejun Yoo, and Jong Chul Ye. Beyond deep residual learning for image restoration: Persistent homology-guided manifold simplification. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017.

[131] Pengju Liu, Hongzhi Zhang, Wei Lian, and Wangmeng Zuo. Multi-level wavelet convolutional neural networks. *IEEE Access*, 7:74973–74985, 2019.

[132] Qiufu Li, Linlin Shen, Sheng Guo, and Zhihui Lai. Wavecnet: Wavelet integrated cnns to suppress aliasing effect for noise-robust image classification. *IEEE Transactions on Image Processing*, 30:7074–7089, 2021.

[133] Yiping Duan, Fang Liu, Licheng Jiao, Peng Zhao, and Lu Zhang. Sar image segmentation based on convolutional-wavelet neural network and markov random field. *Pattern Recognition*, 64:255–267, 2017.

[134] Wei-Peng Zhang, Qing-Ren Wang, and Yuan Y Tang. A wavelet-based method for fingerprint image enhancement. In *Proceedings. International Conference on Machine Learning and Cybernetics*, volume 4. IEEE, 2002.

[135] Harold C Burger, Christian J Schuler, and Stefan Harmeling. Image denoising: Can plain neural networks compete with bm3d? In *2012 IEEE conference on computer vision and pattern recognition*. IEEE, 2012.

[136] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *European conference on computer vision*. Springer, 2014.

[137] Hang Zhao, Orazio Gallo, Iuri Frosio, and Jan Kautz. Loss functions for image restoration with neural networks. *IEEE Transactions on computational imaging*, 3(1):47–57, 2016.

[138] Lin Zhang, Lei Zhang, Xuanqin Mou, and David Zhang. A comprehensive evaluation of full reference image quality assessment algorithms. In *2012 19th IEEE International Conference on Image Processing*. IEEE, 2012.

[139] Stefan Winkler and Sabine Susstrunk. Visibility of noise in natural images. In *Human Vision and Electronic Imaging IX*, volume 5292, pages 121–129. SPIE, 2004.

[140] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.

[141] Raffaele Cappelli, Dario Maio, and Davide Maltoni. Synthetic fingerprint-database generation. In *2002 International Conference on Pattern Recognition*, volume 3. IEEE, 2002.

[142] Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. A survey on contrastive self-supervised learning. *Technologies*, 9(1):2, 2020.

[143] Xiao Liu, Fanjin Zhang, Zhenyu Hou, Li Mian, Zhaoyu Wang, Jing Zhang, and Jie Tang. Self-supervised learning: Generative or contrastive. *IEEE transactions on knowledge and data engineering*, 35(1):857–876, 2021.

[144] Longlong Jing and Yingli Tian. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 43(11):4037–4058, 2020.

[145] Phuc H Le-Khac, Graham Healy, and Alan F Smeaton. Contrastive representation learning: A framework and review. *Ieee Access*, 8:193907–193934, 2020.

[146] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners, 2020.

[147] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning, 2020.

[148] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

[149] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments, 2021.

[150] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 297–304. JMLR Workshop and Conference Proceedings, 2010.

[151] Carl Doersch. Tutorial on variational autoencoders, 2016.

[152] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollar, and Ross Girshick. Masked autoencoders are scalable vision learners. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2022.

[153] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers, 2021.

[154] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. Simmim: a simple framework for masked image modeling. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2022.

[155] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. 2018.

[156] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[157] Tianyu Gao, Xingcheng Yao, and Danqi Chen. Simcse: Simple contrastive learning of sentence embeddings, 2022.

[158] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations, 2018.

[159] Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization. *Lecture Notes in Computer Science*, page 649–666, 2016.

[160] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. *Lecture Notes in Computer Science*, page 69–84, 2016.

[161] William Lotter, Gabriel Kreiman, and David Cox. Deep predictive coding networks for video prediction and unsupervised learning, 2016.

[162] Ishan Misra, C. Lawrence Zitnick, and Martial Hebert. Shuffle and learn: Unsupervised learning using temporal order verification, 2016.

[163] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers, 2021.

[164] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[165] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction, 2021.

[166] Shuangfei Zhai, Navdeep Jaitly, Jason Ramapuram, Dan Busbridge, Tatiana Likhomanenko, Joseph Yitan Cheng, Walter Talbott, Chen Huang, Hanlin Goh, and Joshua Susskind. Position prediction as an effective pretraining strategy, 2022.

[167] Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2022.

[168] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning, 2020.

[169] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations, 2020.

[170] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning, 2020.

[171] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks, 2019.

[172] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.