ON TRAINING GENERATIVE ADVERSARIAL NETWORK FOR ENHANCEMENT OF LATENT FINGERPRINTS

Indu Joshi*, Adithya Anand, Sumantra Dutta Roy and Prem Kumar Kalra

Indian Institute of Technology Delhi
*Corresponding Author: Indu.Joshi@cse.iitd.ac.in

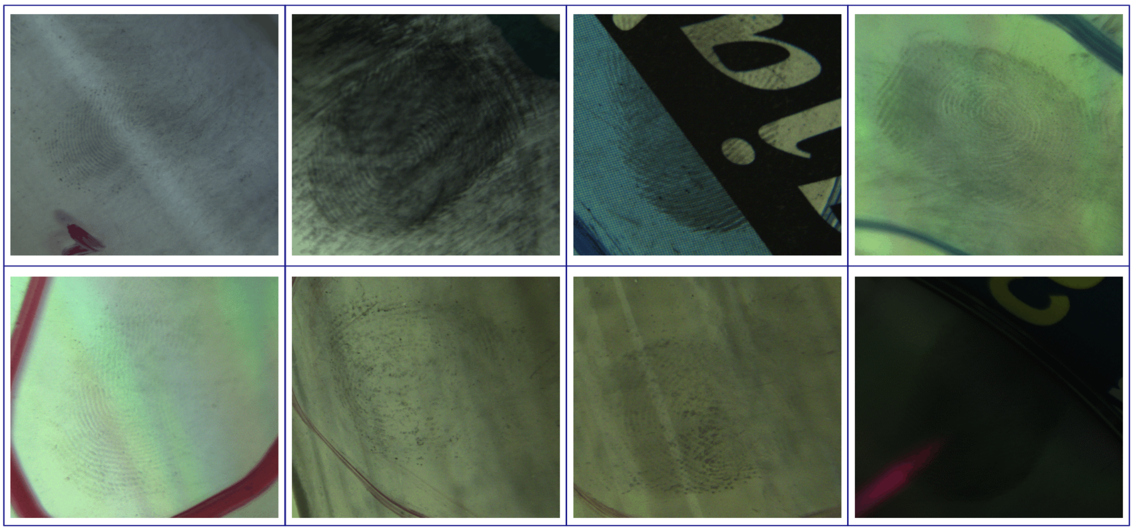
Abstract. Latent fingerprints are the fingerprints which are left unintentionally on a surface, while touching it. These are of great interest for the forensics experts for criminal identification. Latent fingerprints usually possess high non-linear distortion. Furthermore, these may be overlapping with background text or other fingerprints. These fingerprints can be extracted from different surfaces leading to the varying background. The presence of structured and unstructured background noise adversely affects minutiae (ridge bifurcation/ridge ending) extraction in latent fingerprints which in turn leads to poor matching performance. A latent fingerprint enhancement algorithm removes the background noise and predicts the missing ridge information. It also improves the ridge clarity which helps to improve minutiae extraction and thereby improving matching performance. Traditionally, latent fingerprints are enhanced by approximating the orientation field and then applying contextual filtering using the approximated orientations. However, recently the attention has been shifted towards developing models which can directly denoise the fingerprints and reconstruct the missing ridge structure without explicitly estimating the orientation field.
Inspired by the success of Generative Adversarial Network (GAN) in image processing applications, we propose a GAN based latent fingerprint enhancement model. However, one of the key issues with GANs is that they are difficult to train. Through this work, we contribute our efforts towards sharing details on successfully training a generative adversarial network. The proposed latent fingerprint enhancement model preserves ridge structure including minutiae. We discuss the role of training data i.e. various noise models which should be considered for modeling a latent fingerprint, during training a GAN. In addition to this, we discuss the significance of choice of loss function and the role of hyper parameters such as batch size, weight of each loss term and no. of epochs for training the GAN. We evaluate the proposed enhancement model on publicly available latent databases: Indraprastha Institute of Information Technology Delhi Multi-sensor Optical and Latent Fingerprint (IIITD-MOLF) and Indraprastha Institute of Information Technology Delhi Multi-surface Latent Fingerprint (IIITD-MSLF).

Keywords: Latent Fingerprints, Denoising, Generative Adversarial Network, Enhancement.

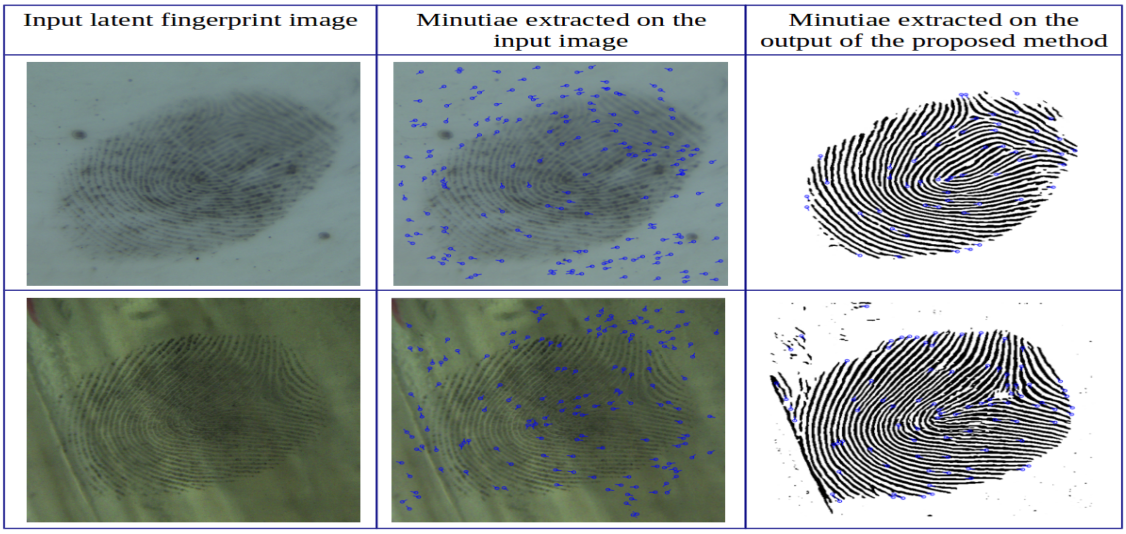
1 Introduction

Latent fingerprints are the impressions of the ridges on the fingertips which are unintentionally deposited on the surface of an object when the subject touches it. These fingerprints are lifted by forensic experts using specialised techniques like dusting or chemical processing. Latent fingerprints have unclear ridge structure, partial ridge information and uneven contrast between ridges and valleys. They also possess structured noise due to overlapping text, lines, stains and sometimes overlapping fingerprints in the background. Fig. 3.1(a) showcases sample latent fingerprint images from IIITD-MSLF database [1].

Latent fingerprints picked up from the crime scene are matched with fingerprints in the law-agency's fingerprint database, to find crime suspects. Standard fingerprint matching systems are designed for good quality fingerprints. However, due to the poor quality of latents, standard fingerprint feature (minutiae) extractors which perform well on plain and rolled fingerprints often fail on latent fingerprint [2]. Fig. 3.1(b) showcases that many times, true minutiae are missed due to smudged and blurred ridges and many spurious minutiae are extracted due to background noise. As a result, the matching accuracy achieved by the standard fingerprint matchers on latent fingerprints is far from satisfactory to be used for latent fingerprint matching.



(a)



(b)

Fig. 3.1: (a) Sample latent fingerprints from IIITD-MSLF database depicting background noise, degraded fingerprint ridges, background with textures and multiple fingerprints overlapping with each other (b) Fingerprints exhibiting the improvement of minutiae detection on enhanced images generated by proposed algorithm. Left column exhibits the original fingerprints, middle column showcases the minutiae detected (shown by blue dots) on original fingerprints using NBIS tool [5]. Right column shows improved minutiae detection post enhancement.

Due to this, latent fingerprints are manually matched by the latent fingerprint examiners which pose a huge burden on them. Further, studies have reported inconsistency across evaluations of latent fingerprint examiners [3] [4]. This poses a serious need to automate the process of latent fingerprint matching which can facilitate fast and accurate matching performance over the whole fingerprint database and not just a small subset of suspects.

One of the key techniques to improve the latent fingerprint matching performance is an enhancement module. An enhancement algorithm improves the contrast between ridges and valleys, removes background noise and predicts the missing ridge information and thus facilitates correct minutiae extraction, in turn improving the matching performance. Fig. 3.2 depicts the overall framework of latent fingerprint matching.

In this chapter, we provide details on training generative adversarial network (GAN) which removes the background noise and improves ridge clarity while preserving the ridge structure

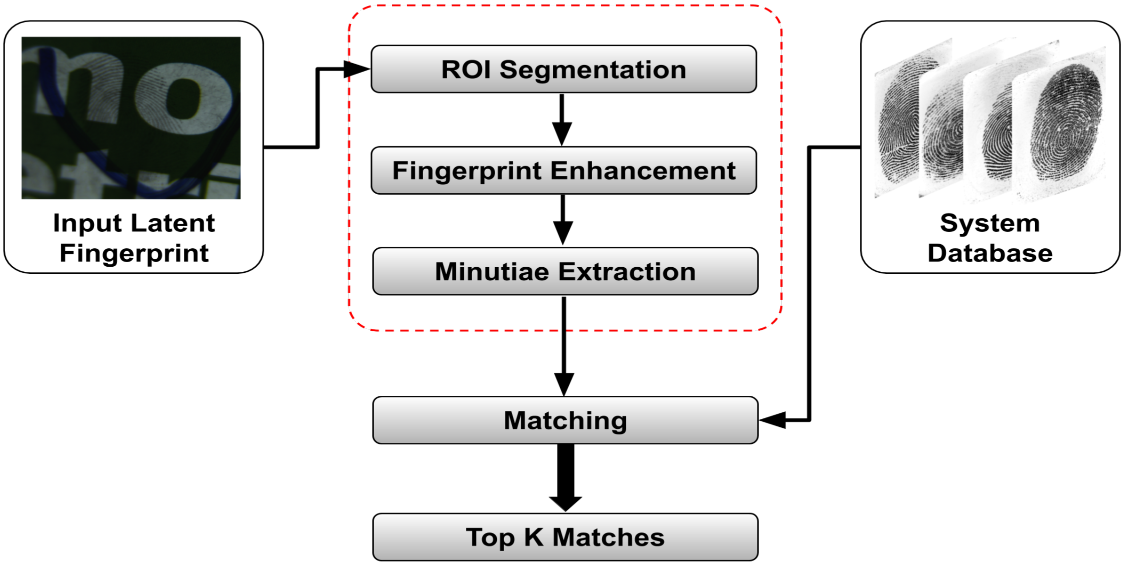


Fig. 3.2: Latent fingerprint matching framework.

including minutiae. Several studies have pointed out the difficulties in training a GAN based model. We discuss the role of loss function, various hyper-parameters and training data in successfully training a GAN for latent fingerprint enhancement.

2 Related Work

The early literature on latent fingerprint enhancement focuses on accurate estimation of orientation field of ridges in latent fingerprints. The estimated orientations are then fed to the Gabor filter to enhance latent fingerprints. Given below are the approaches of latent fingerprint enhancement which approximate orientation field and utilize it to enhance latent fingerprints:

Yoon et al. [6] propose an orientation estimation algorithm that requires manually marked ROI (Region of Interest) and singular points. At first, orientation skeleton image is derived from Verifinger [7] (the state-of-art commercial fingerprint matching tool). From these orientations, reliable and unreliable blocks are found out. Reliable blocks have orientations coherent with the neighbouring blocks. For the unreliable blocks, the re-estimation of orientations is performed by interpolations of orientations from the reliable blocks. Using the interpolated orientations, fingerprint rotation and skin distortion model is estimated. Further, computation of orientations from singular points is done using zero-pole technique. Finally, orientation is estimated using orientation obtained through zero-pole method and estimated distortion model. Gabor filtering is applied on the estimated orientation to obtain the enhanced image.

Yoon et al. [8] estimate orientation field estimation assuming that the manually marked ROI and singular points are available for the input latent fingerprint image. Initial orientation field is computed by STFT (Short Time Fourier Transform) enhancement algorithm. However, the performance of STFT can be easily affected by the unstructured background noise. They employ a two-level approach in which firstly they merge compatible orientation elements in a neighbourhood into an orientation group. Next, they generate top-ten best global orientation using R-RANSAC (Randomized Random sample consensus). Gabor filters with all the ten orientations are employed to obtain ten enhanced latent fingerprint images. Matching is done with all the ten images and maximum match score serves as the final output match score of the latent.

Feng et al. [9] argue that the orientation estimation is analogous to spelling correction in a sentence. They propose to create a dictionary of orientation patches estimated from good quality fingerprint patches. Creating dictionary helps to eliminate non-word errors i.e. predicting such orientations which cannot exist in real-life. They further discuss that just as contextual information can help in spelling correction, similarly orientation of neighbouring patches should be utilized for the estimation of orientation of a given patch. To begin with, they compute an initial estimate of orientation field using STFT. They, then, compare the initial estimate with each dictionary element and identify potential candidates. They use compatibility between neighbouring patches to find the optimal candidate. Orientation information of all orientation patches is then summarized to obtain the final orientation field.

Yang et al. [10] utilize spatial locality information present in fingerprints to improve the quality of the estimate. Authors claim that only specific orientations occur at a given location, e.g. the orientations at the middle of fingerprints will be different than the orientations at the top of fingerprints. In order to exploit this information, they introduce localized dictionaries i.e. create a dictionary for every location in a fingerprint. Due to this, each dictionary contains only a limited number of orientations leading to faster dictionary look-ups. Moreover, this technique leads to even fewer non-word errors.

Chen et al. [11] observe that the average size of noise is not same in all latent fingerprints. Rather, it varies across different qualities of latent fingerprints. For a poor quality image, one can obtain better results by using a dictionary with bigger patch size and vice-versa. So, a dictionary created for only a particular size of orientation patches will not work for all latent fingerprints. The authors solve this problem by creating multi-scale dictionaries i.e. dictionaries of different patch sizes. They use compatibility between neighbours across different scales to find the optimal orientation patch for a given estimate.

Cao and Jain [12] discuss the limitations of dictionary-based methods. They further argue that there is a need for methods which can learn the orientation field from poor quality latent fingerprints. They formulate estimation of orientation field from a fingerprint image as a classification problem. They address this problem using a CNN based classification model. The real challenge in using a deep architecture is to have a large amount of latent fingerprints for training the network. For this purpose, they propose a model to simulate texture noise as present in latent fingerprints. Several structured and unstructured noise patterns are injected into good quality fingerprints for synthesizing latent fingerprints. K-means clustering is performed on orientation patches of good quality images to select 128 representative orientation patch classes. They extract 1000 orientation patches for each orientation class and trained the network with corresponding simulated latent. After training the model for each patch in input latent fingerprint, an orientation class is predicted by the model.

Liu et al. [13] pose the estimation of orientations as a denoising problem and propose sparse coding for denoising of orientation patches. Authors create multi-scale dictionaries from good quality fingerprints. After computing the initial estimate, they, then, reconstruct the orientation using dictionary of smallest size with sparse coding. The quality of an orientation patch is then estimated based on compatibility with neighbours. If the quality is below a certain threshold, then the orientation patch is reconstructed using a dictionary of bigger patches. This process is continued until the quality of reconstructed orientation patch is satisfactory.

Chaidee et al. [14] propose sparse coded dictionary learning in frequency domain which fuses responses from Gabor and curved filters. In the offline stage, dictionary is constructed from the frequency response. In the online stage, spectral response is computed which is then encoded by spectral encoder. The sparse representation of the spectral code is computed and then decoded by spectral decoder to reconstruct the fourier spectrum. A weighted sum of the reconstructed image is obtained from both the filters is computed to obtain the final enhanced image. Recently, the attention has been shifted to straight away generate enhanced fingerprint without explicitly approximating orientation field. We now describe such latent fingerprint enhancement algorithms:

Qu et al. [15] propose a deep regression neural network which outputs orientation angle values. The input latent fingerprint image is first pre-processed using total variation decomposition and Log-Gabor filtering. The pre-processed latent is then given as an input to the network and orientation is estimated. Boosting is performed to further improve the prediction accuracy.

Li et al. [16] propose a multi-task learning based enhancement algorithm which works on the patch level. Input latent fingerprint image is pre-processed using Total Variation Decomposition and the texture component is used as an input for the proposed model. Proposed solution is based on encoder-decoder architecture trained with a multi-task learning loss. One branch enhances the latent fingerprint and the other branch predicts orientation for the input image. This algorithm requires orientation field information as a part of training data to train the network to generate the enhanced fingerprint image. Thus, this algorithm is beyond the scope of this chapter.

Svoboda et al. [17] suggest an end-to-end convolutional auto-encoder architecture which implicitly minimizes orientation and gradient loss between the target enhanced fingerprint and the fingerprint produced by their model. The objective function is designed such that it only minimizes $l2$-loss and it cannot address perceptual information. A brief summary of limitation of the state-of-the-art is provided at Table 3.1.

To summarize, the traditional state-of-the-art latent fingerprint enhancement algorithms focus on accurate orientation estimation for latent fingerprints and exploit only Gabor filters to enhance latent fingerprints. Recent state-of-art techniques, on the other hand, propose learning-based end-to-end latent fingerprint enhancement models which directly generate enhanced fingerprints without only relying on Gabor filters. The weights of the kernels in convolutional neural networks are rather learnt for the problem in hand. However, none of the above mentioned latent fingerprint enhancement models exploit the perceptual information in the fingerprints.

Algorithm	Proposed Approach	Limitation	Reference
Classical Image Processing and hand-crafted models	Orientation estimation using zero-pole method and distortion model	Requires manually marked ROI and singular points	[6]
	R-RANSAC is used to find top-ten global orientations. All the ten enhanced images are used for matching	Requires manually marked ROI and singular points. Matching with ten enhanced images is an overhead	[8]
Dictionary Learning	Dictionary learning based orientation estimation	Incorrect estimation around singular points, high computation time	[9]
	Localized dictionary learning based orientation estimation	Algorithm first performs pose estimation and then orientation estimation leading to high computational complexity	[10]
	Multi-scale dictionary learning based orientation estimation	Global multi-scale dictionaries are used due to which local apriori fingerprint information is not utilized	[11]
	Spectral dictionary	Requires manually marked core points	[14]
	Sparse coded dictionary learning based orientation estimation	Global multi-scale sparse coded dictionaries are used due to which local apriori fingerprint information is not utilized	[13]
Deep Learning	Convolutional neural network based classification for orientation estimation	Number of orientation patch classes is very limited, due to which the orientation estimation may not be accurate	[12]
	Deep regression neural network for orientation estimation	Requires pre-processing before orientation estimation. Moreover, algorithm is not evaluated on any of the publicly available latent fingerprint databases	[15]
	Multi-task learning based autoencoder	The autoencoder is designed for pre-processed latent fingerprints	[16]
	Convolutional auto-encoder that minimizes orientation and gradient loss	Fails to preserve minutiae in case of poor quality input images	[17]

Table 3.1: Table summarizing the literature on latent fingerprint enhancement.

Generative adversarial networks (GANs) generate sharper images compared to auto-encoders which generate blurred images. As a result, GANs are better suited for generating fingerprint images as they can generate sharp images with clear ridge structure and good ridge-valley contrast. This inturn facilitates improved minutiae extraction and matching performance.

The information on training GAN for latent fingerprint enhancement provided in this chapter is based on the latent fingerprint enhancement algorithm proposed by Joshi et al. [18]. The enhancement model proposed by the authors is trained not only with the reconstruction loss to preserve the ridge structure, but it also limits spurious pattern generation by employing a classification network trained with an adversarial loss to classify the reconstructed image as real or fake. Furthermore, the proposed GAN model is trained on synthetic latent fingerprint images due to which the training is not affected by the limited availability of publicly available latent fingerprint images.

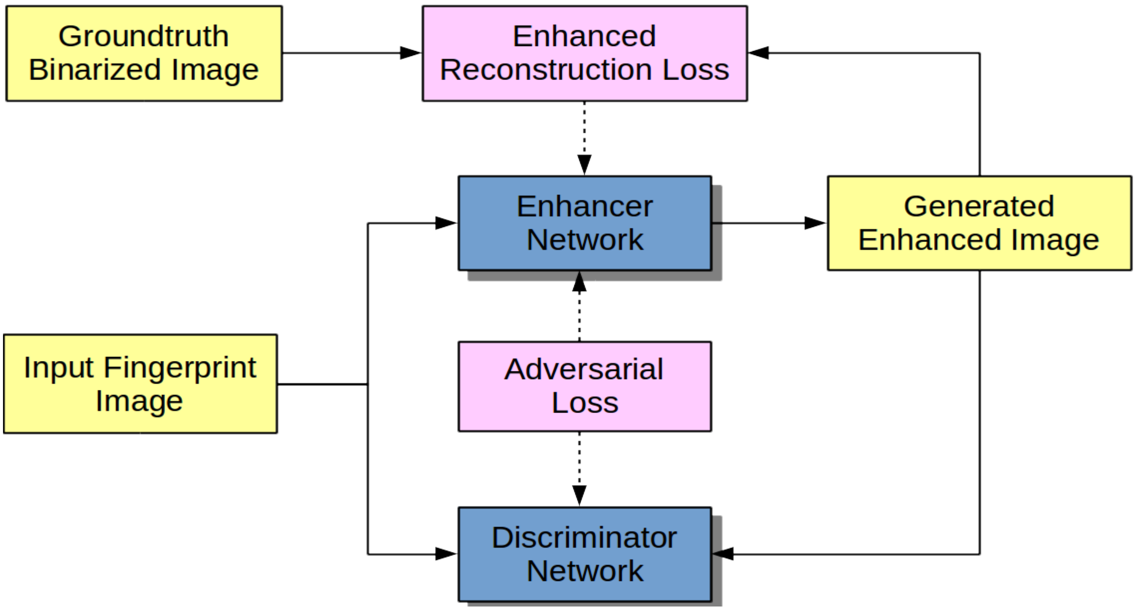


Fig. 3.3: Proposed model for enhancement of Latent fingerprints. The back propagation of losses while training Enhancer network and Discriminator network is shown by dotted lines.

3 Proposed Algorithm

3.1 Problem Formulation and Objective Function

We propose a conditional generative adversarial networks [19] [20] based latent fingerprint enhancement algorithm. Given a latent fingerprint, the proposed algorithm generates a fingerprint image with clear ridge structure and removes structured and non-structured background noise present in a latent fingerprint. The motivation behind using a conditional GAN is that the generator has to not only generate a "real-looking" binarized fingerprint image but it should also generate a fingerprint which has similar ridge structure as the input latent fingerprint image. Thus, we formulate latent fingerprint enhancement as a conditional GAN based image-to-image translation problem [21].

The proposed model has two networks: a latent fingerprint enhancer network and an enhanced fingerprint discriminator (See Fig. 3.3). For a given latent fingerprint image x, enhancer network generates a binarized enhanced image $\mathcal{E}nh_L(x)$). The enhancer network learns the transformation from a latent fingerprint to a binarized enhanced image, while preserving the overall ridge structure and ridge features including minutiae, without compromising the identity information in the fingerprint. The discriminator network classifies a given enhanced image as real or fake. Fig. 3.3 depicts the proposed model for latent fingerprint enhancement. The loss function optimized by the proposed model is described below:

(1) Adversarial Loss:

$$L_{adv} = E_{(x,y)\sim p(x,y)}[log(\mathcal{D}is_E(x,y))] + E_{x\sim p_x(x)}[log(1 - \mathcal{D}is_E(x, \mathcal{E}nh_L(x)))]$$

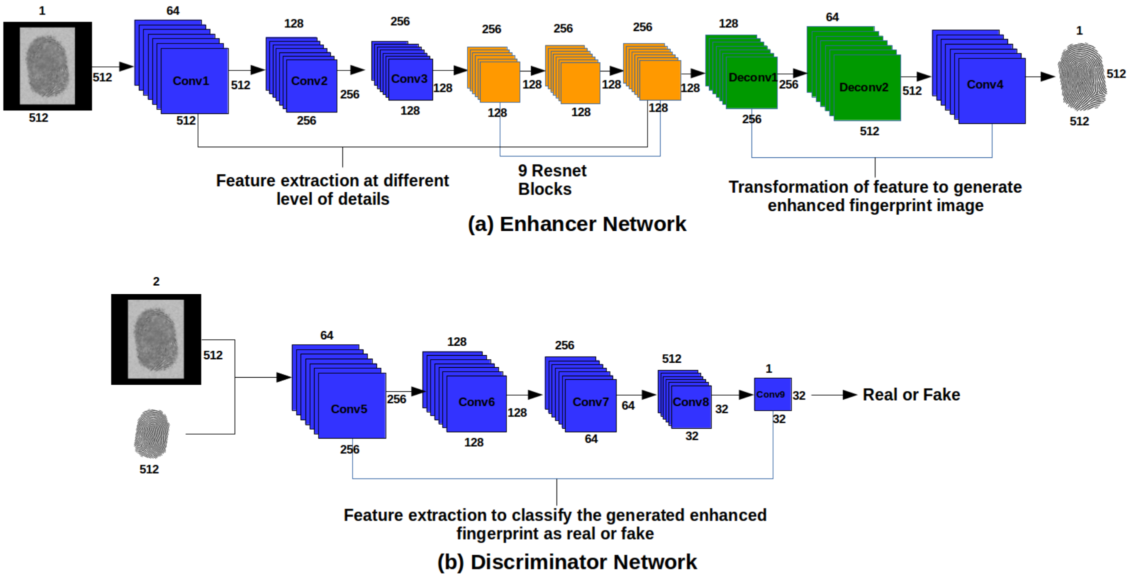


Fig. 3.4: Architecture of Enhancer ($\mathcal{E}nh_L$) and Discriminator ($\mathcal{D}is_E$)

Enhancer network is trained such that the adversarial loss is minimized. On the other hand, discriminator network is trained to maximize the adversarial loss. A penalty is imposed on the enhancer network if the image generated by the enhancer network ($\mathcal{E}nh_L(x)$)), is deemed fake by the discriminator. Due to this loss, enhancer network learns the necessary transformation and associated features required to generate an enhanced fingerprint from a given latent fingerprint image.
Discriminator network is penalized if it misclassifies an enhanced fingerprint image generated by the enhancer network as a real fingerprint. As a result, discriminator learns the discriminating features for differentiating the enhanced images produced by the enhancer from the ground truth binarized images.
Note that the discriminator is conditioned by the input latent fingerprint image so that the discriminator network doesn't just classify an enhanced image as real or fake but the discriminator can also classify whether the enhanced fingerprint image has the ridge structure similar to the input latent fingerprint image.

(2) Enhanced Fingerprint Reconstruction Loss:

$$L_{rec} = ||y - \mathcal{E}nh_L(x))||_1$$

The task of generating a binarized enhanced image corresponding to an input latent fingerprint image is an ill-posed problem with only adversarial loss. We include fingerprint reconstruction loss into the objective function. This loss only penalizes the enhancer network. It guides the enhancer network to generate enhanced fingerprint similar to the ground-truth binarized fingerprint image. The reconstruction loss facilitates the enhancer network to learn to preserve low-frequency details in the enhanced image. *l1* norm is used in the loss function to encourage the enhancer to produce sharp images. *l2* norm is not used as it generates blurred images.

(3) Overall Loss: The final objective function is given as:

$$\begin{aligned} min_\alpha max_\beta \;\; [E_{(x,y)\sim p(x,y)}[log\mathcal{D}is_E(x,y)] + E_{x\sim p_x(x)}[log(1 - \mathcal{D}is_E(x, \mathcal{E}nh_L(x))) \\ +\lambda||y - \mathcal{E}nh_L(x))||_1]] \end{aligned}$$

where α and β denote the parameters of enhancer and discriminator respectively. λ is the weight parameter for the reconstruction loss.

Reconstruction loss helps to preserve the low frequency details in the fingerprint image. However, fingerprints are oriented textured patterns which have a lot of high frequency details. To ensure that the proposed model is able to capture high frequency details, we use a patch GAN based model which classifies each 8×8 patch as real or fake. Furthermore, reconstruction loss is a pixel based loss which assumes that each output pixel is independent of its neighbouring pixels. Patch GAN on the other hand, considers the joint distribution of the pixels in a patch which introduces a texture loss which inturn forces the enhancer network to preserve fine ridge details including minutiae and thus helps to preserve the identity information in the fingerprint image.

3.2 Training Data Preparation

The proposed model is a supervised generative model which is trained to output an enhanced image given an input latent fingerprint image. Being a supervised model, it requires paired training data of latent fingerprints and their corresponding enhanced binarized images. However, there are no publicly available latent fingerprints datasets which have latent fingerprints and their corresponding enhanced images. Additionally, lack of large latent fingerprint database further complicates the training of a deep neural network based latent fingerprint enhancement model. Thus, we need to generate synthetic latent fingerprints which have similar noise characteristics as observed in real latent fingerprints (See Fig. 3.5) for training the proposed enhancement model.

The proposed model is trained on 9042 synthetic latent fingerprint images and 2423 fingerprint images from National Institute of Standards and Technology Special Database 4 (NIST SD4) and their corresponding binarized fingerprints. Due to training on synthetic latent fingerprints, the training of the proposed model is not affected by the limited availability of the latent fingerprint database. We now give details on preparing the training data for the proposed model.

(1) Datasets for preparing the training data

- **(i) Anguli:** Anguli [22] is an open-source implementation of state-of-the-art synthetic fingerprint generator SFinGe [23], which simulates synthetic live fingerprints with similar features as real-live fingerprints. It can generate multiple impressions of a fingerprint with varying level of noise.

- **(ii) NIST SD4:** NIST SD4 [24] is a publicly available fingerprint database which has 2000 rolled fingerprints. These are inked fingerprints with uniformly distributed fingerprint pattern type, namely left loop, right loop, arch, tented arch and whorl. Due to the uniform distribution of pattern type, the training dataset covers varieties of ridge patterns. Further, as these fingerprints are inked prints, they have similar characteristics of non-uniform ink similar to latent fingerprints which have non-uniform powder content in many patches. We use NIST SD4 fingerprints with NIST Finger Image Quality 2 (NFIQ2) [25] quality score greater than or equal to 70. (NFIQ2 is an open-source state-of-the-art fingerprint quality assessment algorithm which gives a quality score in the range 1-100 to each fingerprint image where 1 denotes the worst quality and 100 denotes the best quality.) Although it is helpful to include poor quality inked prints as the training data, however, the ground truth binarization achieved through NBIS on poor quality fingerprints is poor which can adversely affect the performance of model. So, we only use good quality NIST SD4 fingerprints for training the model.

(2) Generation of Synthetic Latent Fingerprints

Latent fingerprints due to their acquisition conditions are often blurred, have structured noise such as lines, overlapping text and sometimes overlapping fingerprints. We add the following noise into good quality fingerprints generated by Anguli to create a representative synthetic latent fingerprints dataset for training the proposed model:

- **(i) Line-like noise:** It has been observed that line-like noise due to their similarity with fingerprint ridges often lead to failure of standard fingerprint matching algorithms. To simulate line-like noise, we blend fingerprint images with straight lines having different orientations and different width.

Fig. 3.5: Sample images showcasing the training dataset. The eleven fingerprints (from top-left) have the same binarized ground-truth image (bottom-right image). Varying textures and backgrounds are used for training the algorithm for simulating conditions of acquisition of latent fingerprint.

(ii) Blurring: Sometimes smudging of fingerprint ridges leads to missing minutiae. We observe that latent fingerprint often have non-uniform smudge patterns. To make the model invariant towards different levels of smudging, we add different level of gaussian noise on randomly selected fingerprint patches. The different patch size used are 10×10 and 40×40. The blur radius=2 is used for gaussian noise.

(iii) Overlapping text and fingerprints: Latent fingerprints have complex background noise which can have overlapping text and sometimes overlapping fingerprints. To simulate those scenarios, we blend fingerprint images with text images of varying fonts and styles. We also blend fingerprint images with partial fingerprint images to address challenges of overlapping fingerprints.

(iv) Different Surfaces: Latent fingerprints can be collected from different surfaces. Surfaces can be plane/curved, porous/non-porous, shiny or can have uniform background. It has been reported that the surfaces which have high reflectance generate occluded ridge patterns [1]. Further, the area and the quality of latent fingerprint left on a surface varies depending on the pressure exerted by the finger, surface characteristics and adherence of the finger's natural secretions on that surface. Some surfaces have poor adherence property due to which the latent fingerprint deposited on such surface is often partial. To train the proposed model to be invariant towards various intra-class variations introduced due to various surfaces, we blend fingerprint image with varying textures such as wood surface, cardboard surface, plastic and glass-surface.

(3) Ground-truth Binarization

Ground-truth binarized image to train the proposed model is obtained using NIST Biometric Image Software (NBIS). A fingerprint image is binarized by NBIS based on the ridge flow direction. The image is divided into 7×9 grids, if there is a ridge pattern in a grid, the grid is rotated so that the grid is parallel to the ridge flow direction. For the pixel of interest, the neighbourhood grey values which also lie in the rotated grid are analyzed to label a pixel as black or white.

3.3 Network Architecture and Training Details

(1) Enhancer Network: Enhancer network has an encoder-decoder (autoencoder) architecture. Convolutional layers(Conv1, Conv2, Conv3) in the network extract features at different scales from the input latent fingerprint image capturing coarse to fine level details (See Fig. 3.4). ResNet blocks help to circumvent the problem of vanishing gradient while training a deep network. Decoder layers(Deconv1, Deconv2 and Conv4) transform the features extracted from the latent fingerprint to an enhanced binarized fingerprint image.

(2) Discriminator Network: The input latent fingerprint and binarized image are concatenated along the input channel dimension so that the discriminator can classify whether the binarized image corresponds to the input latent fingerprint image. Discriminator has a typical architecture as used in image classification. The convolutional layers in the discriminator(Conv5, Conv6, Conv7, Conv8 and Conv9) extract features at different scales capturing at different levels which helps the discriminator to classify an input fingerprint image as real or fake.

The details of the network architecture are given in Table 3.2. Adam optimizer is used to optimise the objective function. The following hyper-parameters are used: learning rate=0.02, β_1=0.5, β_2=0.999, λ=10 and batch size=2. The model is trained on 2 GPUs each with 12 GB RAM.

4 Performance Evaluation

4.1 Databases and tools used

The proposed model is evaluated on two publicly available latent fingerprint databases:

Block	Layers	Kernels	Size	Stride	Padding
Conv1	Convolutional Layer + Batch Normalization + ReLu	64	7	1	3
Conv2	Convolutional Layer + Batch Normalization + ReLu + Convolutional Layer + Batch Normalization	128	3	2	1
Conv3	Convolutional Layer + Batch Normalization + ReLu + Convolutional Layer + Batch Normalization	256	3	2	1
ResNet Block	Convolutional Layer + Batch Normalization + ReLu + Conv Layer + Batch Normalization	256	3	2	1
Deconv1	Convolutional Layer + Batch Normalization Layer + ReLu + Convolutional Layer + Batch Normalization	128	3	2	1
Deconv2	Convolutional Layer + Batch Normalization + ReLu + Conv Layer + Batch Normalization	64	3	2	1
Conv4	Convolutional Layer + Tanh	1	7	1	3
Conv5	Convolutional Layer + LeakyReLu	64	4	2	1
Conv6	Convolutional Layer + Batch Normalization + LeakyReLu	128	4	2	1
Conv7	Convolutional Layer + Batch Normalization + LeakyReLu	256	4	2	1
Conv8	Convolutional Layer + Batch Normalization + LeakyReLu	512	4	1	1
Conv9	Convolutional Layer	1	4	1	1

Table 3.2: Architecture of $\mathcal{E}nh_L$ and $\mathcal{D}is_E$.

(1) **IIITD-MOLF Database [26]:** IIITD-MOLF is the biggest latent fingerprint database which is available in the public domain. It has latent fingerprints and live fingerprints acquired through different optical sensors. These fingerprints are collected from 100 subjects. This database has 4400 latent fingerprints and 4000 live fingerprints corresponding to each sensor.

(2) **IIITD-MSLF Database [1]:** IIITD-MSLF database has latent fingerprints extracted from 8 different surfaces like transparent glass, compact disc, ceramic mug, hardbound cover etc. It has 551 latent fingerprints of 51 subjects.

The standard latent fingerprint database provided by NIST, NIST-SD27 has now been removed from the public domain due to which we cannot evaluate the proposed model on NIST-SD27 database. The proposed model is designed for the standard sized 500 dpi fingerprint image whose spatial dimensions are 512×512 pixels. The latent fingerprints are pre-processed and zero-padded to have a fixed size of 512×512. Table 3.3 provides the list of publicly available tools used in this work.

Tool	**Purpose**	**Usage**
MINDTCT module of NBIS	Minutiae extraction	During testing, to extract minutiae from enhanced image and gallery images
NFIQ module of NBIS	Evaluates fingerprint image quality	During testing, to evaluate quality of enhanced fingerprints
BOZORTH module of NBIS	To match fingerprints	During testing, to perform fingerprint matching on minutiae extracted by MINDTCT
MCC fingerprint matcher	To match fingerprints	During testing, to perform fingerprint matching on minutiae extracted by MINDTCT
NFIQ2	Evaluates fingerprint image quality	To evaluate quality of NIST SD4 images and keep good quality images for training the model
Binarization module of NBIS	Binarize the fingerprint image	To generate the ground-truth binarization of training images

Table 3.3: Table summarizing the publicly available tools used.

4.2 Evaluation Criteria

Every fingerprint enhancement algorithm is designed to increase the clarity of ridges and valleys while preserving the ridge details to improve minutiae extraction and thereby improving fingerprint matching performance. We evaluate the proposed enhancement algorithm using the metrics given below:

(1) **Fingerprint Quality Analysis:** Quality of a fingerprint image is determined as the ability of a fingerprint matcher to correctly match the image. Poor quality fingerprints often result in poor matching performance. We evaluate the fingerprint quality of latent fingerprints before and after enhancement using NIST Finger Image Quality (NFIQ) module of NBIS. NFIQ calculates quality of a fingerprint image using features such as: clarity of ridges and valleys, number of minutiae, size of the fingerprint image etc. NFIQ scores a fingerprint image between 1 and 5 where 1 signifies the best fingerprint image quality and 5 means the worst quality. We compare the histogram of quality scrores obtained by NFIQ before and after enhancement. Another publicly available tool to evaluate the quality of fingerprint images is NFIQ2 [25] which returns a score between 1-100. NFIQ2 is a more robust fingerprint quality assessment metric than NFIQ. However, NFIQ2 fails to process raw latent fingerprint images of IIITD-MOLF database. As a result, we only compare fingerprint quality score obtained using NFIQ.

(2) Ridge Structure Preservation: The most crucial factor for any fingerprint enhancement is that it should retain the ridge structure while improving clarity of ridges and valleys. To showcase ridge structure preservation (including minutiae) by the proposed model, we synthetically generate some test cases by adding noises and backgrounds on good quality fingerprints. We showcase the similarity between ground-truth binarization and the enhanced fingerprint image generated by the proposed algorithm using the following two measures:

  (i) We calculate **Structural Similarity Index Metric (*SSIM*)** [27] between the ground-truth binarized image and the enhanced fingerprint. SSIM is a metric which computes similarity between image a and image b based on the contrast, luminance and structure.

$$SSIM(a,b) = \frac{(2\mu_a\mu_b + C_1)(2\sigma_{ab} + C_2)}{(\mu_a^2 + \mu_b^2 + C_1)(\sigma_a^2 + \sigma_b^2 + C_2)}$$

  where μ_a, μ_b are the mean, σ_a, σ_b are the standard deviation and σ_{ab} is the covariance between image a and image b.

  (ii) We also calculate **match score (using Bozorth)** between ground-truth binarized image and the enhanced image generated by the proposed model. High match scores demonstrate that the proposed algorithm preserves minutiae while enhancing the input latent fingerprint image.

(3) Matching Performance: The ultimate success of a fingerprint enhancement algorithm is when it is able to improve the fingerprint matching performance. We extract minutiae using MINDTCT module of NBIS and use Bozorth and Minutia Cylinder Code (MCC) [28], [29], [30] fingerprint matchers to evaluate fingerprint matching performance. We compare matching performance before and after enhancement using Rank-50 accuracy. Rank-k accuracy is defined as:

Rank-k accuracy= no. of probe fingerprint for which the matching fingerprint in gallery achieved top-k scores×100/total no. of probe fingerprints

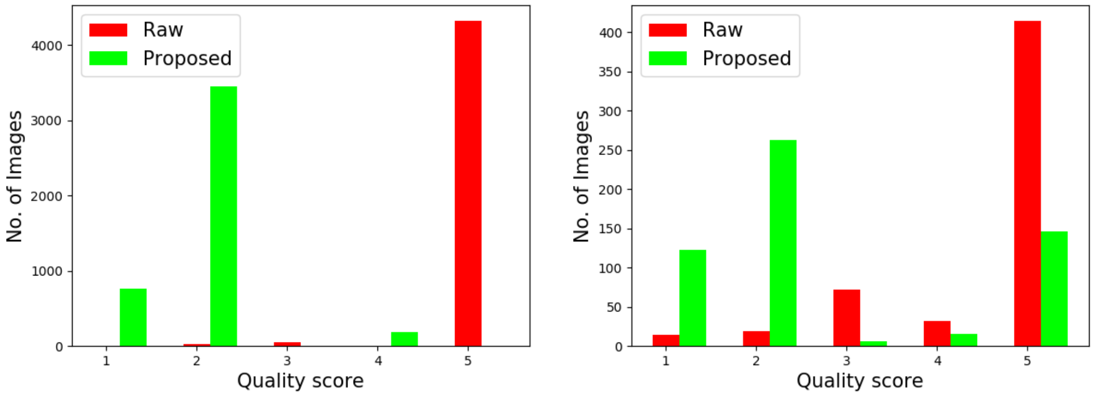
We also plot Cumulative Matching Curve (CMC curve), which is a Rank-k accuracy plot over varying values of k. CMC curve is a standard summarization technique to quantify the matching performance of a closed-set identification system. We compare CMC curve before and after enhancement in Fig. 3.11.

5 Results and Analysis

(1) Fingerprint Quality Analysis: Fig. 3.7 represents the histogram of NFIQ scores before and after enhancement. The average NFIQ score has improved from 4.96 to 1.91 after enhancement (smaller score means better quality) on IIITD-MOLF and 4.48 to 2.64 on IIIT-D MSLF database (See Table 3.9 (a)) which validates the improved clarity of ridges and valleys (thereby improving the quality score) in the enhanced fingerprints generated by the proposed model.

(2) Ridge Structure Preservation: In Fig. 3.8, we present some sample test cases with their ground-truth binarization and the output of the proposed algorithm. High match score and high SSIM value between the ground-truth binarized image and the output of the proposed method illustrate that the proposed algorithm preserves the ridge information of input latent fingerprint images including fingerprint class, orientation of ridges and minutiae, while enhancing them.

(3) Matching Latent Fingerprints to Multi-Sensor Fingerprints: In Fig. 3.10, we show CMC curves for matching performance achieved by Bozorth and MCC matcher on the enhanced image generated by the proposed model across two different galleries. We also compare the Rank-50 accuracy of the proposed model with the recently proposed latent fingerprint algorithm [17] (See Table 3.9 (b)). The magnitude of improvement obtained over raw images using the proposed algorithm is much more than the previous work (Rank-50 accuracy of 34.43% on



(a) (b)

Fig. 3.7: Evaluation of quality of fingerprint images using the NFIQ module of NBIS [5] for latent fingerprint images from (a) IIITD-MOLF and (b) IIITD-MSLF database.

DB1 and 30.50% on DB2 gallery using the proposed algorithm compared to 22.36% on DB1 and 19.50% on DB2 gallery by the previous work [17]). This demonstrates that the proposed algorithm performs better than [17] in improving ridge-valley contrast, removing background noise while preserving ridge details due to which improved feature extraction and thereby, improved matching performance is obtained.

(4) **Matching Multi-Surface Latent Fingerprints to gallery of live-scan fingerprints:** The rank-50 accuracy before and after enhancement on IIITD-MSLF database using Bozorth are 11.43% and 12.80% respectively. The CMC curve is shown in Fig. 3.11. The accuracy obtained on IIITD-MSLF database is lesser compared to the accuracy achieved on IIITD-MOLF. This is due to the complex background present in IIITD-MSLF database images. We observe that the intensity values of foreground and the background fingerprint regions have similar distribution in many images. This leads to spurious pattern generation by the proposed algorithm, which adversely affects the matching performance.

(5) **Significance of Latent Fingerprint Reconstruction Loss:** To demonstrate the significance of the reconstruction loss in the objective function, we train the proposed model with only adversarial loss (λ=0). We observe that the model becomes unstable and doesn't converge. In Fig. 3.12, we show the sample results obtained with only reconstruction loss. Therefore, we conclude that the reconstruction loss is essential to stabilize the proposed model.

(6) **Role of Hyper-parameters:** During the various experiments conducted in this chapter, we find that the that MCC is a better fingerprint matcher for latent fingerprints (See Table 3.4, Table 3.5, Table 3.6 and Table 3.7). We conclude our observations based on the results obtained using MCC matcher. By default, the hyper-parameters used are λ=10, batch size=2 and number of epochs=200.

   (i) **Weight Hyper-parameter (λ):** We observe that the Rank-50 achieved by the proposed model increases as the weight of enhanced fingerprint reconstruction loss is increased from 1 to 5 (As depicted in Table 3.7, Fig. 3.14(a), Fig. 3.14(b), Fig. 3.15(a) and Fig. 3.15(b)). However, on increasing the weight further, the performance starts degrading. Best Rank-50 accuracy of 34.43% across DB1 and 30.50% across DB2 gallery are achieved for λ=5. The quality of the other hand, improves while λ is increased from 1 to 10. On increasing λ further, the quality starts degrading (See Table 3.11 (b) and Fig. 3.16(a)). This suggests that the model is sensitive to the choice of weight parameters and a careful combination of adversarial loss and reconstruction loss is required to efficiently train the proposed model.

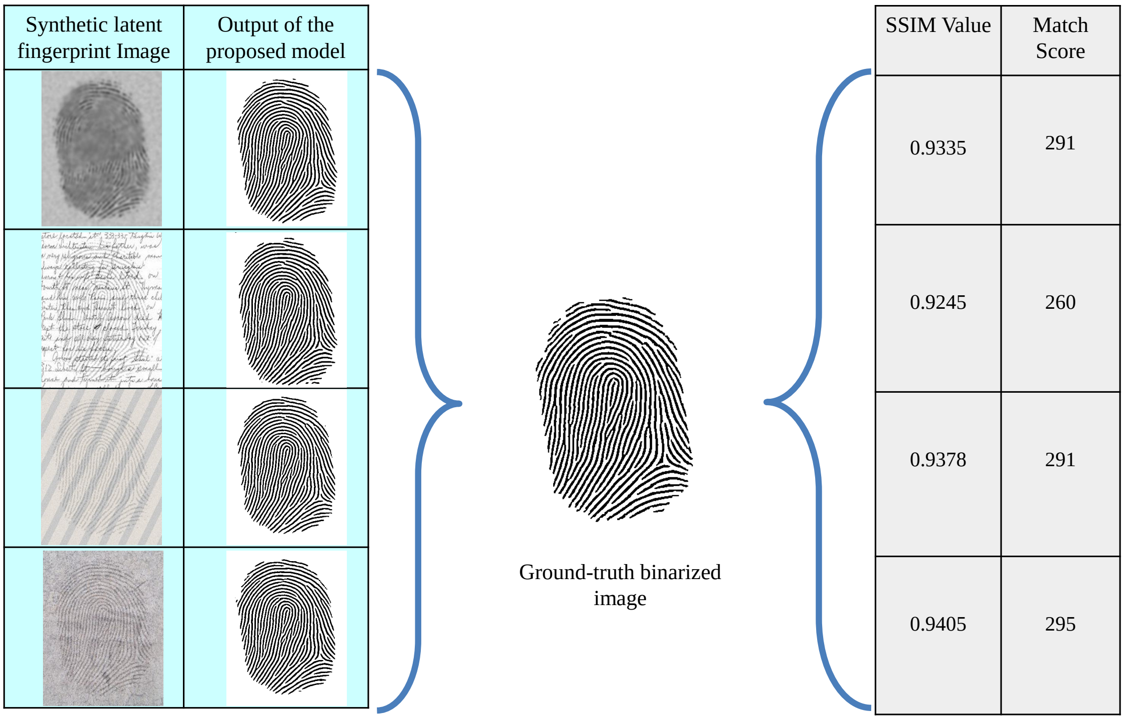
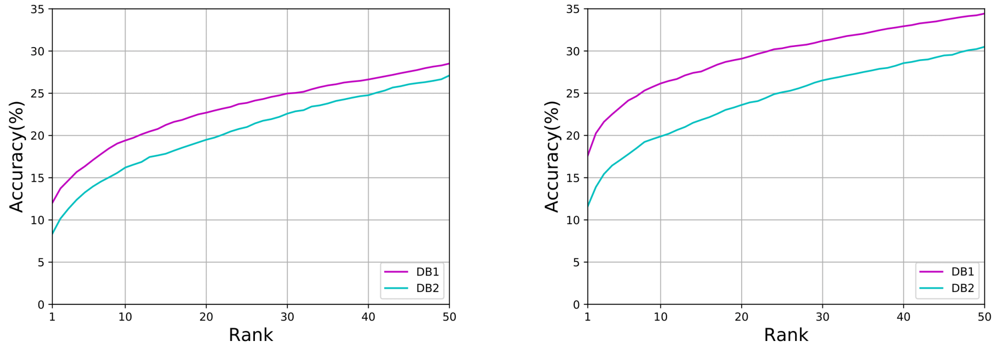


Fig. 3.8: Left side shows the enhanced fingerprint generated from the synthetic latent fingerprint, corresponding to the ground truth binarized image (shown in the middle). Right side shows the SSIM value and the matching score (obtained using Bozorth) for each enhanced image corresponding to the ground truth image.

(ii) Number of epochs: As shown in Table 3.4, Fig. 3.14(c), Fig. 3.14(d), Fig. 3.15(c) and Fig. 3.15(d), the rank-50 accuracy, initially improves with number of epochs till 60 epochs, after which it fluctuates and is approximately the same till 200 epochs. The performance of the model starts degrading after 200 epochs due to over-fitting. NFIQ score on the other hand, is improved initiallty till 150 epochs and then it fluctuates and no clear trend is found (See Table 3.11 (a)).

(iii) Batch size: We compare the rank-50 accuracy achieved by the model at batch size=2, 4 and 8 (See Table 3.6, Fig. 3.14(e), Fig. 3.14(f), Fig. 3.15(e) and Fig. 3.15(f)). As the batch size is increased, the number of parameter updates per epoch reduces which leads to faster training of the model. Batch size=2 turns out to be the best value of hyper-parameter batch size. The best performance at batch size=2 is attributed to more parameter updates and thus better training.

Better quality score is obtained for batch size=8 than batch size=2 and batch size=4, as can be seen in Table 3.11 (c) and Fig. 3.16(c). This is a counter-intuitive result as the enhanced fingerprint generated at batch size=8 has large missing regions compared to the fingerprints generated at batch size=2. Many of the images generated at batch size=2 have poor ridge smoothness compared to the image generated at batch size=8 which have very small reconstructed fingerprint area but better smoothness (See Fig. 3.13). Due to this, NFIQ gives a better score to images generated at batch size=8 than the images generated at batch size=2. Thus, we conclude that the anomaly in the quality score is due to the limitation of NFIQ. In reality, the quality of fingerprints generated at batch size=2 and batch size=4 is better than the quality of fingerprints generated at batch size=8 which is evident through the much higher Rank-50 accuracy achieved for batch size=2 and batch size=4 than batch size=8.



(a) (b)

Fig. 3.10: CMC curve for proposed algorithm's matching performance for the IIITD-MOLF DB1 gallery and DB2 gallery at λ=5, using (a) Bozorth (b) MCC.

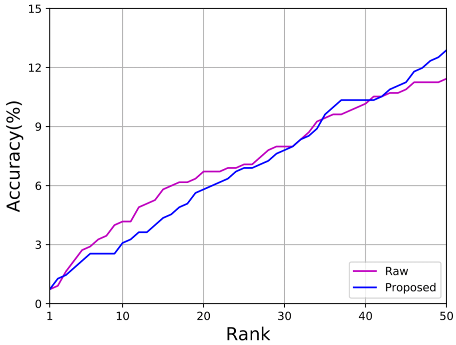


Fig. 3.11: CMC curve representing matching performance on IIITD-MSLF database, before and after enhancement by the proposed algorithm using Bozorth.

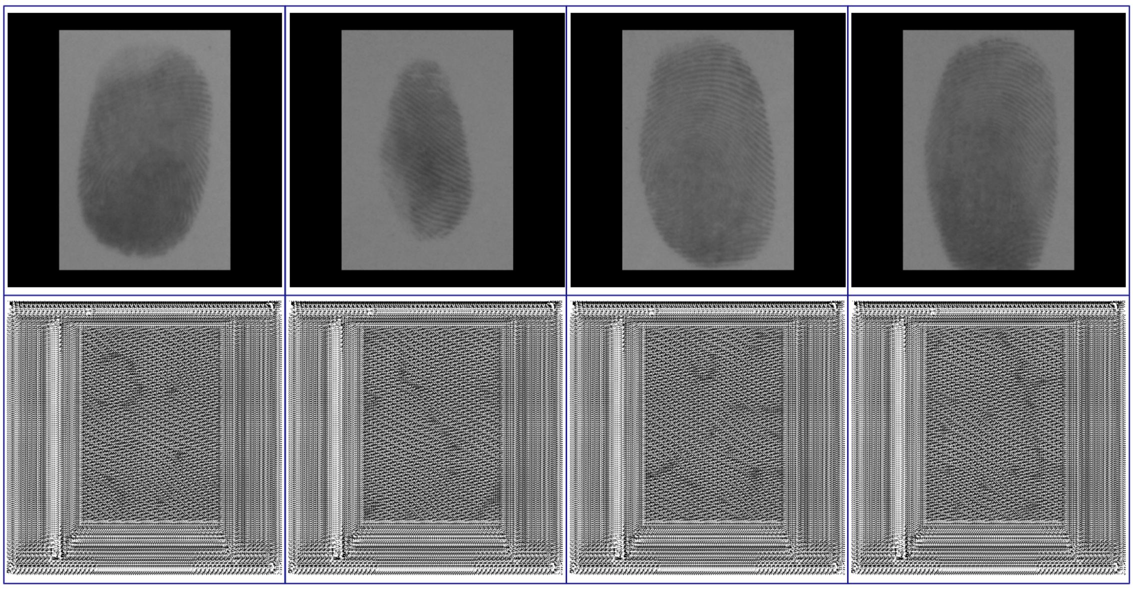


Fig. 3.12: Sample enhanced images obtained by the model when trained without latent fingerprint reconstruction loss.

(7) Effect of training the model with real NIST SD4 images: The rank-50 accuracy achieved by the images generated by the proposed model trained on both synthetic latent fingerprints

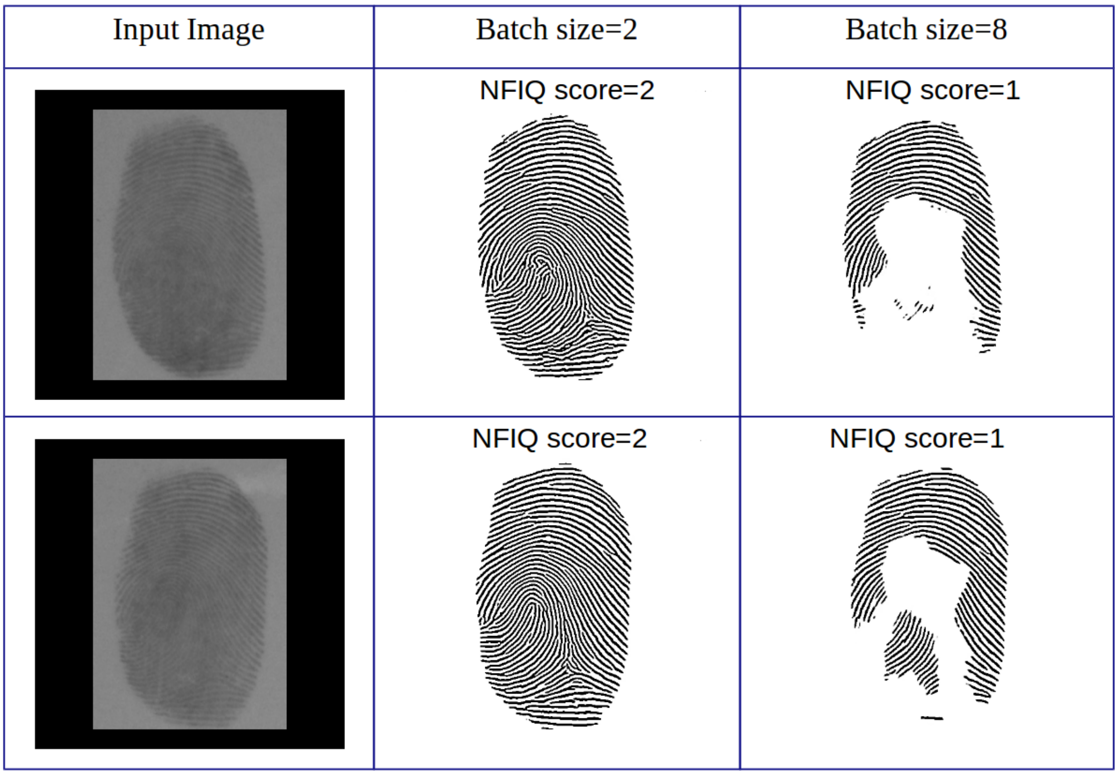


Fig. 3.13: Failure cases of NFIQ (lower score means better quality).

and real fingerprints from NIST SD4 is better compared to the model trained on only synthetic latent fingerprints, as shown in Fig. 3.17, Fig. 3.14(g), Fig. 3.14(h), Fig. 3.15(g) and Fig. 3.15(h)). Similar trend is seen in the NFIQ quality scores of the enhanced fingerprints obtained using the proposed model (See Table 3.11 (d) and Fig. 3.16(d)). The real fingerprints have practical cases of non-linear distortion and non-uniform ridge width which are also observed in latent fingerprints. Thus, the real inked fingerprints like those of NIST SD4 database help the model to learn to be invariant to such distortions.

Epoch	**DB1(Bozorth)**	**DB2(Bozorth)**	**DB1(MCC)**	**DB2(MCC)**
30	24.80	23.02	28.16	25.98
60	28.11	25.05	**33.61**	29.36
90	28.61	25.05	33.14	29.43
120	28.63	**26.75**	33.55	**30.14**
150	24.66	24.05	29.23	26.93
180	**28.77**	26.70	33.34	29.93
200	27.25	25.64	32.02	29.32
210	25.93	24.50	30.50	28.30
240	25.34	23.84	30.16	27.05
270	24.75	23.84	29.34	26.59

Table 3.4: Rank-50 accuracy obtained over different epochs on IIITD-MOLF latent fingerprints.

Training Data	DB1(Bozorth)	DB2(Bozorth)	DB1(MCC)	DB2(MCC)
Without SD4	**27.70**	**26.30**	30.43	29.2045
With SD4	27.25	25.64	**32.02**	**29.32**

Table 3.5: Rank-50 accuracy obtained on IIITD-MOLF latent fingerprints with and without adding SD4 images in training data.

Batch Size	DB1(Bozorth)	DB2(Bozorth)	DB1(MCC)	DB2(MCC)
2	27.25	**25.64**	**32.02**	**29.32**
4	**27.93**	26.61	30.45	26.659
8	18.03	17.28	15.41	15.41

Table 3.6: Rank-50 accuracy obtained IIITD-MOLF latent fingerprints for different batch size.

$\boldsymbol{\lambda}$	DB1(Bozorth)	DB2(Bozorth)	DB1(MCC)	DB2(MCC)
1	24.00	22.0	28.09	23.70
3	28.11	24.89	31.70	28.11
5	**28.52**	**27.11**	**34.43**	**30.50**
10	27.25	25.64	32.02	29.32
15	26.60	25.27	31.89	25.39
20	25.66	23.43	29.34	27.55

Table 3.7: Rank-50 accuracy obtained on IIITD-MOLF latent fingerprints over different values of λ.

Dataset	Enhancement Algorithm	NFIQ Score
IIITD-MOLF	Raw Image	4.96
IIITD-MOLF	Raw Image	**1.91**
IIITD-MSLF	Proposed	4.48
IIITD-MSLF	Proposed	**2.64**

(a)

Enhancement Algorithm	Rank-50 Accuracy (DB1)	Rank-50 Accuracy (DB2)
Raw Image	5.45	5.18
Svoboda et al. [17]	22.36	19.50
Proposed	**34.43**	**30.50**

(b)

Table 3.9: (a) Average NFIQ scores before and after enhancement by the proposed model on IIITD-MOLF and IIITD-MSLF databases. (b) Rank-50 obtained on IIITD-MSLF database before and after enhancement by the proposed model.

Epoch	NFIQ Score
30	2.07
60	2.03
90	2.00
120	1.86
150	1.82
180	1.84
200	1.83
210	1.83
240	**1.81**
270	1.83

(a)

$\boldsymbol{\lambda}$	NFIQ Score
1	2.06
3	1.99
5	1.91
10	**1.83**
15	1.87
20	1.91

(b)

Batch Size	NFIQ Score
2	1.83
4	1.83
8	**1.18**

(c)

Training Data	NFIQ Score
Without SD4	2.33
With SD4	**1.83**

(d)

Table 3.11: Average NFIQ scores of the enhanced fingerprints obtained for IIITD-MOLF database using the proposed algorithm over (a)different epochs (b) different values of λ (c) with and without adding SD4 images in training data (d) different values of batch size.

6 Challenges Observed

While conducting different experiments, it has been found that the proposed algorithm improves matching performance. However, we observe some cases where the proposed algorithm does not generate good results. Analysis of these cases is given in the following points:

(1) We find that many of the input latent fingerprint images have low ridge information. However, even for such images, the proposed algorithm enhances those regions of the latent fingerprint image which have some ridge information (See left-most column of Fig. 3.20). We understand that it will be difficult for any enhancement algorithm to enhance such cases while preserving the minutiae details.

(2) While matching latent fingerprint images, ROI is manually marked by forensic experts and the enhancement is performed only on ROI. However, the proposed algorithm automatically segments the foreground and background and then enhances the foreground fingerprint. Due to this, it sometimes misinterprets the background as foreground (See last three columns from right in Fig. 3.20) when the intensity distributions of background and foreground fingerprint are similar.

(3) We found that the NFIQ is not a robust fingerprint quality assessment metric (See Fig. 3.13). NFIQ2 is a more effective metric than NFIQ, however, it fails to process latent fingerprints. Thus, there is a need to introduce a more robust latent fingerprint quality assessment tool in the public domain to facilitate improved research in latent fingerprint matching.

(4) The proposed model is observed to be highly sensitive to the choice of hyper-parameters and does not perform well if the training hyper-parameters are not carefully chosen.

(5) The loss function is carefully designed for enhancement of latent fingerprints. Any change in the loss function can lead to unstable training of the model (as observed while training the model without enhanced reconstruction loss, as shown in Fig. 3.12).

7 Conclusion

Motivated by the successful applications of GANs in various image processing applications, we formulate latent fingerprint enhancement like an image-to-image translation problem. The proposed model is trained using an enhancer and a discriminator network in an adversarial fashion. The model is trained using both synthetic and real fingerprints due to which it is robust to distortions observed in latent fingerprints. Moreover, the proposed model does not need a real latent fingerprint database to train the network. Two latent fingerprints databases available in the public domain are used for evaluating the proposed enhancement model. A detailed analysis of performance of model over hyper-parameters such as lambda, number of epochs, batch size is done. We also give insights on the role of real inked prints while training the model and the significance of reconstruction loss in the objective function.

We analyse the failure cases and some cases have been encountered when the ridge information is insufficient and the proposed algorithm generates spurious features. To address these limitations, the possibility of recoverability needs to be explored such that the algorithm can decide which portions of fingerprints can be reconstructed and which ones cannot. Training with a larger database with more variations in texture and background can help to achieve even better performance on IIITD-MSLF database. The proposed algorithm can also be utilized in challenging scenarios like latent to latent fingerprint matching.

8 Acknowledgement

The authors thank IIT Delhi HPC facility for computational resources. The authors express their gratitude to Mayank Vatsa and Richa Singh from IIIT Delhi for their inputs. The authors also thank Himanshu Gandhi and Vijay Kumar for helpful discussions.

References

1. A. Sankaran, A. Agarwal, R. Keshari, S. Ghosh, A. Sharma, M. Vatsa, and R. Singh. Latent fingerprint from multiple surfaces: Database and quality analysis. In *IEEE 7th International Conference on Biometrics Theory, Applications and Systems*, 2015.
2. D. Nguyen, K. Cao, and A. K. Jain. Robust minutiae extractor: Integrating deep networks and fingerprint domain knowledge. In *2018 International Conference on Biometrics (ICB)*, 2018.
3. B. T. Ulery, R. A. Hicklin, J. Buscaglia, and M. A. Roberts. Repeatability and reproducibility of decisions by latent fingerprint examiners. *PloS one*, 7:e32800, 2012.
4. B. T. Ulery, R. A. Hicklin, J. Buscaglia, and M. A. Roberts. Accuracy and reliability of forensic latent fingerprint decisions. *Proceedings of the National Academy of Sciences*, 108, 2011.
5. NBIS-NIST biometric image software. `http://biometrics.idealtest.org/`.
6. S. Yoon, J. Feng, and A. K. Jain. On latent fingerprint enhancement. In *Biometric Technology for Human Identification VII*, volume 7667, page 766707, 2010.
7. Neurotechnology inc., verifinger. `http://www.neurotechnology.com`.
8. S. Yoon, J. Feng, and A. K. Jain. Latent fingerprint enhancement via robust orientation field estimation. In *2011 international joint conference on biometrics (IJCB)*, pages 1–8, 2011.
9. J. Feng, J. Zhou, and A. K. Jain. Orientation field estimation for latent fingerprint enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(4):925–940, 2013.
10. X. Yang, J. Feng, and J. Zhou. Localized dictionaries based orientation field estimation for latent fingerprints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(5):955–969, 2014.
11. C. Chen, J. Feng, and J. Zhou. Multi-scale dictionaries based fingerprint orientation field estimation. In *International Conference on Biometrics*, pages 1–8, 2016.
12. K. Cao and A. K. Jain. Latent orientation field estimation via convolutional neural network. In *International Conference on Biometrics*, pages 349–356, 2015.
13. S. Liu, M. Liu, and Z. Yang. Sparse coding based orientation estimation for latent fingerprints. *Pattern Recognition*, pages 164–176, 2017.
14. Watcharapong Chaidee, Kittipol Horapong, and Vutipong Areekul. Filter design based on spectral dictionary for latent fingerprint pre-enhancement. In *2018 International Conference on Biometrics (ICB)*, pages 23–30, 2018.
15. Zhenshen Qu, Junyu Liu, Yang Liu, Qiuyu Guan, Chunyu Yang, and Yuxin Zhang. Orienet: A regression system for latent fingerprint orientation field extraction. In *International Conference on Artificial Neural Networks*, pages 436–446, 2018.
16. J. Li, J. Feng, and C.C. J. Kuo. Deep convolutional neural network for latent fingerprint enhancement. *Signal Processing: Image Communication*, 60:52–63, 2018.
17. J. Svoboda, F. Monti, and M. M. Bronstein. Generative convolutional networks for latent fingerprint reconstruction. In *IEEE International Joint Conference on Biometrics*, pages 429–436, 2017.
18. I. Joshi, A. Anand, M. Vatsa, R. Singh, S. D. Roy, and P. Kalra. Latent fingerprint enhancement using generative adversarial networks. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 895–903, 2019.
19. I. Goodfellow, A. Pouget, Jean, M. Mirza, B. Xu, D. F. Warde, Sherjil Ozair, Aaron Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27*, pages 2672–2680. 2014.
20. M. Mirza and S. Osindero. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014.
21. P. Isola, J. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.
22. A.H.Ansari. Generation and storage of large synthetic fingerprint database. Technical report, Indian Institute of Science, Bangalore, 2011.
23. Raffaele Cappelli, D Maio, and D Maltoni. Sfinge: an approach to synthetic fingerprint generation. In *International Workshop on Biometric Technologies (BT2004)*, pages 147–154, 2004.
24. NIST special database 4. `http://www.nist.gov/srd/nistsd4.cfm`.
25. E. Tabassi. NFIQ 2.0: NIST fingerprint image quality. *NIS-TIR 8034*, 2016.
26. A. Sankaran, M. Vatsa, and R. Singh. Multisensor optical and latent fingerprint database. *IEEE Access*, 3:653–665, 2015.
27. Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13, 2004.
28. R. Cappelli, M. Ferrara, and D. 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.
29. R. Cappelli, M. Ferrara, and D. Maltoni. Fingerprint indexing based on minutia cylinder-code. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):1051–1057, 2011.
30. M. Ferrara, D. Maltoni, and R. Cappelli. Noninvertible minutia cylinder-code representation. *IEEE Transactions on Information Forensics and Security*, 7(6):1727–1737, 2012.

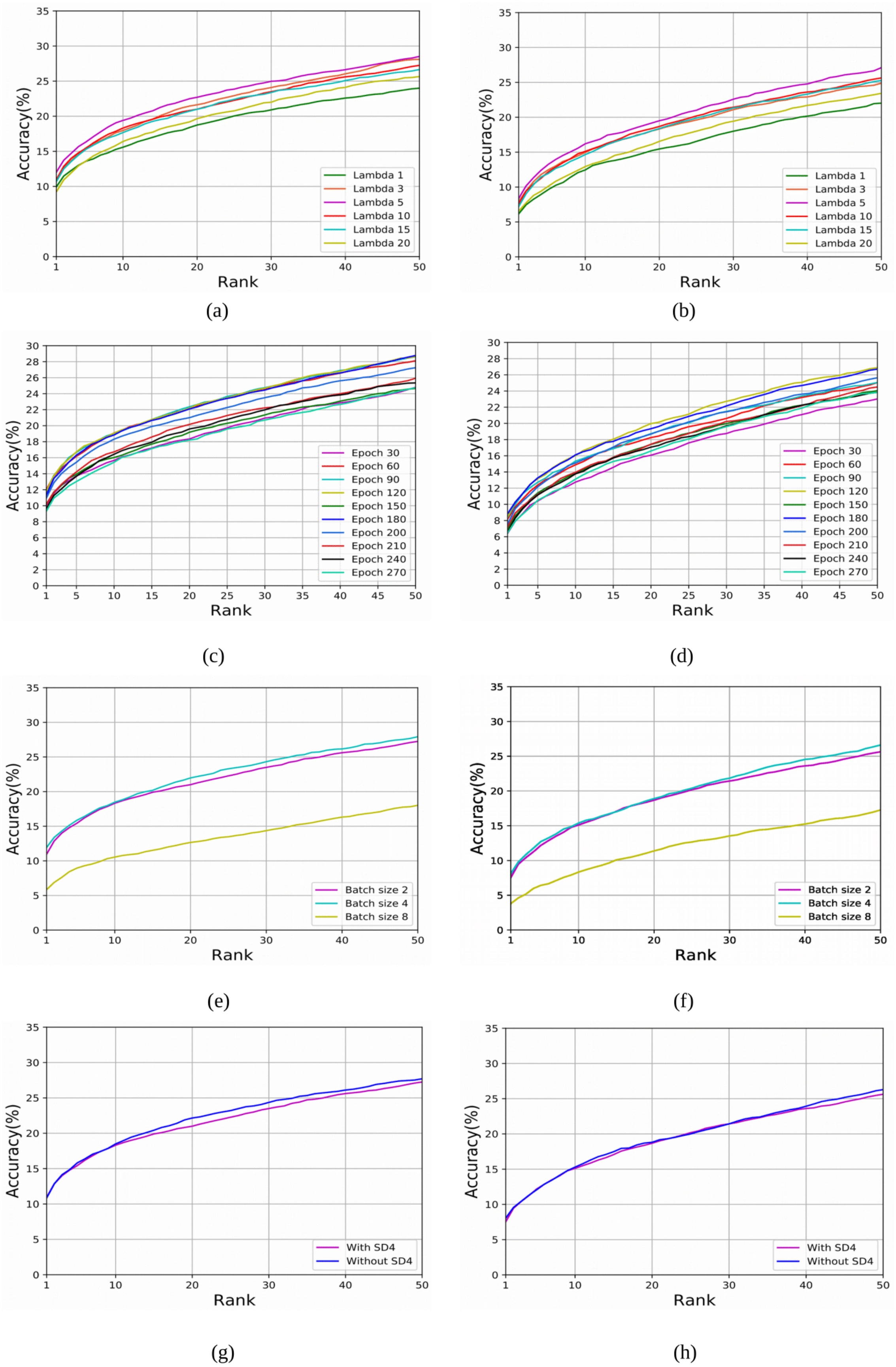


Fig. 3.14: CMC curve for matching the the proposed algorithm using Bozorth on the IIITD-MOLF DB1 and DB2 galleries across different training settings.

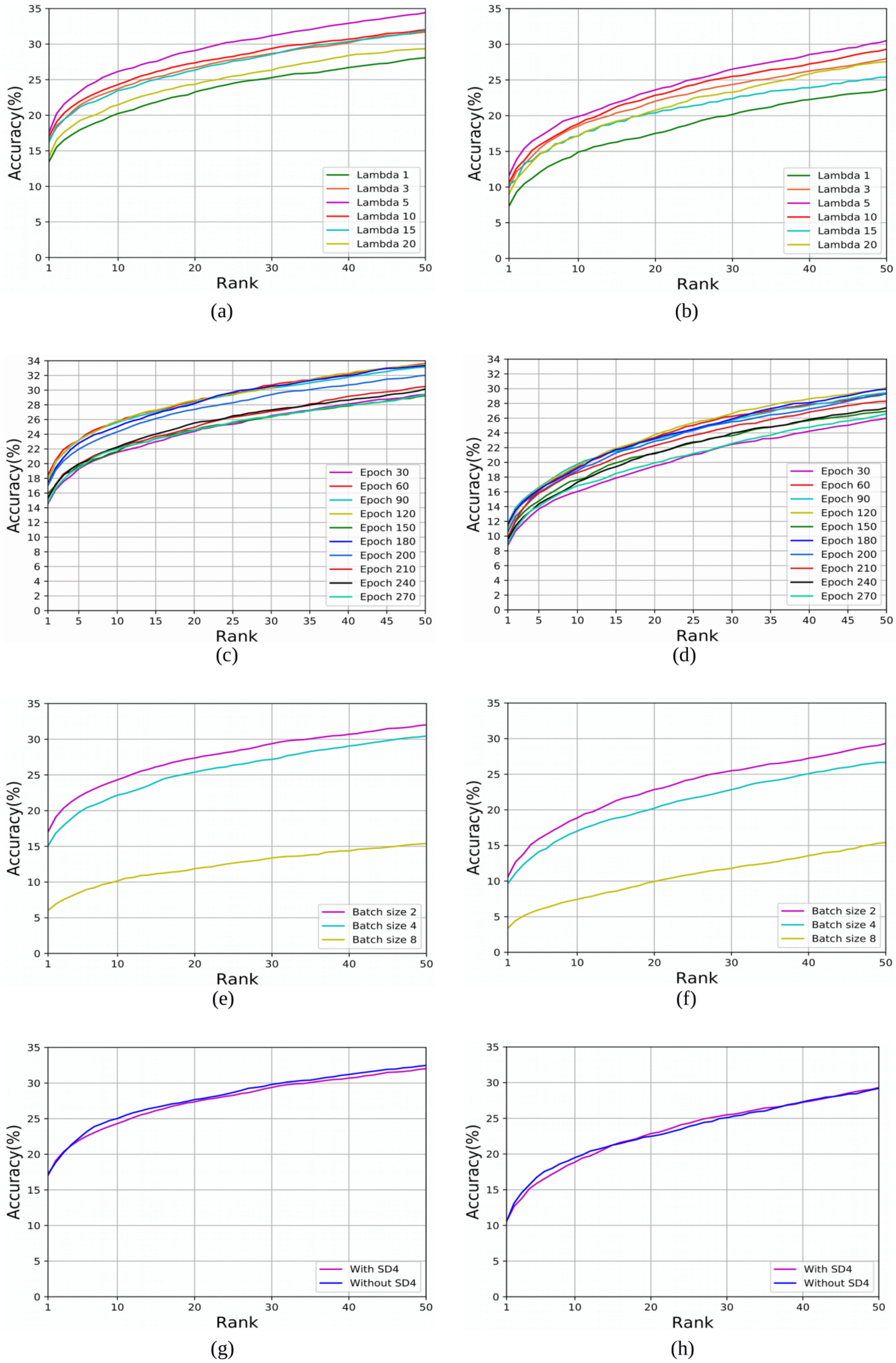
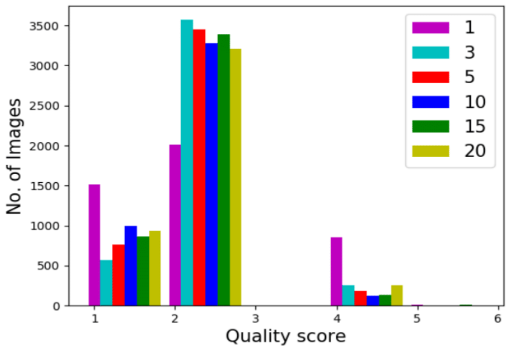
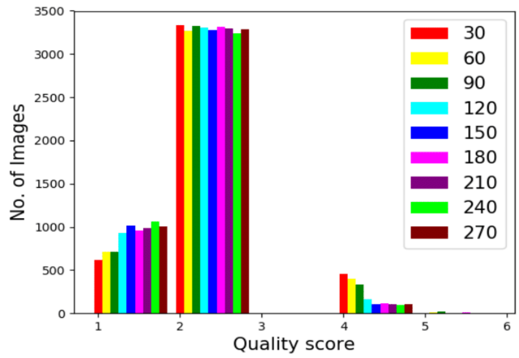


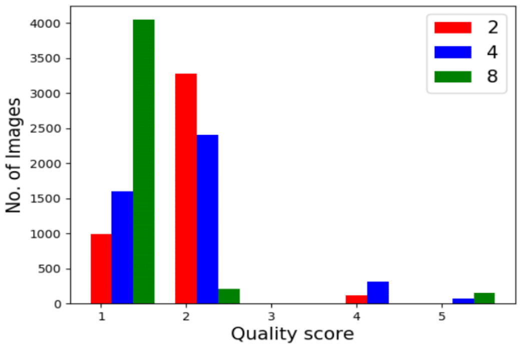
Fig. 3.15: CMC curve for matching the the proposed algorithm using MCC on the IIITD-MOLF DB1 and DB2 galleries across different training settings.



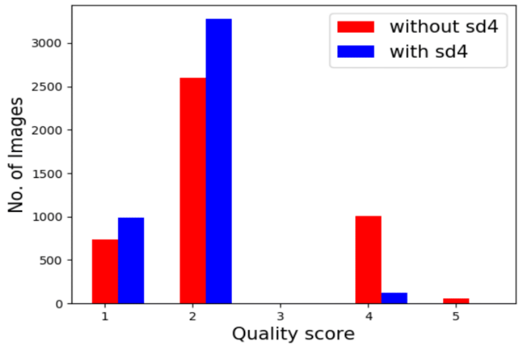
(a)



(b)



(c)



(d)

Fig. 3.16: NFIQ score distribution of the enhanced images produced by the proposed algorithm across different training settings.

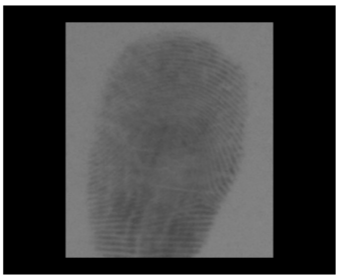
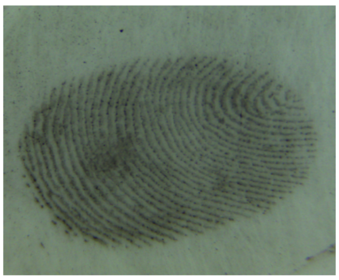
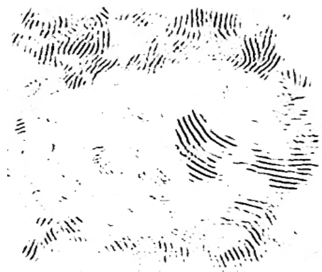
Input latent fingerprint image	Output when the model is trained without SD4 images	Output when the model is trained with SD4 images
		
		

Fig. 3.17: Sample results obtained by the model when trained with and without NIST SD4 images in the training dataset.

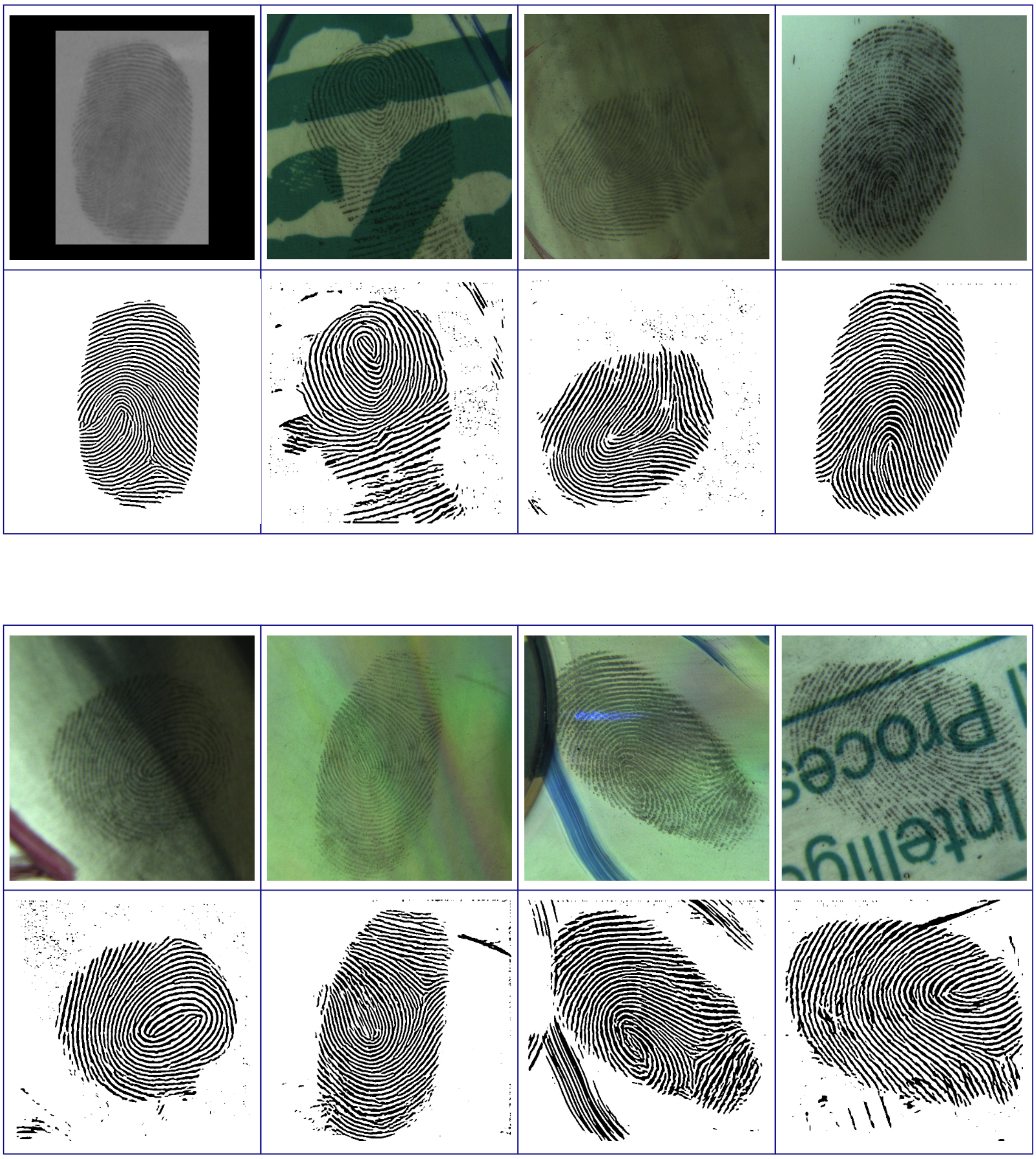


Fig. 3.19: Samples of successful enhancement of latent fingerprints by the proposed model.

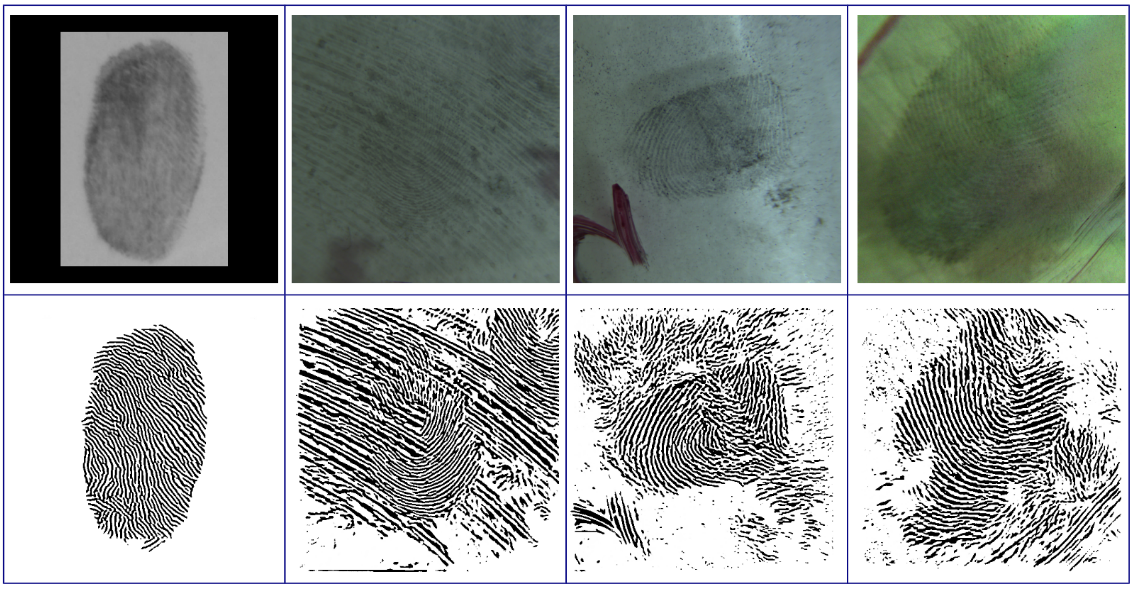


Fig. 3.20: Some challenging cases for the proposed model.