# MinNet: Minutia Patch Embedding Network for Automated Latent Fingerprint Recognition

Halil İbrahim Öztürk Berkay Selbes Yusuf Artan
HAVELSAN A.S.
{hozturk, bselbes, yartan}@havelsan.com.tr

## Abstract

*In this study, we proposed a novel minutia patch embedding network (MinNet) model for latent fingerprint recognition task. Embedding vectors generated for a fixed-size patch extracted around a minutia are used in the local similarity assignment algorithm to produce a global similarity match score. Unlike earlier minutia embedding models that aim to discriminate between latent image and sensor image minutia pair embeddings using $\ell_2$ distance between the embedding vectors in the training process, MinNet model jointly optimizes the spatial and angular distribution of neighboring minutiae and ridge flows of the patches. Even though the proposed model is trained using weakly labeled training data, it produces state-of-the-art results thanks to it ability to generate discriminative embeddings. Proposed method has been evaluated on several public and private datasets and compared to popular latent fingerprint recognition methods presented in earlier studies. Our proposed method significantly outperforms existing methods on all three databases utilized in our study.*

## 1. Introduction

Biometric security systems are used in the identification or verification of an individual using his/her biological, morphological or behavioral characteristics [4,9,23]. While biological biometrics uses the genetic and molecular characteristics of individuals (e.g. DNA), morphological biometry is related to the person's physical characteristics such as fingerprint, palm print, iris and face, to name a few. Behavioral biometrics systems, on the other hand, utilize behavioral characteristics (person's gait, voice etc.) specific to each person. For the past several decades, fingerprint recognition systems have been the most widely adopted biometrics in the world due to the ease of data acquisition process and the cost effectiveness of the overall system. Some of the popular application domains of the fingerprint recognition systems include law enforcement & forensic agencies, border control, social security, and national ID programs.

Fingerprint recognition task can be described as the retrieval of mate of a query fingerprint within a database of reference prints. Fingerprint images can be broadly classified into 3 categories; rolled, slap/plain and latent [4, 9]. While rolled fingerprints are typically larger in size and contain larger number of minutia, plain fingerprints are often less distorted and have clear ridges. Since rolled and slap images are in general collected using digital fingerprint scanners under the supervision of an enforcement officer, they usually have high image quality. However, latent fingerprint scans are collected from the crime scene through a variety of means; from photographing the scene to more complex dusting and chemical processing [9]. Therefore, latent images generally have low image quality and contain unclear ridge structure, deformations and artifacts that would render the overall fingerprint recognition task more difficult.

Many early research on fingerprint recognition subject treated fingerprint matching as a 2D minutiae point cloud matching task aimed at resolving a global alignment problem leading to an optimal minutia pairings [19, 21]. However, these global matching methods were computationally expensive and lacked robustness against distortions and missing minutiae. To remedy these problems, local descriptor based fingerprint matching methods were developed utilizing only spatial coordinates and angle information associated with each minutiae in their analysis [3, 4, 16]. In these approaches, local descriptors extracted within a fixed neighborhood of minutiae are used to measure similarity of images corresponding to rolled and slap fingerprint impressions. Even though, these methods were efficient and worked well for datasets containing sensor images, their performance was considerably lower for latent fingerprint images. Several studies have proposed methods to improve local descriptor based method's performance in latent fingerprint recognition using features beyond minutiae such as ridge counts, orientation maps [18, 27]. More recently, [17] proposed a deformation tolerant extension of local descriptor algorithms that can handle non-linear distortion present

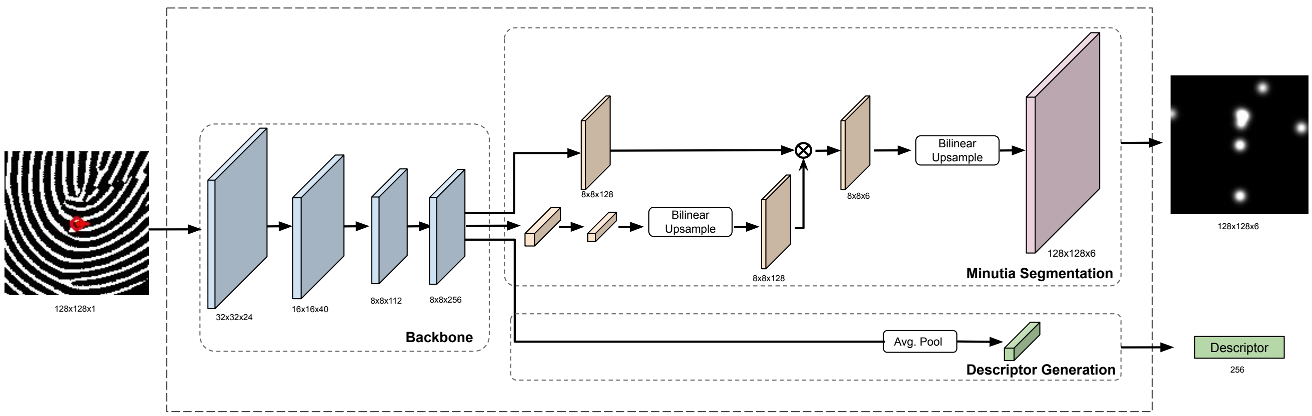



Figure 1. This figure illustrates an overview of the proposed minutia patch embedding network (MinNet) model for a single patch (shown on the left) extracted around a minutiae (shown in red dot). MobileNet v3 large based backbone model extracts features that are used to generate embedding vectors that would also preserve spatial and angular distributions of neighboring minutiae (compressed minutiae map is shown in the right side of the figure). While the Minutia Segmentation branch aims to reconstruct neighboring minutiae's positional and angular information as described in section 2.1, Descriptor Generation branch creates a patch embedding vector to represent ridge flow information of the patch.

in latent images. This algorithm performs clustering of matching minutiae in an iterative process, in which algorithm finds multiple overlapping clusters of matching minutiae and the best clusters are merged to deal with the non-linear distortion of the fingerprints.

For the past several years, we have seen a surge in the automatic latent fingerprint recognition research using deep learning techniques [1, 2, 7, 24]. Many methods have been proposed towards subtasks of latent fingerprint recognition such as latent image minutia extraction, latent image quality assessment, orientation field estimation and minutia descriptor generation tasks using various convolutional neural network (CNN) based architectures [7, 24]. For instance, in a recent study [26], authors utilized deep learning techniques for joint estimation of pose and singular points in fingerprints. Another study [7] considers latent image quality assessment task using deep learning techniques. [1] utilized CNNs to calculate ridge flow, extract minutia descriptors and minutiae templates of sensor and latent images, and used these descriptors and templates to produce match scores. In another study, [2] proposed and end-to-end latent fingerprint recognition system, which performs automatic segmentation of latent print region, pre-processing, features extraction and matching operations, sequentially. Tang *et al.* [24] proposed a unified framework named FingerNet for minutiae extraction task along with orientation field estimation, latent print segmentation, and latent print enhancement. These earlier works show the great potential of utilizing deep neural networks in the latent fingerprint recognition task.

This paper presents a novel method for latent fingerprint recognition task using a deep learning based local descriptor generation model, MinNet. Unlike earlier approaches presented in [1, 2], proposed local descriptor model learn to represent local ridge flow and spatial/angular minutiae distribution around a fixed neighborhood of a minutiae simultaneously through a novel cost function. Earlier local descriptor generator methods were designed to discriminate between fixed-size image patches extracted around minutiae using an $\ell_2$ difference between latent and sensor embeddings for these patches.Proposed method has been evaluated on several datasets and compared to earlier latent fingerprint recognition works presented in recent years [2, 3, 17].

The contributions of the paper are as follows:

- Proposed a novel local descriptor generation model that not only captures the local ridge flow information around each minutiae but it also captures the spatial and angular distribution of neighboring minutiae.
- Proposed model is able to work very well for both latent fingerprint and sensor fingerprint recognition tasks, and it produces state-of-the-art results on both sensor-to-sensor and latent-to-sensor image matches as shown in our experiments.
- Weakly supervised minutiae pairings from different impressions of the same finger are utilized to train a model without any labor intensive minutiae labeling process.
- Two novel latent datasets are created to evaluate the proposed methods' latent fingerprint recognition performance.

We also want to emphasize that the proposed MinNet local descriptor generator model is trained using only rolled

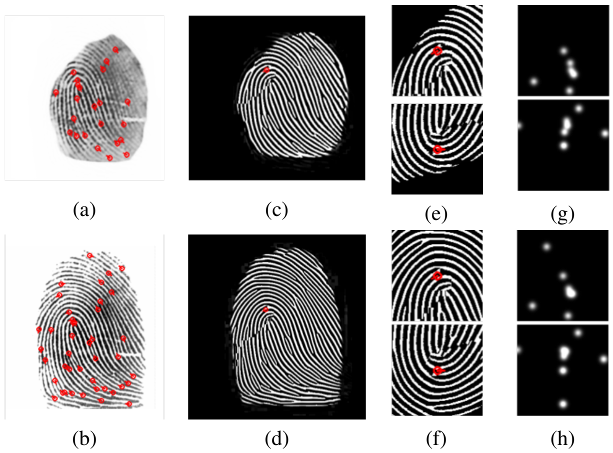



Figure 2. a) Latent Fingerprint with overlaid minutiae in red, b) Slap Fingerprint with overlaid minutiae in red, c-d) Enhancement Maps with a single minutia in red, d-e) Local minutia patch with and without rotation, g-h) Minutia segmentation maps.

and slap fingerprint images and tested on latent fingerprint recognition task.

This paper is organized as follows. In section 2, we explain the details of the proposed method. To evaluate the effectiveness of the proposed technique on real-world datasets, we conduct experimental analysis in section 3. Finally, section 4 presents the conclusions of this study.

## 2. Methods

### 2.1. Descriptor Generation

**Local Patches:** Local ridge flows, spatial and angular distributions of neighboring minutiae provide the distinctive information for a fixed size patch extracted around a minutia of interest. In order to match minutiae in latent and sensor fingerprints, encoding the information around the minutia to descriptors is essential. We generate the minutia descriptor with our network MinNet from a local patch cropped around of the minutia as shown in Fig. 2e and 2f. However, before cropping the patch, we enhance fingerprint image to eliminate noise and to enhance local ridge flow. After the enhancement operation, we segment fingerprint region from the image to remove the unnecessary background. To this end, we extract segmentation masks, enhancement maps and minutiae using FingerNet [25] algorithm that is trained for latent fingerprint minutiae extraction task. This allows us to bring latent fingerprints and sensor fingerprints into the same domain as illustrated in Fig. 2c and 2d. Last step of minutia patch generation is rotating the patch with minutia angle (counterclockwise) so that the minutia angle aligns with horizontal +x axis, which is needed to make descriptors rotation invariant.

**Label Generation:** We train the proposed MinNet model using a private dataset composed of matching rolled and slap fingerprint images. During the training stage, matching minutia pairs from rolled and slap fingerprints of the same finger are used to extract patches around the minutiae. Since the annotation of matching minutia pairs is a labor intensive task, we generated minutia pairs using our own implementation of minutia cylinder coding (MCC) algorithm [3]. In order to eliminate noisy ground truth pairs, we utilized only images pairs with high global matching score (note that this is MCC matching score) and selected top 8 minutia pairs that have the highest local similarity scores in the MCC algorithm.

**Descriptor Generator:** In the training stage, proposed MinNet model aims to generate similar embedding vectors for the patches of the same minutia from different fingerprint impressions while increasing the dissimilarity of embedding vectors corresponding to non-matching minutia patches. To achieve this objective, we utilized additive angular margin loss (AAM) [5] in our model training process as discussed in the next paragraphs.

$$L_A = -\frac{1}{N}\sum_{i=1}^{N} \frac{e^{(s\times cos(\theta_{(i,i)}+m))}}{e^{(s\times cos(\theta_{(i,i)}+m)} + \sum_{j=1,j\neq y_i}^{n} e^{s\times cos(\theta_{(i,j)})}} \tag{1}$$

Additive angular margin loss is built upon cross entropy loss and softmax operations as shown in Eq. 1. This loss function requires a linear layer at the end of the backbone as shown in the lower branch of Fig. 1. That linear layer contains weight vectors $W_i$ for each class. In training stage, degree between a class vector $x_i$ and class weight $W_i$ decreases, while the degree between different class vector $x_i$ and class weight $W_j$ increases to a margin $m$ at least. Since the linear layer is not used after training, it is removed from the model. We set the margin $m$ parameter of the additive margin loss of Eq. 1 as 28.6 degree, and scale parameter $s$ is chosen as 16 in the training stage.

**Minutia Map:** Relative positions and angles of neighbouring minutiae with respect to a minutia of interest provides discriminative information for matching local minutia patches. Since, this knowledge provides better matching performance, we encode positions and angles of the minutiae within the local patches. Instead of passing *apriori* derived minutiae neighbourhood information directly into the model, our model learns to reconstruct the neighboring minutiae from the generated descriptor. In this way, MinNet model gains the ability to encode neighbour minutiae's positional and angular information explicitly.

Reconstruction of neighboring minutiae for a patch is performed with Minutia Segmentation branch as illustrated in 1. Segmentation branch of MinNet generates segmentation map by using the feature map produced by the backbone. Descriptor generation branch performs global average pooling operation over the same feature map to obtain minutia descriptor as in Fig. 1. Since any weight parameter is not used in the pooling operation, the descriptor also contains positional and angular information of neighboring minutiae that is required by minutia segmentation branch.

In the training stage, we utilize a multi-channel minutia segmentation maps $M^{wxhx6}$ as target of minutia segmentation branch. These minutia segmentation maps encode positional and angular information of the minutiae as in [6]. Each minutia is represented with Gaussian distribution with variance $\sigma^2$ around the minutia center $(x, y)$. In order to take account angle of the minutia, the distribution is weighted as in Eq. 3.

$$W_{x,y,k} = |\theta - \frac{2k\pi}{6}| \quad (2)$$

$$M_{(x,y,k),(i,j)} = \exp\left(-\frac{(x-i)^2 + (y-j)^2}{\sigma^2}\right) * W_{x,y,k} \quad (3)$$

Equation 3 yields a minutia map value at position $(i, j)$ of $k^{th}$ channel for the minutia at $(x, y)$ with $\theta$ degree. $\sigma$ value is chosen 5 as in [6].

**Final Loss:** Mean squared error (MSE) is used as the loss function for Minutia Segmentation map while AAM loss is used for Descriptor Generation branch of the proposed MinNet model. Therefore, when combining Minutia Segmentation and Descriptor Generation branches, we used a weighted combination of AAM loss and MSE loss as shown in Eq. 4 to train the model. We control the contribution of the losses with $\lambda_1$ and $\lambda_2$ values. In our experiments, we set $\lambda_1$ and $\lambda_2$ value to 1 and 64, respectively.

$$L = \lambda_1 L_A + \lambda_2 L_{MSE} \quad (4)$$

**Data Augmentation:** To further increase matching performance of model descriptors, we increase intra-class variance of training minutia patch pairs by using augmentation techniques. The augmentations are applied randomly with %25 probability while the rest of the enhanced minutia patches are not augmented with any augmentation technique. Applied augmentations are: (i) **rotation:** After rotating the enhanced minutia patches with angle of the minutiae, we rotate the patches with randomly selected degree within [-10, 10] range. (ii) **scaling:** We scale the patch with randomly selected ratio from (0.8, 0.9, 1.1, 1.2) ratio values. We apply the augmentation before cropping the patch from the enhanced fingerprint to avoid information loss in edges of the patches. We apply same augmentations to the segmentation maps of the augmented patches.

**Implementation Details:** We utilized MobileNet v3 [10] large version as backbone in our proposed MinNet model. Rotated and cropped local patches are normalized with instance normalization before passing to the network. We used Adam [11] optimizer with batch size of 512. We trained the model for 200 epochs using a learning rate of 1e-3.

### 2.2. Matching Algorithm

Once the embedding vectors are generated for each minutia of latent and sensor fingerprints, we measure the local similarity between latent minutia embedding $(v_i)$ and sensor minutia embedding $(v_j)$ using cosine similarity measure shown in Eq. 5.

$$s(v_i, v_j) = \frac{v_i^T \cdot v_j}{\|v_i\| \|v_j\|} \quad (5)$$

Therefore, for a given sensor and latent minutia embedding templates $A = \{a_1, a_2, \dots, a_M\}$ and $B = \{b_1, b_2, \dots, b_N\}$, respectively;

• $s(a, b)$ denote the local similarity between minutia $a \in A$ and $b \in B$, with $s(\cdot) : A \times B \rightarrow [-1, 1]$,

• $\Gamma \in [0, 1]^{n_A \times n_B}$ denote the similarity matrix corresponding to embedding templates $A$ and $B$ containing the local similarity between minutia embeddings with $\Gamma[r, c] = s(a_r, b_c)$.

When comparing two minutiae embedding templates (corresponding to latent image and sensor image), a global score value denoting the overall similarity needs to be obtained from these local similarities. In this study, we utilized local similarity assignment (LSA) algorithm to generate global similarity score value. Hungarian algorithm [12] is used to solve the linear assignment problem on matrix $\Gamma$ to find the set of $n_P$ pairs $P = (r_i, c_i)$ that would maximize global score without considering the same minutiae more than once. Global score, also known as match score, is computed as in Eq. 6. In our experiments, we set $n_P$ parameter equal to $min(12, min(N, M))$, where $N$ and $M$ corresponds to template size of $A$ and $B$, respectively. Note that in our matching algorithm we intentionally not added a global relationship checking approach for paired minutiae such as Local Similarity Sort with Relaxation (LSS-R) presented in [3]. Due to distortions present in latent images, compatability among minutiae relationships tends to low, hence, degrades the performance of the local similarity obtained using minutiae embedding vectors.

$$\Gamma(A, B) = \frac{\sum_{(r,c) \in P} \Gamma[r, c]}{n_P} \quad (6)$$

## 3. Experiments

In this section, we first describe the datasets used in the training and test stages of the proposed algorithm. Next, we report the performance results of the proposed method on test datasets and compare it with the performance of the existing methods on these datasets.

### 3.1. Datasets

In this study, several public and private fingerprint databases are utilized in the training and test stages of the proposed method. Since NIST SD27 [8] latent test set is not accessible, we utilized 2 new test databases that are generated by applying a CycleGAN based latent emulation model on selected images from 2 public datasets (FVC 2000-2004 Dataset [13–15] and Tsinghua Distorted Fingerprint Dataset [22]) in addition to a large private database (EGM Test Dataset) consisting of 5560 latent and rolled image pairs. While these new databases are significantly smaller in size compared to EGM Test dataset, they are comparable in size to popular databases such as NIST SD27 [8] and IIIT-D Latent Fingerprint dataset [20]. Table 1 presents an overview of the datasets used in this study and in earlier studies. These emulated datasets will be made publicly available in the link[1].

| | No. of Latents | Type |
|---|---|---|
| NIST SD27 [8] | 258 | Latent-Rolled |
| IIIT-D Latent [20] | 1046 | Latent-Slap |
| EGM Test Dataset | 5560 | Latent-Rolled |
| FVC Latent | 316 | Latent-Slap |
| Tsinghua Latent | 168 | Latent-Latent |

Table 1. An overview of the datasets used in this study and other private datasets used in the literature. Note that all the datasets listed above has 1-to-1 pairing images.

(a) (b) (c) (d)

Figure 3. Pairs of rolled (left) and latent (right) images from EGM dataset; (a)-(b) and (c)-(d)

**Training Dataset (Private):** This is a private dataset obtained from General Directorate of Citizenship Affairs (NVI) in Ankara, Turkey. Dataset consists of 10-print images belonging to 4716 individuals, and contains 34814 distinct rolled and slap fingerprint images of the same finger collected with digital optical scanners. We want to emphasize that proposed methods' training process utilized only images from this sensor dataset, hence, no latent fingerprint is used in the training stage of the MinNet model. As mentioned in section 2, a custom implementation of MCC algorithm is used to determine the matching minutiae pairs (with only 8 minutia pairs with high local similarity scores in MCC) for rolled and slap impressions.

**EGM Test Dataset (Private):** This is a private dataset obtained from General Directorate of Police (EGM) in Ankara, Turkey. Dataset consists of 5560 latent fingerprint images along with their corresponding rolled image for the same finger. These rolled images are collected using inkpad and optical scanners. Fig. 3 presents sample latent-rolled image pairs from this dataset.

**FVC-Latent Test Dataset (Public):** FVC is a multi-dataset collected using various sensor technologies such as optical, thermal and capacitive [13–15]. These datasets have been provided for benchmarking purposes in algorithm performance evaluation. To extend these datasets for the latent fingerprint recognition study, we generate latent like images of some images of these public datasets using a deep learning based general adversarial network model. To this end, we trained a CycleGAN [28] model and applied random distortion, random rotation and selection operations, sequentially, on the sensor image to emulate latent images.

CycleGAN is generally used for transferring characteristics from one image to another without paired examples of transformation in between source and target domain. We trained CycleGAN on our private dataset (consisting of 2200 latent images that are not included in EGM Test dataset) to transfer characteristics of latent fingerprints such as noise and deformation to slap and roll fingerprints. In order to generate a latent dataset, firstly, the images in the FVC dataset are given to input of the CycleGAN and the fingerprint image is synthesized with the latent characteristics extracted from the output of the CycleGAN. After transferring image characteristic from latent to slap fingerprint images, random rotation and distortion are applied to the image to change minutiae spatial position between generated latent fingerprint and slap fingerprint images. Fingerprints were selected based on the difference in the number of minutiae between the original slap fingerprint and the generated latent fingerprint. After the minutiae based selection, the one-to-one dataset is randomly selected with respect to different impressions from the FVC dataset. Fig. 4 presents several samples from this emulated dataset. We have utilized databases from 2000 to 2004 to generate a FVC-Latent test dataset comprising of 316 images.

**Tsinghua-Latent Test Dataset (Public):** This dataset is an extension of the Tsinghua Distorted Fingerprint Database images provided in [22]. Similar to the FVC-Latent test dataset, we applied CycleGAN model on the original and distorted images to generate corresponding synthetic latent images. Original dataset provides 320 pairs

[1] https://github.com/FingerGeneration/Generated_Latent_Fingerprint_Dataset


Figure 4. FVC Latent Test Dataset Examples a-c) Slap Fingerprint b-d) Fake Latent Fingerprint

of normal fingerprints and distorted fingerprints from 185 different fingers, but we discovered that there are only 168 unique fingers in this database. Therefore, our generated database consists of one to one latent image pairs for 168 unique fingers. Fig. 5 presents several samples from this emulated dataset.


Figure 5. Pairs of latent images from generated Tsinghua latent dataset; (a)-(b) and (c)-(d)

### 3.2. Results

This section reports the performance of the proposed algorithm for 3 test datasets discussed in section 3.1. When comparing the proposed approach with existing methods presented in [2] and [17], we utilized either the executable or SDK provided by the authors. Since the windows executable of the deformable minutia clustering (DMC) algorithm provided by [17] took approximately 0.55 second to finish one latent-to-sensor image comparison, we ran DMC only for 100 random sensor images (inclusive of corresponding sensor image) per each latent image in EGM test dataset. Therefore, DMC algorithm results are expected to be lower than our reported results for EGM test dataset. For the other test sets, we ran DMC on the whole set. In terms of the MCC algorithm [3], we utilized our in-house developed MCC variant that has some modifications to the original MCC algorithm, which also performs better than author provided SDK on these datasets. In our experiments, we utilized minutiae extracted using FingerNet [25] algorithm when evaluating the performance of MCC [3], DMC [17] and the proposed method.

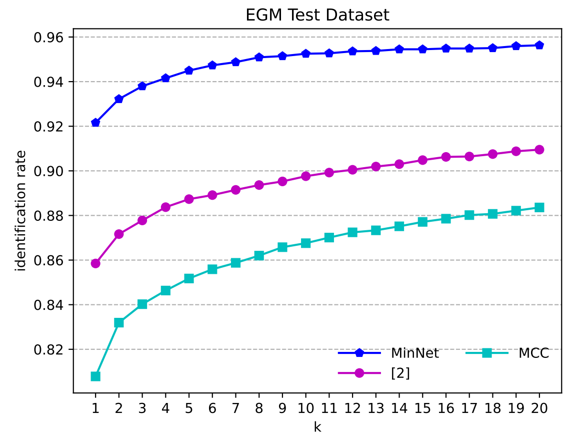



Figure 6. Cumulative Match Characteristic (CMC) curves for EGM Test dataset of methods listed in Table 2.

#### 3.2.1 EGM Test Dataset Results:

For this dataset, we generated a match score for each latent image by comparing it to 5560 sensor images in our database. Table 2 presents a comparison of rank-1, rank-5 and rank-10 accuracy values of our proposed method along with methods of [2], [17] and [3]. Rank-1 performance of the proposed method is considerably higher than that of the existing methods. Table also shows the result of MinNet model without the Minutiae Segmentation Maps (MinNet w/o Seg.), which has a rank-1 value lower than the original MinNet model. This shows the positive impact of encoding positional and angular local minutiae information when creating the patch embedding vector. Fig. 6 shows the cumulative match characteristic curve (CMC) for methods listed in Table 2.

| Methods | EGM Test Dataset | | |
|---|---|---|---|
| | Rank-1 | Rank-5 | Rank-10 |
| DMC [17] | 44.43 | 59.35 | 66.11 |
| MCC [3] | 80.59 | 84.98 | 86.66 |
| [2] | 85.88 | 88.91 | 89.92 |
| MinNet w/o Seg. | 89.87 | 92.39 | 93.15 |
| MinNet w Seg. | **92.39** | **94.71** | **95.30** |

Table 2. Rank-1, rank-5 and rank-10 identification accuracy values for EGM Test Dataset using the proposed method and earlier methods in the literature.

#### 3.2.2 FVC-Latent Test Dataset Results:

A match score is generated for each latent image in the FVC-Latent Test dataset by comparing it to 316 sensor images within the database. Table 3 presents a comparison of rank-1, rank-5 and rank-10 accuracy values of our proposed method along with methods of [2], [17] and [3]. Fig. 7.(a) illustrates top 15 minutia matches corresponding to latent and sensor image. Fig. 7.(b) shows the similarity rankings for several query patches of latent image. Green color and blue color indicates that patch is a genuine matching and impostor matching patch of the sensor image, respectively. As shown in the figure, genuine matching patches achieve the highest similarity scores. Fig. 8 shows the cumulative match characteristic curve (CMC) for methods listed in Table 3.

| Methods | FVC Latent | | |
|---|---|---|---|
| | Rank-1 | Rank-5 | Rank-10 |
| DMC [17] | 20.90 | 31.33 | 38.92 |
| MCC [3] | 72.23 | 80.13 | 83.28 |
| [2] | 65.61 | 74.44 | 77.91 |
| MinNet w/o Seg. | 93.98 | 96.83 | 97.15 |
| MinNet w Seg. | **95.57** | **97.15** | **97.47** |

Table 3. Rank-1, rank-5 and rank-10 identification accuracy values for FVC-Latent Test Dataset using the proposed method and earlier methods in the literature.

Proposed methods outperforms earlier proposed methods by a large margin. This is due to the ability of the proposed method to generate discriminative embeddings of patches around each minutia. We noticed a significant performance drop in the method of [2] due to autoencoders (AEC) poor performance on minutia extraction. Even though FingerNet algorithm was able to extract minutias for this dataset without any problems, AEC performed poorly in minutia extraction for some images.

| Embedding Size | FVC Latent | | |
|---|---|---|---|
| | Rank-1 | Rank-5 | Rank-10 |
| 32d | 87.34 | 90.51 | 92.09 |
| 64d | 91.72 | 93.67 | 95.24 |
| 128d | 94.30 | 96.20 | 96.20 |
| 256d | **95.57** | **97.15** | **97.47** |

Table 4. MinNet algorithm performance on FVC-Latent dataset for various embedding sizes.

In order to analyze the effect of embbedding size on the overall recognition performance, we added linear layer at the end of the pre-trained MinNet to decrease embedding size. Freezing pre-trained MinNet prevents decreasing contribution of neighboring minutiae information over the minutia descriptor. For this reason, we froze pre-trained MinNet model while training the initialized linear layer with new weights. Table 4 presents rank-1, rank-5 and rank-10 accuracy results over FVC Latent dataset. Fig. 9 shows the cumulative match characteristic curve (CMC) for methods listed in Table 4. As evident from these results, reducing the embedding size from original value of 256 negatively impacts the overall recognition accuracy.

#### 3.2.3 Tsinghua Latent Dataset Results:

Similar to the earlier results, we evaluated the performance of the proposed method using the emulated latent images for Tsinghua Distorted fingerprint database. Table 5 presents the latent fingerprint identification accuracy for various algorithms. While the overall performance of the algorithms are higher on this dataset than the performances on earlier datasets, proposed algorithm achieves superior results compared to the others. Due to the poor minutia extraction of AEC in [2], we have not reported performance results for the method of [2] on this dataset.

| Methods | Tsinghua Latent | | |
|---|---|---|---|
| | Rank-1 | Rank-5 | Rank-10 |
| DMC [17] | 85.71 | 92.86 | 96.43 |
| MCC [3] | 94.64 | 95.60 | 97.02 |
| MinNet | **99.40** | **99.98** | **99.99** |

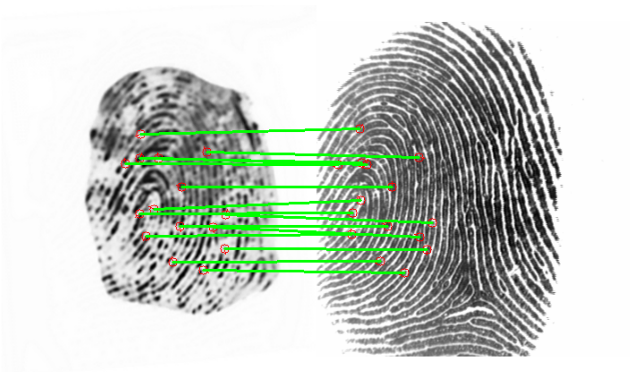
Table 5. Rank-1, rank-5 and rank-10 identification accuracy values for Tsinghua Latent Test Dataset using the proposed method and earlier methods in the literature.

#### 3.2.4 FVC 2004 Sensor Dataset Results:

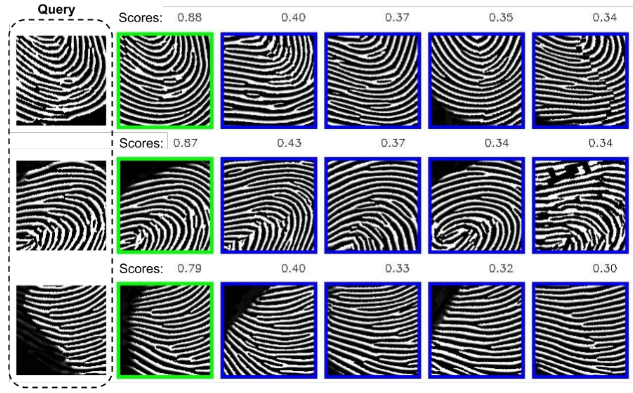
| FVC 2004 DB1 A | | | |
|---|---|---|---|
| MinNet(Ours) | DeepPrint [6] | Verifinger | Innovatrics |
| **98.46%** | 97.53%* | 96.75%* | 96.57%* |

Table 6. TAR@FAR of 0.1% values for FVC 2004 DB1 A Dataset using the proposed method and earlier methods in the literature. * Results are taken from [6].

In order to compare performance of MinNet in sensor



(a)




(b)

Figure 7. (a) Top 15 minutia matches of FVC 2002 Db1 104 1.fingerprint and 7.fingerprint. (b) In order to show best minutia matches we collected minutia descriptors in gallery. Green color indicates the matched minutia is from ground truth peer fingerprint.

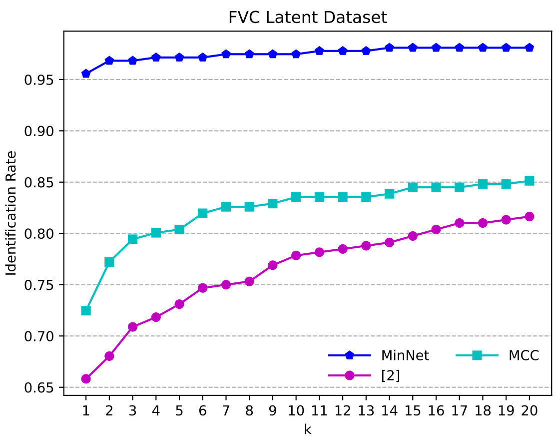



Figure 8. Cumulative Match Characteristic (CMC) curves for FVC Latent Test dataset of methods listed in Table 3.

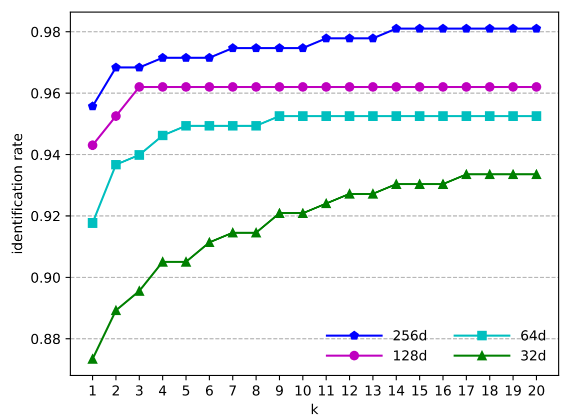



Figure 9. Cumulative Match Characteristic (CMC) curves for FVC dataset of MinNet with different descriptor sizes listed in Table 4.

fingerprint recognition task, we generated a match score for each sensor image by comparing it to images in FVC2004 DB1-A dataset similar to [6], resulting in 2800 genuine matches and 4950 impostor matches. Table 6 presents TAR@FAR of 0.1% performance for various algorithms (including [6] which is current considered state-of-the art).

The results show that MinNet outperforms earlier proposed methods as well. This shows that proposed method works well for sensor recognition tasks in addition to latent recognition tasks.

## 4. Conclusion

In this study, we proposed a novel local minutia embedding model for latent fingerprint recognition problem. Local embedding vectors generated for a fixed neighborhood around a minutia are used in the local similarity assignment algorithm to produce a global similarity match score. Proposed method has been evaluated on several public and private datasets as shown in Tables 2, 3, 5 and 6. For EGM Test dataset consisting of 5560 latent and rolled image pairs, proposed method achieved a rank-1 accuracy of % 92.39 while the best performing method of existing studies achieved a ran-1 accuracy of % 85.88. As part of this study, we have also created two new databases for latent image recognition studies using CycleGAN latent image emulation model. As show in Table 3 and 5, proposed method achieved the best performance for these databases as well; yielding a rank-1 accuracy of % 95.57 and % 99.40 , respectively. As shown in our experiments, MinNet produces state-of-the-art results not only for latent recognition task but in sensor recognition task as well.

## References


[1] Kai Cao and Anil K. Jain. Automated latent fingerprint recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(4):788–800, 2019.

[2] Kai Cao, Dinh-Luan Nguyen, Cori Tymoszek, and Anil K. Jain. End-to-end latent fingerprint search. *IEEE Transactions on Information Forensics and Security*, 15:880–894, 2020.

[3] 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.

[4] A.K. Jain S. Prabhakar D. Maltoni, D. Maio. *Handbook of Fingerprint Recognition*. Springer-Verlag, New York, 2009.

[5] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019.

[6] Joshua J Engelsma, Kai Cao, and Anil K Jain. Learning a fixed-length fingerprint representation. *IEEE transactions on pattern analysis and machine intelligence*, 43(6):1981–1997, 2019.

[7] Jude Ezeobiejesi and Bir Bhanu. Latent fingerprint image quality assessment using deep learning. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 621–6218, 2018.

[8] M. Garris. Latent fingerprint training with nist special database 27 and universal latent workstation, nist interagency/internal report (nistir), national institute of standards and technology, gaithersburg, md, 2001.

[9] R.E Gaensslen (eds.) H.C. Lee. *Advances in Fingerprint Technology*. CRC Press, New York, 2001.

[10] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1314–1324, 2019.

[11] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[12] H. W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(4):83–97, 1955.

[13] D. Maio, D. Maltoni, R. Cappelli, J.L. Wayman, and A.K. Jain. Fvc2000: fingerprint verification competition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):402–412, 2002.

[14] D. Maio, D. Maltoni, R. Cappelli, J.L. Wayman, and A.K. Jain. Fvc2002: Second fingerprint verification competition. In *2002 International Conference on Pattern Recognition*, volume 3, pages 811–814 vol.3, 2002.

[15] D. Maio, D. Maltoni, R. Cappelli, J.L. Wayman, and A.K. Jain. Fvc2004: Second fingerprint verification competition. In *International Conference on Biometric Authentication (ICBA04*, pages 1–7, 2003.

[16] Miguel Angel Medina-Perez, Milton Garcia-Borroto, Andres Eduardo Gutierrez-Rodriguez, and Leopoldo Altamirano-Robles. Robust fingerprint verification using m-triplets. In *2011 International Conference on Hand-Based Biometrics*, pages 1–5, 2011.

[17] Miguel Angel Medina-Pérez, Aythami Morales Moreno, Miguel Ángel Ferrer Ballester, Milton García-Borroto, Octavio Loyola-González, and Leopoldo Altamirano-Robles. Latent fingerprint identification using deformable minutiae clustering. *Neurocomputing*, 175:851–865, 2016.

[18] Alessandra A. Paulino, Jianjiang Feng, and Anil K. Jain. Latent fingerprint matching using descriptor-based hough transform. *IEEE Transactions on Information Forensics and Security*, 8(1):31–45, 2013.

[19] N.K. Ratha, K. Karu, Shaoyun Chen, and A.K. Jain. A real-time matching system for large fingerprint databases. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):799–813, 1996.

[20] Anush Sankaran, Mayank Vatsa, and Richa Singh. Hierarchical fusion for matching simultaneous latent fingerprint. In *2012 IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pages 377–382, 2012.

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

[22] Xuanbin Si, Jianjiang Feng, Jie Zhou, and Yuxuan Luo. Detection and rectification of distorted fingerprints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3):555–568, 2015.

[23] Maneet Singh, Richa Singh, and Arun Ross. A comprehensive overview of biometric fusion. *Information Fusion*, 52:187–205, 2019.

[24] 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.

[25] 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.

[26] Qihao Yin, Jianjiang Feng, Jiwen Lu, and Jie Zhou. Joint estimation of pose and singular points of fingerprints. *IEEE Transactions on Information Forensics and Security*, 16:1467–1479, 2021.

[27] Soweon Yoon, Jianjiang Feng, and Anil K. Jain. Latent fingerprint enhancement via robust orientation field estimation. In *2011 International Joint Conference on Biometrics (IJCB)*, pages 1–8, 2011.

[28] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2242–2251, 2017.