

Latent Fingerprint Matching via Dense Minutia Descriptor

Zhiyu Pan Yongjie Duan Xiongjun Guan Jianjiang Feng* Jie Zhou
Department of Automation, Tsinghua University, China
{pzy20, dyj17, gxj21}@mails.tsinghua.edu.cn
{jfeng, jzhou}@tsinghua.edu.cn

Abstract

Latent fingerprint matching is a daunting task, primarily due to the poor quality of latent fingerprints. In this study, we propose a deep-learning based dense minutia descriptor (DMD) for latent fingerprint matching. A DMD is obtained by extracting the fingerprint patch aligned by its central minutia, capturing detailed minutia information and texture information. Our dense descriptor takes the form of a three-dimensional representation, with two dimensions associated with the original image plane and the other dimension representing the abstract features. Additionally, the extraction process outputs the fingerprint segmentation map, ensuring that the descriptor is only valid in the foreground region. The matching between two descriptors occurs in their overlapping regions, with a score normalization strategy to reduce the impact brought by the differences outside the valid area. Our descriptor achieves state-of-the-art performance on several latent fingerprint datasets. Overall, our DMD is more representative and interpretable compared to previous methods.

1. Introduction

Fingerprints found at crime scenes, often referred to as latent fingerprints, are crucial for identifying suspects. Considering the global reliance of law enforcement agencies on latent fingerprint recognition technology [28], the inherent poor quality of such fingerprints—with indistinct ridge lines—necessitates professional examiner annotations in forensic investigations. Yet, these annotations are susceptible to discrepancies due to variations among examiners [2]. Thus, the development of an automated latent fingerprint recognition and matching system would significantly bolster the ability of law enforcement agencies to solve crimes.

*Jianjiang Feng is the corresponding author.

This work was supported in part by the National Natural Science Foundation of China under Grant 62376132.

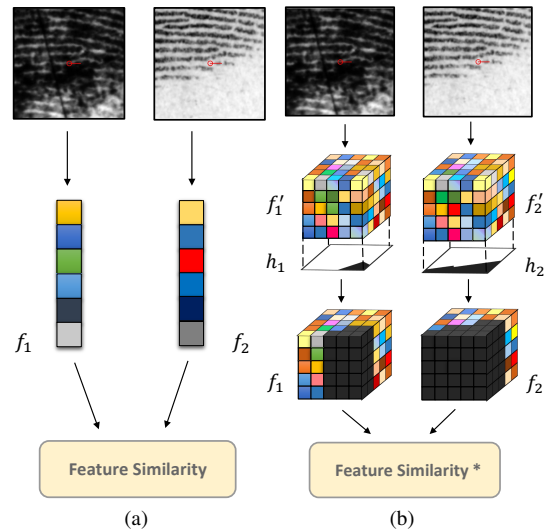


Figure 1. Compared with (a) one-dimensional minutia descriptor, (b) our Dense Minutia Descriptor (DMD) is a three-dimensional representation, and explicitly considers the overlapping area for score normalization. Score normalization is denoted as *.

Due to the complex nature of latent fingerprint acquisition, ridge lines are often blurred and may be subject to background noise interference. As a result, many researchers have focused on effectively extracting or enhancing the level-1 (orientation field, frequency map, etc. [12, 42, 8]) and level-2 (ridge skeleton map, minutiae, etc. [26, 22, 43, 25, 6, 36, 27]) features of latent fingerprints. These methods have significantly enhanced the matching performance of conventional fingerprint matching techniques; however, these feature extraction steps may introduce noisy features or destroy original features, underscoring the need for latent fingerprint matching algorithms to exhibit robustness against the challenges posed by low-quality fingerprints.

Given the limitations of handcrafted features [24, 11, 4, 32] in adapting to diverse fingerprint types and low quality situations, deep learning has been explored for abstract feature extraction in fingerprint matching. These methods are categorized into fixed-length descriptors and minutia-

based representations. The former encodes fingerprints into fixed-length vectors, enhancing indexing efficiency, with approaches like multi-scale descriptors [34, 20] and integrating minutiae with texture features [35, 10, 41]. Recent works have also utilized Vision Transformer’s potent extraction abilities for more comprehensive descriptors [16, 17]. However, these methods struggle with the nuanced description of latent fingerprints, which challenge single-vector representations due to their interference propensity. Additionally, fingerprint pose alignment [23, 9] necessary for most fixed-length techniques is compromised by the blurriness and incompleteness of latent fingerprints.

Minutia-based fingerprint matching techniques hinge on aligning fingerprint images using each minutia’s location and direction to subsequently extract local patch features. This approach, inherently resistant to overall fingerprint pose changes, provides superior accuracy to most fixed-length methods, albeit less efficiently [18]. Works by [5, 29, 31] have concentrated on encoding minutia relationships within a patch, designating one minutia as the anchor point, with Öztürk et al. [31] incorporating CNNs for concurrent texture feature encoding. Beyond using minutiae as anchors, Cao et al. [2, 3] integrated orientation fields as dense anchors for comprehensive depictions, termed virtual minutiae or texture templates [1]. Similarly, Gu et al. [19] applied dense uniform sampling points as anchors to learn relative patch alignment and descriptor extraction.

In this study, we introduce a deep-learning representation termed Dense Minutia Descriptor (DMD) which is representative and interpretable. Our approach diverges from the conventional use of one-dimensional deep representations, instead employing a dense descriptor in a three-dimensional form. This format not only retains spatial relations intrinsic to the original image, enhancing interpretability, but also aligns closely with the actual image structure, where the two dimensions represent a coarse image mapping and the third encodes texture features in depth. Our method’s interpretability facilitates direct comparison between descriptors, mirroring specific local correspondences of the source images. To further refine matching precision, our network generates a segmentation map that isolates overlapping regions, thereby reducing background noise in descriptor comparisons. The matching between DMDs is considered only in their overlapping region. Drawing inspiration from [7], we also incorporate a matching score normalization technique based on the overlapped area, minimizing the influence of the area of overlapping region. The comparison between DMDs matching and other methods is shown in Figure 1. Architecturally, our model adopts a dual-branch system akin to that of Engelsma et al. [10], isolating the extraction of texture and minutiae-specific features to bolster fingerprint recognition accuracy.

In this study, we conduct the experiments on two most

commonly used latent datasets, NIST SD27 [15] and NIST SD302 Latent subset (N2N Latent) [14]. To thoroughly validate the effectiveness of the descriptors, we do not employ any preprocessing of the fingerprint images like image enhancement. The experiments demonstrate that our method outperforms other deep-learning based descriptor methods [3, 31], conventional well-designed descriptor [5], and Commercial Off-The-Shelf (COTS) method [30]. Besides, DMD maintains good performance even after binarization, thus indicating its potential for practical applications as an automated fingerprint recognition system.

2. Method

2.1. Descriptor Extraction Network

The basic backbone architecture is modified from ResNet-34 [21] by removing the first max pooling layer for preserving the details of fingerprint ridges. Furthermore, we design a dual-branch structure which is split at the second residue block sets. One branch is tailored to produce a texture descriptor, complemented by a segmentation map as an auxiliary output. Correspondingly, the second branch is focused on generating a minutiae descriptor, alongside a minutiae map, also serving as an auxiliary output. In order to augment the network’s ability to incorporate spatial information during comparing descriptors, we employ a 2D positional embedding [38] with the well-known sinusoidal form at each branch. The final Dense Minutia Descriptor (DMD) is the concatenation of two descriptors timed by the segmentation map. The overall structure is shown in Figure 2.

Texture Descriptor. At the texture branch, it outputs the segmentation map h with auxiliary 2D convolution layers denoted as segmentation head. It enables a heightened focus on the distribution of the ridge lines. Additionally, the segmentation map h is essential for DMD and serves a critical role in matching score normalization. The texture descriptor head shares the same feature map from the last residue block sets as the segmentation head. Consequently, we obtain the texture descriptor f_t and the segmentation map h , where $h \in \mathbb{R}^{1 \times 8 \times 8}$ and $f_t \in \mathbb{R}^{C \times 8 \times 8}$, with C indicating the depth dimension.

Minutiae map. Considering the large number and complex figure of minutiae within fingerprint images, we conceptualize the distribution of minutiae’s position and orientation to a 6-channel 3D heatmap called minutiae map inspired by [37, 10]. Here, two dimensions correspond to the image plane, while the third dimension encompasses angles ranging from 0° to 360° . Each minutia is depicted using a Gaussian distribution, characterized by a variance of σ^2 , centered around its specific position and orientation denoted by (x, y, θ) . For our specific application, we have chosen to set the parameter σ equal to 1.

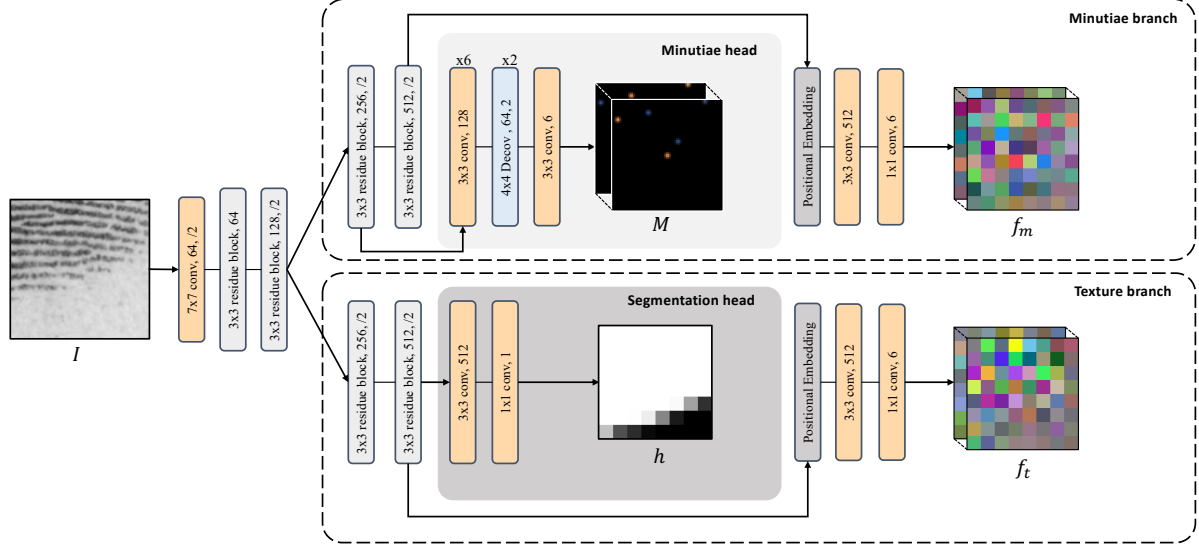


Figure 2. The detailed structure of our DMD extraction network. The content boxes display operation names, output channels, and spatial scales separated by commas. The third one is omitted if scale equals 1.

Minutiae Descriptor. Minutiae is the detailed feature (level-2) of fingerprint, and hence we extract the feature of penultimate residue block sets to feed the minutiae. The minutiae head is composed by sets of 2D convolution layers and 2D Deconvolution layers to resume the high resolution minutiae map. It enables this branch focus more on the minutiae-related feature. The minutiae descriptors head is directly connected to the last result block sets. Consequently, we can obtain the minutiae descriptor $f_m \in \mathbb{R}^{C \times 8 \times 8}$ and minutiae map $M \in \mathbb{R}^{6 \times 64 \times 64}$, with the C is the same as the one in texture descriptor.

Finally, we can get the dense descriptor $f \in \mathbb{R}^{2C \times 8 \times 8}$ by concatenating two types descriptors and multiplexing the segmentation map h as

$$f = (f_t \oplus f_m) \odot h, \quad (1)$$

where \oplus denotes the concatenation and \odot denotes the dot product.

2.2. Training Loss

Classification Loss. We incorporate the robust CosFace loss [39] used in face recognition to refine the learning of fingerprint feature representations. This loss function is applied distinctly to both minutiae and texture descriptors. For each descriptor, the process begins with flattening, followed by passing through a fully connected (FC) layer that categorizes into V classes. Here, V also signifies the count of minutiae-centered training image pairs, a procedure detailed in Sec. 2.3. The loss is calculated by

$$\mathcal{L}_{\text{cls}}^i = -\frac{1}{N} \sum_{n=1}^N \log \frac{e^{A(\cos(\theta_{y_n}^i) - b)}}{e^{A(\cos(\theta_{y_n}^i) - b)} + \sum_{v=1, v \neq y_n}^V e^{A \cos(\theta_v^i)}}, \quad (2)$$

where $\cos(\theta_v^i)$ is calculated by

$$\cos(\theta_v^i) = W_v^T f_i, \quad W_v = \frac{W_v}{\|W_v\|}, \quad f_i = \frac{f_i}{\|f_i\|}. \quad (3)$$

The A is employed for normalizing magnitude, with i signifying the type such that $i \in \{t, m\}$. The W_i represents the learnable weight vector of the v -th class. The term b represents the margin, N quantifies the count of samples per batch, and y_n indicates the class label of the sample f_i .

Segmentation Loss and Minutiae Loss. We adopt the binary cross entropy loss for calculating segmentation loss \mathcal{L}_{seg} and utilize mean square error for calculating minutiae loss \mathcal{L}_{mnt} . We utilize the VeriFinger v12.0 to extract the target minutiae and segmentation maps.

Similarity Loss. To ensure the local feature consistency within fingerprints from the same finger with different distortion or valid area, we simulate a counterpart plain fingerprint using the segmentation map from Diverse Pose Fingerprint dataset (DPF) [9] and a simulated distortion field following the model [33] according to the original rolled one. Similarity loss is to keep the corresponding region's feature of the rolled fingerprint similar to the feature of plain one. It is defined as

$$\mathcal{L}_{\text{sim}} = \frac{1}{|h_p \cap r|} \sum_{(i,j) \in h_p \cap r} \|f_p^{ij} - f_r^{ij}\|^2, \quad (4)$$

where f_p and f_r denote the representations extracted from plain and rolled fingerprints respectively.

Therefore, the overall supervision loss is defined as

$$\mathcal{L} = \sum_i^{\{t, m\}} \mathcal{L}_{\text{cls}}^i + \lambda_{\text{seg}} \mathcal{L}_{\text{seg}} + \lambda_{\text{mnt}} \mathcal{L}_{\text{mnt}} + \lambda_{\text{mnt}} \mathcal{L}_{\text{sim}}, \quad (5)$$

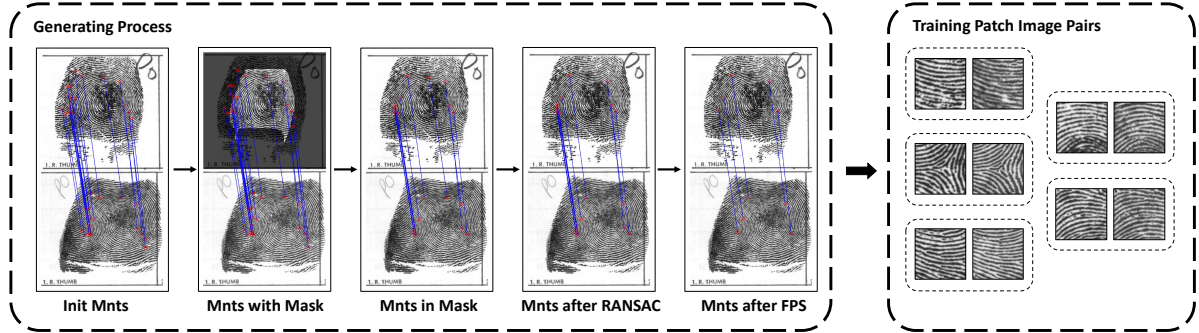


Figure 3. The process of selecting training minutiae pairs.

where λ_{seg} , λ_{mnt} , and λ_{mnt} are weight to balance the loss components.

2.3. Training Sample Generation

In contrast to the approach of directly selecting minutiae according to MCC [5] minutiae pair matching scores as presented in [31], our methodology incorporates a multitude of selection strategies. These strategies are designed to identify minutiae that not only are correctly matched but also resilient to distortion. Furthermore, they facilitate the selection of distinct regions for network training. This approach enables the network to learn distinctive features across varying fingerprint patches, enhancing its capability to differentiate between unique minutiae configurations. The training samples generation process is shown in Figure 3.

Extracting Minutiae. We utilize VeriFinger v12.0 [30] to extract minutiae from fingerprint images. Faced with a scarcity of available public latent fingerprints for our training needs, we resort to employing the once publicly available rolled fingerprint dataset NIST SD14 [40] as our training dataset.

Mated Minutiae. We employ the Minutia Cylinder-Code (MCC) [5] method to identify corresponding minutiae pairs in genuine fingerprint matches. However, it’s important to note that not all identified minutiae pairs are accurate. Initially, we prioritize the top K minutiae pairs based on their matching scores. Subsequently, considering the bad training quality of regions located at the edges of the fingerprint’s foreground which is susceptible to erroneously identifying minutiae, We utilize the segmentation map obtained from the enhancement process conducted by VeriFinger v12.0, applying erosion, to exclude minutiae located in invalid regions. Moreover, we employ the RANSAC algorithm to compute a 2D affine transformation matrix that aligns the source minutiae with the target minutiae. This step facilitates the removal of incorrectly matched minutiae pairs as well as those affected by significant fingerprint distortions. Finally, we acknowledge that training images generated from closely situated minutiae often resemble each other, thus posing a challenge for differentiation during the

training phase. To mitigate this issue, we implement Farthest Point Sampling (FPS) to judiciously choose a subset comprising a maximum of K minutiae, which ensuring that the selected minutiae are spaced sufficiently far apart. We designated $N = 12$ and $K = 5$ for our training.

Generate Image Samples. After the creation of matched minutiae pairs, we proceed to transform the training image samples. This transformation involves translating and rotating the fingerprint images to align with the position and orientation of each specific minutia, and then cropping. As a result, the transformed patch images are centered on the anchor minutiae, with these minutiae oriented horizontally to the right. In our application, we opted for a patch size of 128×128 pixels in 500 ppi.

2.4. Fingerprint Matching






Our fingerprint matching process unfolds in two stages: calculating local similarities between two minutia sets from two fingerprint images; getting the final matching score of two images from the local similarity matrix.


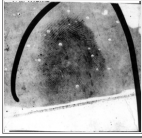

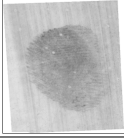
Initially, we compute the initial score matrix $S_{(A,B)}$ by comparing minutiae dense descriptors for the pair of fingerprints (A, B) under comparison. The matching score for a pair of minutiae dense descriptors is determined through the cosine similarity between the two flattened descriptors. Subsequently, we adopt the score normalization technique outlined in [7] to mitigate the impact of variations in the area of the overlapping region on the score. After getting the descriptors from two minutiae (a_i, b_j) by Eq. (1) and flattening them (f_{a_i}, f_{b_j}) , the score is computed by

$$S_{(A,B)}(i, j) = \frac{\langle f_{a_i}, f_{b_j} \rangle}{\|f_{a_i}\| \|f_{b_j}\|} \cdot \sqrt{\frac{h_o}{H_o}}, \quad h_o = |h_{a_i \cap b_j}|, \quad (6)$$

where (a, b) represent the minutiae sets from fingerprints (A, B) , h_o is the area of overlapping region, H_o is a constant that reflecting the average of overlapping area. We set $H_o = 1326$ in our method.

Subsequently, we apply the Local Similarity Assignment with Relaxation (LSA-R) method, as introduced in

| Dataset | NIST SD14 [†] | THU Contact10K [‡] | | DPF | |
|-------------|---|---|---|---|---|
| | Rolled | Rolled or Plain | | Plain | |
| Image |  |  |  |  |  |
| Sensor | Inking | Inking / Optical | | Optical | |
| Description | 27,000 pairs | 10,458 fingerprints | | 40,112 fingerprints | |
| Usage | Training | Testing | | Augmentation | |

| Dataset | NIST SD27 [†] | | N2N Latent | |
|-------------|---|---|--|---|
| | Rolled | Latent | Rolled | Latent |
| Image |  |  |  |  |
| Sensor | Inking | — | Optical | — |
| Description | 258 pairs | | 2,000 fingerprints | 3,318 fingerprints |
| Usage | Testing | | Testing | |

[†] Corresponding datasets are discontinued.

[‡] Corresponding datasets are internal.

Table 1. All fingerprint datasets used in this study.

MCC [5], to derive the final matching score from similarity matrix $S_{(A,B)}$ and minutiae sets (a, b) . The LSA-R method addresses the linear assignment problem using $S_{(A,B)}$ through a combination of the Hungarian algorithm and a relaxation approach that takes into account the geometric configuration of the minutiae sets (a, b) [13]. Then we get the adjusted score matrix $S'_{(A,B)}$, and get the top n_m matching scores related to the minutiae sets $m = \{(a_i, b_i), i = 1, \dots, n_m\}$. The n_m is calculated as

$$n_m = \min_{n_m} + \lfloor \frac{\max_{n_m} - \min_{n_m}}{1 + e^{(-\tau(\min(n_a, n_b) - \mu))}} \rfloor, \quad (7)$$

(n_a, n_b) represents the number of minutiae sets (a, b) . We set the hyper parameters as $\min_{n_m} = 4, \max_{n_m} = 12, \tau = 0.4, \mu = 20$. $\lfloor \cdot \rfloor$ is the rounding operator. Finally, the matching score $\Gamma(A, B)$ between fingerprints (A, B) is calculated as

$$\Gamma(A, B) = \frac{\sum_{(r,c) \in m} S'_{(A,B)}(r, c)}{n_m}. \quad (8)$$

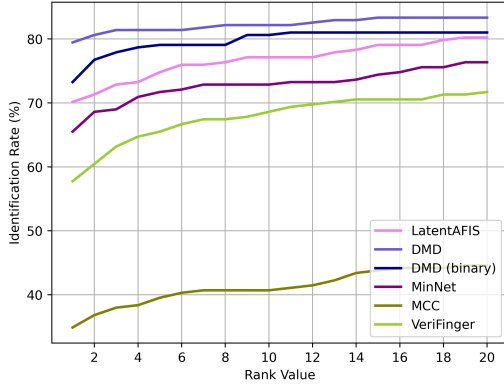
Implementation Details. To enhance data diversity, we generate plain fingerprints by cropping rolled fingerprints using segmentation maps from DPF dataset as described in Similarity Loss of Sec. 2.2. Minutiae outside the segmentation map’s effective area are excluded. Augmentation also includes random translations up to 10 pixels, rotations

within the range of $[-5^\circ, 5^\circ]$, intensity transformation, distortions, and Gaussian noise. We apply the distortion model by Si et al. [33] to generate distortion fields. For optimal performance and computational efficiency, we configure the descriptors dimension C to 6 and adjust the parameters A and b in Eq. (2) to 30 and 0.4, respectively. The parameters $\lambda_{\text{seg}}, \lambda_{\text{mnt}},$ and λ_{mnt} in Eq. (5) are set to 1, 0.01, and 0.00125, respectively. We employ the AdamW optimizer with a learning rate of 3.5×10^{-4} , and apply L2 regularization to the trainable parameters to prevent overfitting.

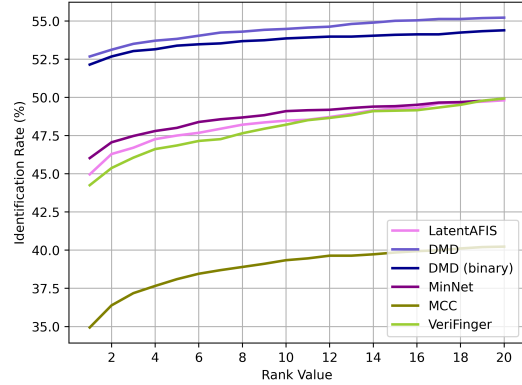
3. Experiment

3.1. Datasets

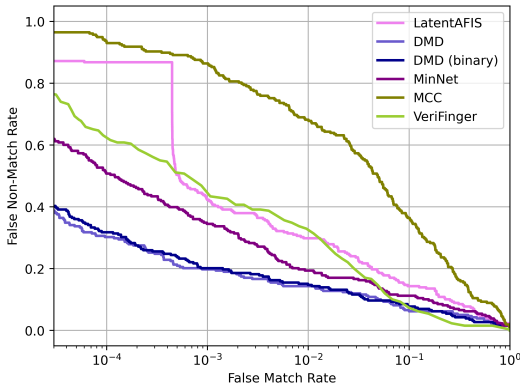
In our study, we primarily rely on five datasets for both training and evaluation. Table 1 presents an overview of these datasets, including examples of fingerprints from each. For the training phase, we employ the NIST SD14 rolled fingerprint dataset, generating a total of 132,550 pairs of minutiae-centered patch fingerprints from its 27,000 pairs of fingerprints. Diverse Pose Fingerprint dataset (DPF) [9], encompassing 776 fingers captured in a variety of poses, is utilized to simulate plain images for data augmentation. It is important to note that we do not engage in any fine-tuning on latent fingerprints for evaluation purposes. The NIST SD302 (N2N) dataset comprises fingerprints from 200 individuals, totaling 2,000 fingers. In



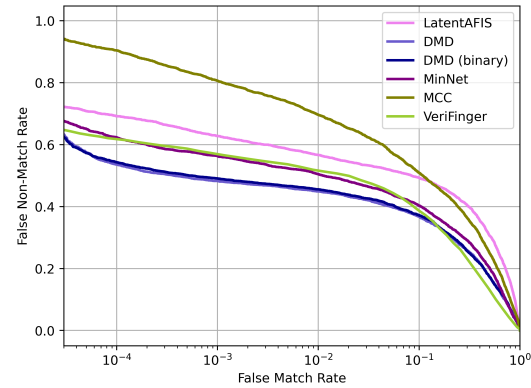
(a) CMC curves of NIST SD27



(b) CMC curves of N2N Latent



(c) DET curves of NIST SD27



(d) DET curves of N2N Latent

Figure 4. Latent fingerprint matching performance on NIST SD27 (a) (c) and N2N Latent (b) (d).

| Type | Approach | Setting |
|---------------|-----------------|----------------------|
| Deep Learning | MinNet [31] | Our reimplementation |
| | LatentAFIS [3] | Original public code |
| Convention | MCC [5] | Our reimplementation |
| COTS | VeriFinger [30] | Commercial SDK |

Table 2. Experiment settings of approaches to be compared.

particular, subset U serves as our gallery. Following the selection criteria detailed in [20], we choose 3,383 latent fingerprints with reasonable quality out of 10,000 available. Additionally, the NIST SD27 dataset includes 258 latent-to-rolled fingerprint pairs. To expand the gallery of NIST SD27, we incorporate a private dataset denoted as THU Contact10K dataset, which contains 10,458 rolled or plain fingerprints from ten fingers of over 1046 different subjects.

3.2. Compared Methods

To ascertain the effectiveness of Dense Minutia Descriptor (DMD), we conduct a comprehensive comparison with both traditional and more recent deep learning minutiae-based fingerprint recognition methods (Table 2). Specifi-

cally, our evaluation of LatentAFIS [3] utilize the publicly available code and the released model weights provided by the authors. We retain the entire pipeline of their system, which includes image enhancement, the extraction of minutiae and virtual minutiae templates, as well as the processing of descriptors. Regarding MinNet [31], we train it with original patch fingerprint images instead of enhancement ones which are the same as ours, and increase the dimensionality of its descriptors to 768 to align with our configuration. As to commercial matcher VeriFinger v12.0, it has two types of matchers: ISO minutia-only template and proprietary template consisting of minutiae and other features. To make a high baseline, we adopt the proprietary template which presents higher performance. And we reimplement MCC [5] to achieve a faster extraction and matching speed while maintaining the same matching performance as its public SDK. The minutiae of testing fingerprint images were extracted using VeriFinger and thus identical across these methods, with the exception of LatentAFIS, which extracts its minutiae and templates using the model weights provided in its release. Fingerprint matching process for MCC, MinNet, and DMD follows the same procedure, as detailed in Sec. 2.4, allowing a fair comparison of different minutia descriptors. And the matching strategies for Ver-

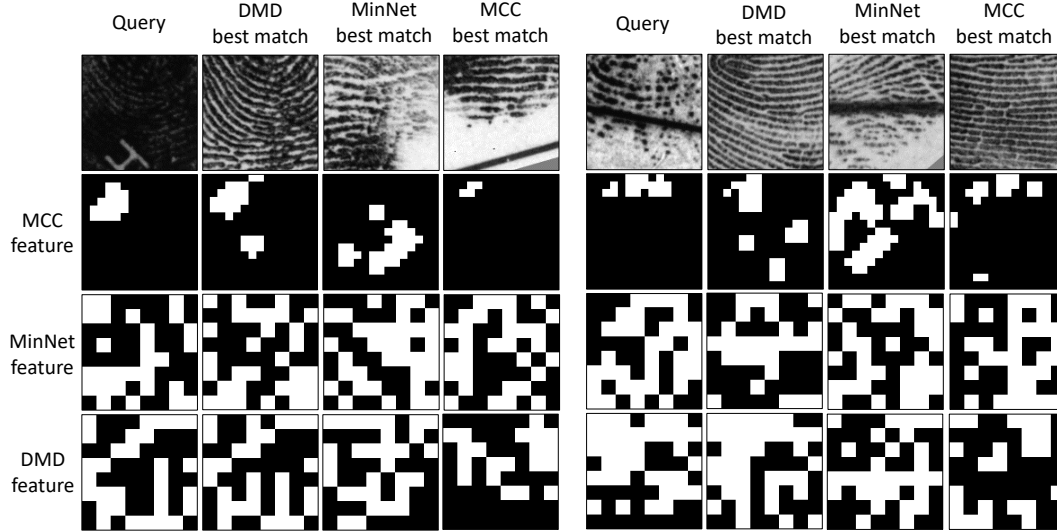


Figure 5. Descriptor visualization on different patch images. Descriptors of MinNet are resized to three-dimension form for visualization. We select a specific channel from the aforementioned descriptors and convert it to a binary format to enhance visualization.

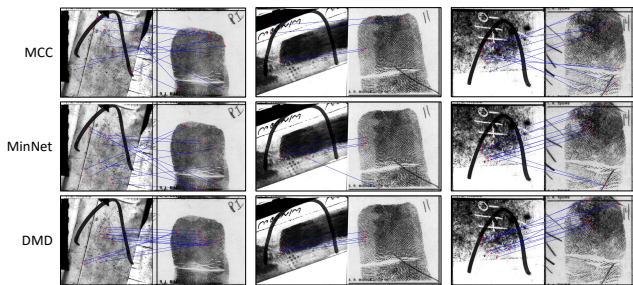


Figure 6. Top n_m minutiae patch matching of three genuine pairs via different methods. n_m is determined by Eq. (7).

| Method | NIST SD27 | | N2N Latent | |
|-----------------|--------------|--------------|--------------|--------------|
| | Rank-1 | TAR | Rank-1 | TAR |
| MCC [5] | 35.27 | 13.57 | 34.94 | 19.42 |
| VeriFinger [30] | 53.10 | 58.91 | 44.31 | 42.71 |
| LatentAFIS [3] | 70.16 | 57.75 | 44.96 | 37.22 |
| MinNet [31] | 65.89 | 65.50 | 46.02 | 43.63 |
| DMD (binary) | <u>73.26</u> | <u>79.84</u> | <u>52.14</u> | <u>50.90</u> |
| DMD | 79.07 | 80.23 | 52.68 | 51.73 |

Table 3. Verification and recognition accuracy (%) on latent fingerprint datasets. TAR@FAR=0.1% is reported.

iFinger and LatentAFIS adhere to the protocols established by their respective systems.

3.3. Latent Fingerprints Matching Performance

The proposed DMD is benchmarked against the array of methodologies as delineated in Table 2. Rank-1 and TAR@FAR=0.1% metrics, alongside the Cumulative Match Characteristic (CMC) curve and the Detection-Error

Tradeoff (DET) curve, serve as crucial tools for the quantitative evaluation of various methods. Moreover, we extend our work to include a binary variant of DMD, wherein $\bar{f}_{t,m}$ is binarized to 0 and h is thresholded at 0.5 for binarization. It culminates in a more streamlined version of the DMD, with each descriptor being condensed to occupy merely 96 bytes.

Compared to other minutia-based fingerprint matching methods, DMD stands out by a significant margin in terms of matching and indexing capabilities, as demonstrated in Table 3 and Figure 4. This superiority continues to hold even when compared to binary DMD. The exceptional performance of DMD can be attributed to its remarkable representation. Figure 5 showcases three primary types of descriptors: spatial representation derived from minutia distribution (MCC), one-dimensional representation consisting of abstract features (MinNet), and spatial representation modeled from texture and minutiae information using abstract features (DMD).

One-dimensional descriptors, which do not closely correlate with the spatial characteristics of original fingerprints, may face challenges. These include difficulty in isolating the impact of noise present in latent fingerprints, which can affect the entire descriptor, as well as limitations in the interpretability of the descriptor itself. And we can observe that it exhibits an irregular pattern across features of these samples in Figure 5. MCC maintains a spatial relationship with the fingerprint’s plane, yet it primarily models the distribution of minutiae in the vicinity, making it highly susceptible to inaccuracies caused by erroneously detected or missing minutiae. DMD not only retains the spacial representation like MCC but also uses the robust abstract features. From Figure 5, it can be observed that the DMD feature pattern

| Modification | NIST SD27 | | N2N Latent | |
|---------------|--------------|--------------|--------------|--------------|
| | Rank-1 | TAR | Rank-1 | TAR |
| w/o Norm | 76.74 | 79.07 | 49.28 | 49.13 |
| $C = 3$ | 75.58 | 78.68 | 48.57 | 48.30 |
| Single Branch | 66.28 | 68.99 | 48.45 | 48.06 |
| None | 79.07 | 80.23 | 52.68 | 51.73 |

Table 4. Ablation study on the verification and recognition accuracy (%) of DMD. TAR@FAR=0.1% is reported.

of the query in the first example closely resembles its best match, except for the lower left part which is affected by the symbol ‘‘H’’. Similarly, in the second example, the upper region of the matching pair set exhibits a strong resemblance in DMD features, despite the incomplete bottom region of the searched latent fingerprint and interference from a black line.

Moreover, the top n_m patch matching selected by feature similarity of DMD is more accurate than others which is illustrated in Figure 6. This indicates that the value of the matching score can, to a great extent, determine whether the match is genuine. This also facilitates a better understanding of our strong performance on the TAR metric and the DET curve. It indicates the potential of DMD used in large-scale fingerprint indexing.

3.4. Ablation Study

In this section, we explore the impact of various modifications made to our proposed DMD, which include omitting the normalization approach outlined in Eq. (6), decreasing the DMD dimensionality from $C = 6$ to $C = 3$, and merging two branches into a single one with segmentation head and minutiae head. It is noteworthy that the descriptor derived from a singular branch retains the dimensionality equivalent to the dual-branch with $C = 3$, implying that $f \in \mathbb{R}^{6 \times 8 \times 8}$. The quantitative outcomes of these modifications are summarized in Table 4.

We can observe that a normalization strategy, which takes the overlapping region into account, significantly improves the DMD matching performance. It effectively addresses the common scenario of low genuine match overlap areas in latent fingerprint matching, hence significantly improving the Rank-1 metric. Besides, DMD’s performance deteriorates as dimensionality reduces ($C = 3$), yet it still outperforms the single-branch one of the same dimensionality. Therefore, the dual-branch design, integrating different features (texture feature and minutiae feature), greatly enhances the model’s performance.

4. Limitation and Future Works

Despite good performance of our proposed Dense Minutiae Descriptor (DMD) on several latent fingerprint datasets,

there remains room for enhancement in several aspects. The effectiveness of DMD heavily relies on the accuracy of the preceding minutiae extraction processes. In this study, we utilize VeriFinger v12.0 [30] for minutiae extraction. Although VeriFinger excels at identifying minutiae in medium or high quality fingerprints, its performance is compromised by many latent fingerprints with complex background patterns, often resulting in the extraction of incorrect minutiae from the background areas (Figure 6). Furthermore, in datasets such as NIST SD27 or N2N Latent, VeriFinger sometimes fails to extract a sufficient number of minutiae for effective matching. Thus, the potential for improving DMD may lie in leveraging a more robust latent fingerprint minutiae extractor or refining the selection of minutiae from existing tools with a precise foreground mask.

Secondly, our current approach does not incorporate the use of enhanced fingerprints as input. Insights from Grosz et al. [18] suggest that the efficacy of latent fingerprint matching methods can significantly benefit from a tailored latent fingerprint enhancement technique. Motivated by this understanding, we aim to develop a bespoke latent fingerprint enhancement method that is specifically designed to improve the performance of DMD.

Given scenarios where the minutiae extractor fails to retrieve an adequate number of minutiae for effective matching, employing virtual minutiae (derived from the orientation field) [2, 1, 3] as anchor points can be a viable solution. By adopting such approach, we have the potential to further enhance the matching performance of DMD in latent fingerprints. Moreover, this methodology could also prove advantageous for the matching of small fingerprints collected from smartphones or other mobile devices.

5. Conclusion

In this study, we introduce a deep network based dense minutia descriptor named DMD. This descriptor is presented as a three-dimensional construct, where two dimensions are aligned with the original image plane, and the third dimension encapsulates robust abstract features. To refine and enhance the representational capacity of DMD, we employ a strategic selection of training samples alongside a dual-branch architecture for its training. Additionally, the feature visualization sheds light on its interpretability within the context of fingerprint matching. We conducted evaluations of DMD against other contemporary methods using the NIST SD27 and N2N Latent datasets. The results demonstrate that DMD significantly outperforms competing methodologies in terms of Rank-1 identification rate and True Acceptance Rate (TAR) metrics. Remarkably, DMD maintains good matching performance even after undergoing a straightforward binarization process, which contributes to improved matching efficiency and the potential for secure template encryption.

References

- [1] K. Cao and A. K. Jain. Latent fingerprint recognition: Role of texture template. In *2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems*, pages 1–9, 2018.
- [2] K. Cao and A. K. Jain. Automated latent fingerprint recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(4):788–800, 2019.
- [3] K. Cao, D.-L. Nguyen, C. Tymoszek, and A. K. Jain. End-to-End latent fingerprint search. *IEEE Transactions on Information Forensics and Security*, 15:880–894, 2020.
- [4] R. Cappelli. Fast and accurate fingerprint indexing based on ridge orientation and frequency. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 41(6):1511–1521, 2011.
- [5] 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.
- [6] L. N. Darlow and B. Rosman. Fingerprint minutiae extraction using deep learning. In *2017 IEEE International Joint Conference on Biometrics*, pages 22–30, 2017.
- [7] J. Daugman. New methods in iris recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 37(5):1167–1175, 2007.
- [8] Y. Duan, J. Feng, J. Lu, and J. Zhou. Orientation field estimation for latent fingerprints with prior knowledge of fingerprint pattern. In *2021 IEEE International Joint Conference on Biometrics*, pages 1–8, 2021.
- [9] Y. Duan, J. Feng, J. Lu, and J. Zhou. Estimating fingerprint pose via dense voting. *IEEE Transactions on Information Forensics and Security*, 18:2493–2507, 2023.
- [10] J. J. Engelsma, K. Cao, and A. K. Jain. Learning a fixed-length fingerprint representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(6):1981–1997, 2021.
- [11] J. Feng. Combining minutiae descriptors for fingerprint matching. *Pattern Recognition*, 41(1):342–352, 2008.
- [12] 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.
- [13] Y. Feng, J. Feng, X. Chen, and Z. Song. A novel fingerprint matching scheme based on local structure compatibility. In *18th International Conference on Pattern Recognition*, volume 4, pages 374–377, 2006.
- [14] G. P. Fiumara, P. A. Flanagan, J. D. Grantham, K. Ko, K. Marshall, M. Schwarz, E. Tabassi, B. Woodgate, and C. Boehnen. NIST special database 302: Nail to nail fingerprint challenge. 2019.
- [15] M. D. Garris and M. D. Garris. *NIST special database 27: Fingerprint minutiae from latent and matching tenprint images*. US Department of Commerce, National Institute of Standards and Technology, 2000.
- [16] S. A. Grosz, J. J. Engelsma, R. Ranjan, N. Ramakrishnan, M. Aggarwal, G. G. Medioni, and A. K. Jain. Minutiae-guided fingerprint embeddings via vision transformers. *arXiv preprint arXiv:2210.13994*, 2022.
- [17] S. A. Grosz and A. K. Jain. AFR-Net: Attention-driven fingerprint recognition network. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, pages 1–1, 2023.
- [18] S. A. Grosz and A. K. Jain. Latent fingerprint recognition: Fusion of local and global embeddings. *IEEE Transactions on Information Forensics and Security*, 18:5691–5705, 2023.
- [19] S. Gu, J. Feng, J. Lu, and J. Zhou. Latent fingerprint registration via matching densely sampled points. *IEEE Transactions on Information Forensics and Security*, 16:1231–1244, 2021.
- [20] S. Gu, J. Feng, J. Lu, and J. Zhou. Latent fingerprint indexing: Robust representation and adaptive candidate list. *IEEE Transactions on Information Forensics and Security*, 17:908–923, 2022.
- [21] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.
- [22] X. Huang, P. Qian, and M. Liu. Latent fingerprint image enhancement based on progressive generative adversarial network. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 3481–3489, 2020.
- [23] M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. *Advances in Neural Information Processing Systems*, 28, 2015.
- [24] A. Jain, S. Prabhakar, L. Hong, and S. Pankanti. FingerCode: A filterbank for fingerprint representation and matching. In *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No. PR00149)*, volume 2, pages 187–193 Vol. 2, 1999.
- [25] L. Jiang, T. Zhao, C. Bai, A. Yong, and M. Wu. A direct fingerprint minutiae extraction approach based on convolutional neural networks. In *2016 International Joint Conference on Neural Networks*, pages 571–578, 2016.
- [26] 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*, pages 895–903, 2019.
- [27] M. Liu and P. Qian. Automatic segmentation and enhancement of latent fingerprints using deep nested unets. *IEEE Transactions on Information Forensics and Security*, 16:1709–1719, 2021.
- [28] D. Maltoni, D. Maio, A. K. Jain, and J. Feng. *Handbook of Fingerprint Recognition Third Edition*. Springer Nature, 2022.
- [29] M. A. Medina-Pérez, A. M. Moreno, M. Á. F. Ballester, M. García-Borroto, O. Loyola-González, and L. Altamirano-Robles. Latent fingerprint identification using deformable minutiae clustering. *Neurocomputing*, 175:851–865, 2016.
- [30] Neurotechnology. Verifinger SDK. <https://www.neurotechnology.com/verifinger.html>.
- [31] H. İ. Öztürk, B. Selbes, and Y. Artan. MinNet: Minutia patch embedding network for automated latent fingerprint recognition. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 1626–1634, 2022.

- [32] A. Sankaran, T. I. Dhamecha, M. Vatsa, and R. Singh. On matching latent to latent fingerprints. In *2011 International Joint Conference on Biometrics*, pages 1–6, 2011.
- [33] X. Si, J. Feng, J. Zhou, and Y. Luo. Detection and rectification of distorted fingerprints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3):555–568, 2015.
- [34] D. Song and J. Feng. Fingerprint indexing based on pyramid deep convolutional feature. In *2017 IEEE International Joint Conference on Biometrics*, pages 200–207, 2017.
- [35] D. Song, Y. Tang, and J. Feng. Aggregating minutia-centred deep convolutional features for fingerprint indexing. *Pattern Recognition*, 88:397–408, 2019.
- [36] Y. Tang, F. Gao, and J. Feng. Latent fingerprint minutia extraction using fully convolutional network. In *2017 IEEE International Joint Conference on Biometrics*, pages 117–123, 2017.
- [37] Y. Tang, F. Gao, J. Feng, and Y. Liu. FingerNet: An unified deep network for fingerprint minutiae extraction. In *2017 IEEE International Joint Conference on Biometrics*, pages 108–116, 2017.
- [38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.
- [39] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu. CosFace: Large margin cosine loss for deep face recognition. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5265–5274, 2018.
- [40] C. Watson. NIST special database 14-mated fingerprint card pairs 2. *National Institute of Standards and Technology*, 1993.
- [41] S. Wu, B. Liu, Z. Wang, Z. Jia, and J. Feng. Minutiae-awarely learning fingerprint representation for fingerprint indexing. In *2022 IEEE International Joint Conference on Biometrics*, pages 1–8, 2022.
- [42] 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.
- [43] Y. Zhu, X. Yin, and J. Hu. FingerGAN: A constrained fingerprint generation scheme for latent fingerprint enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(7):8358–8371, 2023.