# Dense Registration and Mosaicking of Fingerprints by Training an End-to-End Network

Zhe Cui, *Graduate Student Member, IEEE*, Jianjiang Feng, *Member, IEEE*,
and Jie Zhou, *Senior Member, IEEE*

***Abstract*— Dense registration of fingerprints is a challenging task due to elastic skin distortion, low image quality, and self-similarity of ridge pattern. To overcome the limitation of handcraft features, we propose to train an end-to-end network to directly output pixel-wise displacement field between two fingerprints. The proposed network includes a siamese network for feature embedding, and a following encoder-decoder network for regressing displacement field. By applying displacement fields reliably estimated by tracing high quality fingerprint videos to challenging fingerprints, we synthesize a large number of training fingerprint pairs with ground truth displacement fields. In addition, based on the proposed registration algorithm, we propose a fingerprint mosaicking method based on optimal seam selection. Registration and matching experiments on FVC2004 databases, Tsinghua Distorted Fingerprint (TDF) database, and NIST SD27 latent fingerprint database show that our registration method outperforms previous dense registration methods in accuracy. Mosaicking experiments on FVC2004 DB1_A and a small fingerprint database demonstrate that the proposed algorithm produced higher quality fingerprints and led to higher matching accuracy, which also validates the performance of our registration algorithm.**

***Index Terms*— Fingerprint, registration, deep learning, mosaicking.**

## I. Introduction

Although automatic fingerprint recognition system has been widely deployed in various applications, current fingerprint matching algorithms still need improvement, especially for those fingerprints with large skin distortion [1]. As an inherent problem in contact-based fingerprint acquisition, skin distortion increases intra-class variations among different images of a same finger, and thus causes declines in fingerprint matching accuracy [2].

Fingerprint registration algorithms can be employed to reduce negative impact of skin distortion. Conventional fingerprint registration algorithms [3]–[5] typically first find minutiae correspondences between two fingerprints (referred to as input fingerprint and reference fingerprint), and then fit a spatial transformation model to these corresponding minutiae. However, for highly distorted fingerprints, minutiae-based registration methods can only obtain a very sparse set of displacement measures at locations of corresponding minutiae, which cannot align all ridges in two fingerprints.

In contrast to minutiae-based methods, dense fingerprint registration [6] aims to obtain pixel-wise displacement measures between two fingerprints with nonlinear skin distortion, instead of producing only sparse displacement measures. The core of dense registration is a local matching problem: for a pixel in the reference fingerprint, find the corresponding pixel in the input fingerprint. Because of self-similarity of ridge pattern, noise, and distortion, local matching is challenged by large intra-class variations among mated regions and small inter-class variations among non-mated regions, as well as large search space. As shown in Fig. 1, noise leads to changes in ridge patterns, and increases intra-class difference; self-similarity makes fingerprint have low inter-class variation, thus difficult to find the true mate among many highly similar non-mated regions; distortion changes ridge patterns, raises intra-class difference, and enlarges search space.

Regarding these challenges, current dense registration methods [6], [7] still need improvements from aspects of local displacement estimation and global deformation constraints. The phase demodulation method [7] is weak in local displacement estimation, as it can only obtain displacement perpendicular to ridge orientation, less than one ridge period, and it is sensitive to noise. The image correlation method [6] is slightly better than phase demodulation method for local displacement estimation. However, due to three aforementioned challenges in dense fingerprint registration, they cannot accurately measure the displacements. Therefore, they all rely heavily on global deformation constraints. The global deformation constraint for phase demodulation is dependent on the order of phase unwrapping, which has the problem of error accumulation [8]. The image correlation method uses Markov Random Field (MRF) to implement global deformation constraints, but it cannot solve the problem of large local measurement errors. These drawbacks of existing dense registration methods imply for needs for a more powerful method to overcome those challenges.

In recent years, convolutional neural networks prove to provide a better solution to deal with complicated fingerprint

Manuscript received April 13, 2020; revised July 19, 2020; accepted August 11, 2020. Date of publication August 19, 2020; date of current version September 10, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61976121, Grant 61622207, and Grant 61527808 and in part by the Shenzhen Fundamental Research Fund (Subject Arrangement) under Grant JCYJ20170412170438636. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Andrew Beng Jin Teoh. *(Corresponding author: Jianjiang Feng.)*

Zhe Cui and Jie Zhou are with the Beijing National Research Center for Information Science and Technology, Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: cuiz17@mails.tsinghua.edu.cn; jzhou@tsinghua.edu.cn).

Jianjiang Feng is with the Beijing National Research Center for Information Science and Technology, Department of Automation, Tsinghua University, Beijing 100084, China, and also with the Graduate School at Shenzhen, Tsinghua University, Shenzhen 518055, China (e-mail: jfeng@tsinghua.edu.cn).

Digital Object Identifier 10.1109/TIFS.2020.3017926

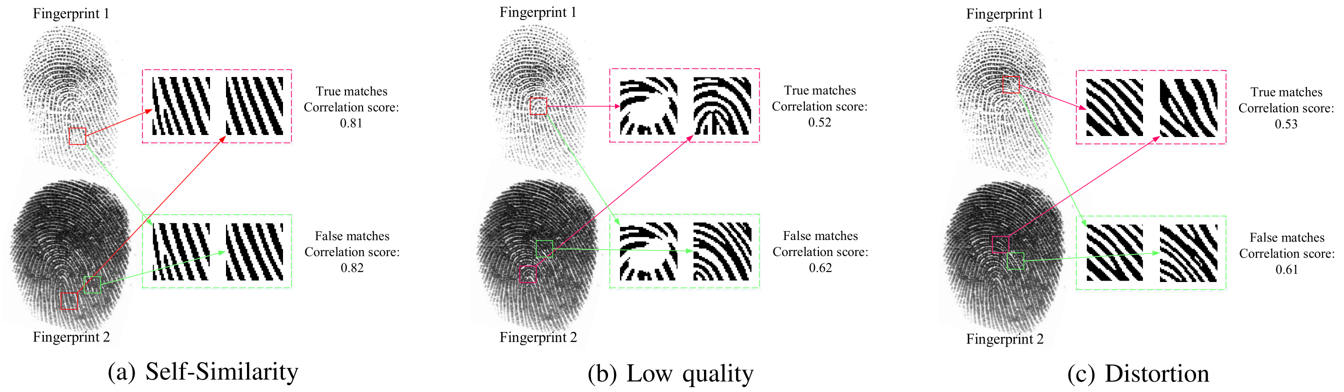


Fig. 1. Dense registration faces three challenges: high self-similarity of ridge pattern, low image quality, and large distortion. The figure shows three examples where true matching scores are lower than false matching scores according to image correlation coefficient. (a) Because of the self-similarity of fingerprints, the correlation score between true mates may be lower than false mates. (b) Ridge curves differ in low quality regions, which troubles local matching by increasing intra-class variations. (c) Distortion increases intra-class variations and enlarges search space for possible matches.

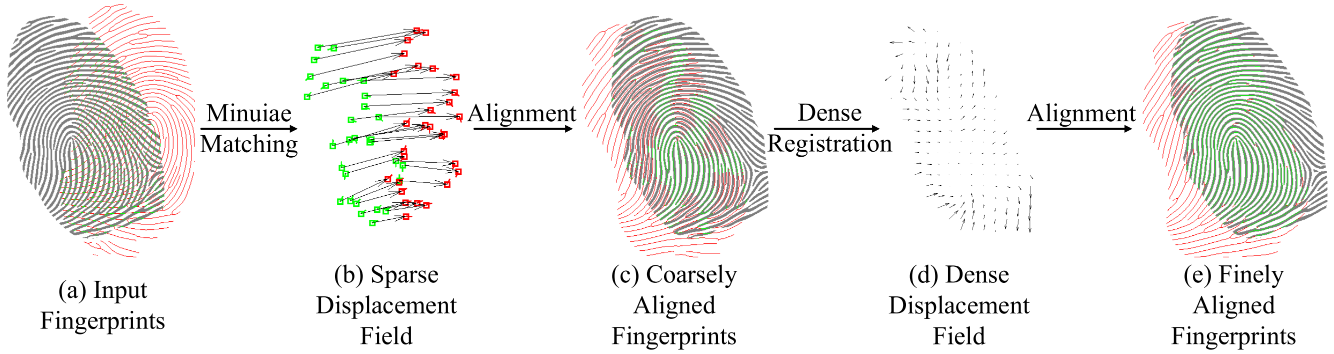


Fig. 2. The whole fingerprint registration algorithm consists of two steps: minutiae-based coarse registration and CNN-based fine registration. Note that the proposed dense registration network estimates a pixel-wise displacement field, but for clarity, a block-wise displacement field is shown here.

recognition problems than handcrafted methods, and has been applied to orientation field estimation [9], [10], minutiae extraction [11], rectification of distorted fingerprints [12], fingerprint matching [13], and fingerprint unwarping [14].

To our knowledge, however, there is no deep learning based method for dense registration of two fingerprints. In this article, we make an attempt to develop an end-to-end convolutional neural network (CNN) for dense registration of fingerprints, in order to deal with the challenges above. Fig. 2 shows the framework of the whole registration algorithm, which consists of minutiae-based coarse registration and CNN-based fine registration. The input fingerprints are first aligned by fitting a spatial transformation model based on matching minutiae pairs, then go through the proposed dense registration network to get a dense displacement field. The proposed network consists of two parts. The first part is a parallel feature network extracting features from two input fingerprints. The second part is a standard encoder-decoder network [15] to generate displacement field.

We apply our fingerprint registration algorithm to fingerprint mosaicking, that is, stitching multiple images from one finger into a larger fingerprint image [16]. The performance of fingerprint mosaicking relies closely on the accuracy of fingerprint registration. If fingerprint registration fails to align ridge patterns of two fingerprints, fingerprint mosaicking will generate false minutiae and missing minutiae in overlapping region. Based on the results of the proposed registration algorithm, we proposed an algorithm for selecting the optimal stitching seam to further improve the quality of mosaicking region.

We evaluate our method by comparing with previous dense registration methods on registration accuracy and matching performances. We run experiments on FVC2004 [17], Tsinghua Distorted Fingerprint (TDF) database [6] and NIST SD27 latent fingerprint database. Experimental results show that our method outperforms previous dense registration methods in accuracy and is faster. The mosaicking method is evaluated according to consistency in minutiae before and after mosaicking, as well as matching experiments on a small fingerprint database with and without mosaicking. Experiments on mosaicked fingerprints further demonstrates the performance of our dense registration algorithm.

A previous work of this article has been published as a conference paper [18]. As a following work of [18], we have made four main improvements:

- **Network structure.** The network in [18] is a local matching network that outputs a single displacement vector from two input fingerprint patches. We upgrade the network to an encoder-decoder structure with smoothing loss that directly outputs a dense displacement field.

The new network captures global information and penalizes complex displacement field, which is an important advantage over [18] to distinguish genuine matching from impostor matching.

- **Generation of training data.** The training data in [18] are selected from FVC2002, which lack distortion and image quality challenges. Also, the training data in [18] consist of two fingerprint local patches and a corresponding displacement vector, which are of small diversity and lack information of larger scale. In this study, the training data are more plentiful and challenging. The displacement fields are extracted from Tsinghua Distorted Fingerprint Video Database [19] to cover various types of fingerprint distortion. The displacement fields are applied on latent and wet fingerprints to generate data of bad image quality and small area.
- **Mosaicking algorithm.** We also develop a fingerprint mosaicking method, which is useful by itself and can reflect the performance of dense registration, and it is not covered in [18].
- **Experiment.** More experiments are conducted to detailedly examine how and why deep learning method exceeds previous dense registration methods, such as registration result on different kinds of fingerprints, matching performance on latent fingerprints, and mosaicking accuracy for the purpose of examining registration accuracy.

The rest parts of this article are organized as follows. Section II reviews related work. Section III proposes our method using end-to-end network. Section IV mainly introduces our fingerprint mosaicking method after dense registration. Section V presents the experimental results of the proposed method versus previous dense registration methods. Finally, Section VI summarizes our work.

## II. Related Work

### A. Elastic Registration of Fingerprints

Early fingerprint registration methods use rigid model to transform fingerprints [4], which cannot align minutiae accurately, not to speak of aligning ridges. Thin-plate spline (TPS) model is introduced by several researchers to deal with elastic fingerprint distortion [3], [5] to solve this problem.

TPS based algorithms mainly use minutiae correspondences to fit transformation models, thus the accuracy of these methods relies on the accuracy of minutiae matching. Although lots of work focus on improving minutiae matching accuracy [4], [20]–[22], the performance of minutiae matching is still limited by distortion and bad image quality. Additionally, minutiae-based matching only provides minutiae pairs, and the fitted TPS model cannot align fingerprint area lacking matching minutiae.

Several methods [23]–[26] further incorporate ridge skeleton features correspondences to register fingerprints. However, these methods are sensitive to ridge skeleton errors, which is very common in bad quality fingerprints. They also lack reasonable constraint for deformation and strong descriptors for establishing accurate correspondences between ridge points which are featureless.

### B. Dense Registration of Fingerprints

Dense registration of fingerprints aims to provide pixel level correspondences between two fingerprints, rather than very sparse minutiae correspondences.

Dense registration of fingerprints is first introduced by [6]. Their method uses image correlation on blocks as similarity measurement, then minimizes the global similarity score to find dense correspondences between two fingerprints. Their method performs better than minutiae-based method, but the computation of image correlation is sensitive to those challenges mentioned in Fig. 1 and is computationally expensive.

Phase demodulation method proposed in [7] transfers the concept of phase demodulation in communication into fingerprint registration. They make use of the fingerprint characteristic that fingerprint patterns look like a 2-D cosine wave, therefore, computing phase shift is equal to computing displacement. But the phase feature they utilized in this method is still a handcraft feature extracted from fingerprint ridge images, and the accuracy of phase feature is bothered in low quality and large distortion area.

## III. Fingerprint Registration

Our dense registration algorithm mainly consists of three steps: (1) initial registration (or minutiae based fingerprint registration) which finds out the mated minutiae pairs from two fingerprints and roughly aligns them by minutiae correspondences; (2) fine registration using an end-to-end network which outputs a dense displacement field from roughly aligned fingerprints; (3) dense registration of the input fingerprint to the reference fingerprint according to the result of step (2).

### A. Initial Registration

This step performs a coarse registration based on matching minutiae pairs, which is also used in previous dense registration methods [6], [7]. In this step, we first use VeriFinger SDK [27] to extract minutiae, and compute similarity scores among all minutiae pairs using MCC minutia descriptor [20]. Then, we use spectral clustering method [28] to find those most probable matching minutiae pairs. Finally, those matching minutiae are used as landmark points to compute a thin plate spline (TPS) model [29], which is then used to align the input fingerprint to the reference fingerprint.

### B. Network for Displacement Field Estimation

As shown by Fig. 3, the proposed network first uses two parallel branches to extract features from two input fingerprints. Then the two features are concatenated and sent into an encoder-decoder to regress the displacement field $\mathcal{D} = (\mathcal{D}_x, \mathcal{D}_y)$ at each pixel. The network is fully convolutional, therefore can be trained and evaluated in an end-to-end manner, and can deal with fingerprint images of arbitrary size.

The training loss consists of two parts: the regression loss between estimated displacement field $\mathcal{D}_{est}$ and ground-truth displacement field $\mathcal{D}$, and the smoothing loss of estimated

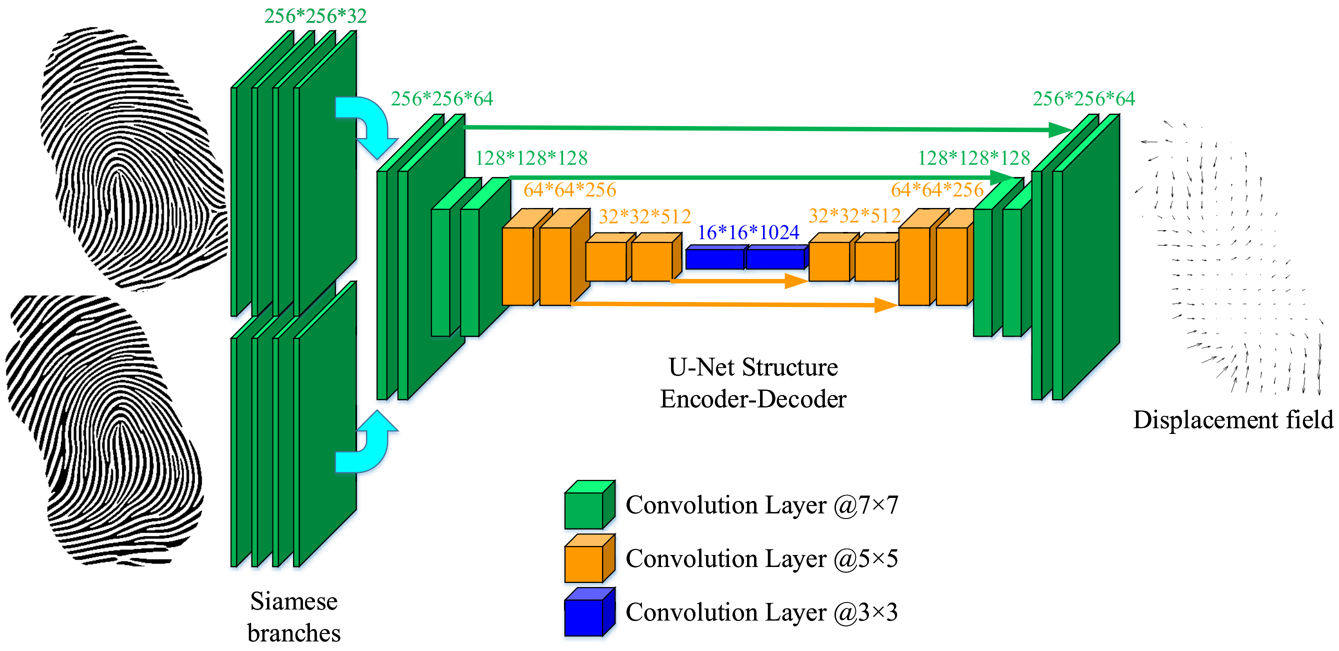


Fig. 3. Architecture of the proposed network, including a siamese network for feature embedding and an encoder-decoder for estimating displacement field. Siamese branches consist of 4 convolution layers. Encoder-decoder includes 4 down blocks, a middle convolution block, and 4 up blocks, and each block has two convolution layers. Layers in different colors imply for different kernel sizes, and layers in different sizes correspond to maxpooling or upsampling.

$\mathcal{D}_{est}$.

$$\mathcal{L} = \mathcal{L}_{est} + \lambda \mathcal{L}_{smo} \tag{1}$$

$$\mathcal{L}_{est} = \sum \|\mathcal{D}_{est} - \mathcal{D}\|_2^2 \tag{2}$$

$$\mathcal{L}_{smo} = \sum_x \|\nabla(\mathcal{D}_{est})\|_2^2 \tag{3}$$

where $x$ denotes a pixel and $\lambda$ is empirically set to 0.8. $\nabla$ is a standard Laplacian filter to compute spatial gradient. Since $\mathcal{D}_{est}$ is two-dimensional, the two components of gradients are computed separately and then the sum of square is the final loss.

$$\|\nabla(\mathcal{D}_{est})\|_2^2 = \left|\nabla(\mathcal{D}_{est}^x)\right|^2 + \left|\nabla(\mathcal{D}_{est}^y)\right|^2 \tag{4}$$

### C. Training Data

*1) Displacement Fields Generation:* For deep learning studies, training data is as important as network structure. To train an end-to-end network in Fig. 3, we need sufficient pairs of mated fingerprints and corresponding dense displacement filed. However, acquiring a large size of ground truth data of displacement fields from challenging mated fingerprints manually is a gigantic task, and is unrealistic.

Therefore, we use Tsinghua Distorted Fingerprint Video Database [19] to generate displacement fields. We select this database for two reasons: 1) A large proportion of fingerprints in this database are of severe distortion, which is beneficial for training a network suitable for large distortion. 2) Although large distortion may cause trouble in fingerprint matching, fingerprint images in adjacent frames are of little translation. By tracing the motion of minutiae between neighboring frame one after another, we can construct reliable minutiae correspondences across all frames even if they are highly distorted. The first and the last frames from one video are used to generate a displacement field, and the displacement field is computed by TPS transform using matching minutiae pairs through minutiae tracking. Therefore, each displacement field is of large distortion and high confidence. We get 320 displacement fields in total from distorted fingerprint videos.

*2) Training Data Building:* For the displacement fields obtained from Tsinghua Distorted Fingerprint Video Database, we do not directly use these data to train a network because 320 pairs of fingerprints of good image quality are clearly not enough from aspects of fingerprint image quality and fingerprint patterns. We build training data from three datasets listed in Table I, i.e. rolled fingerprints, latent fingerprints, and wet fingerprints altogether with obtained displacement fields. Latent fingerprints are provided by local police department, and they are used in order to make our method insensitive to various latent fingerprint qualities. Wet fingerprints were also used to improve our method on wet fingerprint circumstance which is not included in latent fingerprints.

For a fingerprint image $I_i$ in dataset, we use displacement field $\mathcal{D}_j$ to transform and interpolate a transformed fingerprint image $I_{ij} = \mathcal{D}_j(I_i)$. A pair of images $I_i$ and $\mathcal{D}_j(I_i)$ are first binarized by VeriFinger [27] to get enhanced images, then we crop areas of size 256×256 from enhanced images and displacement field. Thereby a set of training data $\{I_1, I_2, \mathcal{D}\}$ is generated.

For latent fingerprints, they are additionally enhanced by FingerNet [11] for better results. We have 10,459 latent fingerprint images and 320 displacement fields, and each latent fingerprint image is combined with 2 random selected displacement fields. Therefore, a total number of 20,918 pairs of training dataset are generated. We do not combine each

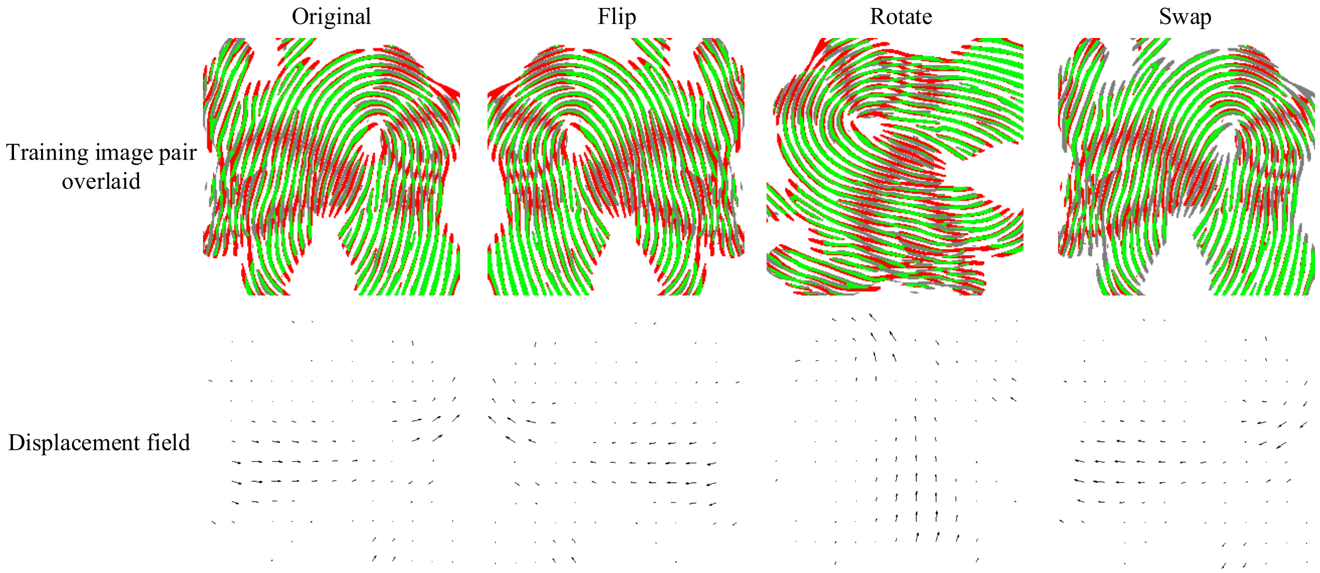


Fig. 4. Examples of three data argumentation methods, including flipping, rotating, and swapping. The first row is training image pairs displayed by overlapping two fingerprint images. Binarized input fingerprints are overlaid on the binarized reference fingerprint (gray line) for visualization. Green pixels indicate regions where input fingerprint ridges align with reference fingerprint ridges, and red pixels indicate non-overlapping regions between input and reference fingerprint ridges. The second row is displacement fields displayed as vector fields.

TABLE I

FINGERPRINT DATABASES USED AS TRAINING DATA

| Databases | Description | Training data |
|---|---|---|
| Tsinghua distorted fingerprint video | largely distorted fingerprints | 320 pairs of displacement fields |
| Rolled fingerprint | 10,459 rolled fingerprints from local police | $10,459 \times 2 = 20,918$ pairs of fingerprints |
| Latent fingerprint | 10,459 latent fingerprints from local police | $10,459 \times 2 = 20,918$ pairs of fingerprints |
| Wet fingerprint | 30 wet fingerprints from FVC2004 DB1/2/3_B | $30 \times 320 = 9,600$ pairs of fingerprints |

fingerprint image with each displacement field to reduce the redundancy of training data. For wet fingerprints, because we only have 30 wet fingerprints from FVC2004 DB1/2/3_B, each wet fingerprint is combined with all 320 displacement fields.

*3) Data Augmentation:* We conduct data augmentation by mirror flipping images, rotating, and swapping the position of $I_1$ and $I_2$, as shown in Fig. 4. When changing fingerprint image patches, their displacements are changed at the same time. The displacement field $\mathcal{D}$ here has size 256×256×2 which is the same as input images. Therefore, flipping and rotating are also operated on $\mathcal{D}$. For clearer explanation, we split $\mathcal{D}$ into $\mathcal{D}_x$ and $\mathcal{D}_y$ in following explanation.

The three augmentation types are:

1) *Mirror flipping:* Augmented $\{I_1, I_2, \mathcal{D}_x, \mathcal{D}_y\}$ at coordinate $(i, j)$ = Original $\{I_1, I_2, -\mathcal{D}_x, \mathcal{D}_y\}$ at coordinate $(-i, j)$, where axis origin is located at the center of image.
2) *Rotate:* Augmented $\{I_1, I_2, \mathcal{D}_x, \mathcal{D}_y\}$ at coordinate $(i, j)$ = Original $\{I_1, I_2, \mathcal{D}_x \cos\theta + \mathcal{D}_y \sin\theta, -\mathcal{D}_x \sin\theta + \mathcal{D}_y \cos\theta\}$ at coordinate $(i \cos\theta + j \sin\theta, -i \sin\theta + j \cos\theta)$, $\theta \in [90°, 180°, 270°]$, where axis origin is located at the center of image.
3) *Swap:* Augmented $\{I_1, I_2, \mathcal{D}_x, \mathcal{D}_y\}$ at coordinate $(i, j)$ = Original $\{I_2, I_1, -\mathcal{D}_x, -\mathcal{D}_y\}$ at coordinate $(i, j)$, where axis origin is located at the center of image.

The three augmentation types can be integrated to enlarge the size of training data. After augmentation, the training set is 16 times bigger, which meets the need of training our network.

### *D. Dense Registration*

The final registration result is a direct nearest neighbor interpolation using the output displacement $\mathcal{D}$ to generate an aligned input fingerprint. Previous dense registration methods [6], [7], [18] use TPS transform to conduct the final step of dense registration because these methods output displacement fields on grid level, size of 20 × 20 pixels for instance. Therefore, fitting a TPS transform is needed to calculate displacements on the rest of pixels.

Our method outputs displacement field directly on pixel level, and the output displacement field is already smoothed because of smoothing loss in training network. So nearest neighbor interpolation can be used in the final step. Comparing with TPS interpolation, nearest neighbor interpolation is much faster.

## IV. FINGERPRINT MOSAICKING

An important application of fingerprint registration is fingerprint mosaicking, which combines a series of smaller fingerprints to reconstruct a larger fingerprint. Comparing

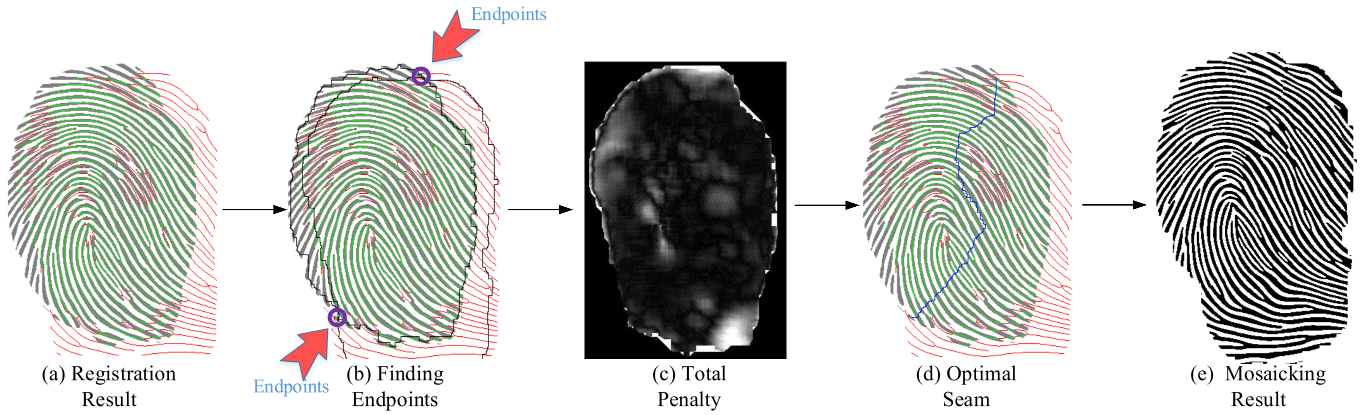


Fig. 5. The flowchart of the proposed fingerprint mosaicking algorithm.

with conventional fingerprint mosaicking method that combines a sequence of fingerprints [30], the proposed fingerprint mosaicking method aims at stitching multiple fingerprints from different acquisition sessions, which is more general and complicated. To combine multiple smaller fingerprints into one larger fingerprint, these smaller fingerprints need to be pre-aligned, and the performances of registration will consequently affect the performances of mosaicking.

In this section, a fingerprint mosaicking method is introduced after applying our registration algorithm. A pair of fingerprints may still have some differences after registration, and a straightforward method based on weighted fusion of fingerprints will lead to mess in badly aligned region [31]. To overcome this problem, we develop an optimal seam selection method to combine two fingerprints and reduce influence from badly registered area. The mosaicking algorithm is detailedly introduced in next subsections. IV-A introduces the proposed mosaicking method, and IV-B talks about the utilization of the proposed mosaicking method on fusing multiple fingerprint images.

### A. Mosaicking of Two Fingerprints

As Fig. 5 illustrates, our fingerprint mosaicking method consists of three steps: 1) find endpoints of the seam to be determined, 2) build an undirected graph based on the differences between two fingerprints, and 3) find the optimal seam and conduct mosaicking.

**Step 1** is quite straightforward: the intersection points of the edges of the two fingerprints are considered as endpoint candidates, as shown in Fig. 5(b). Morphological processing is used to fill holes to get smoothed borders. If the edges of two fingerprints have more than two intersection points, the two farthest intersection points are used as endpoints. If no intersection point is found, which means one fingerprint is inside another fingerprint after pairwise registration, then the larger fingerprint is considered as the mosaicking result, and the smaller fingerprint is left out.

**Step 2** builds an undirected graph using each pixel in the overlapping region as a node, and its 4-connection neighbors as edges. The edges are weighted by the penalties computed from the differences between two fingerprints, as shown in 5(c) where higher intensities implies higher difference values. The differences of two fingerprints, also viewed as the penalty of separation seam, contains three parts: the gray-scale image difference $|I_1(p) - I_2(p)|$, the orientation difference $|O_1(p) - O_2(p)|$ between reference fingerprint and registered input fingerprint, and the penalty by the distance to the edges.

The first two penalties quantify how much the two fingerprints differ, and the third penalty simply prevents the separation seam from being too close to one side, but instead being in the middle and making use of both fingerprints. This border penalty drives separation line away from fingerprint border, which decreases bad influences by border effect during processing fingerprint images. Clearly, the separation seam must be within the overlapping region of two fingerprints, thus we only compute penalties in the overlapping region.

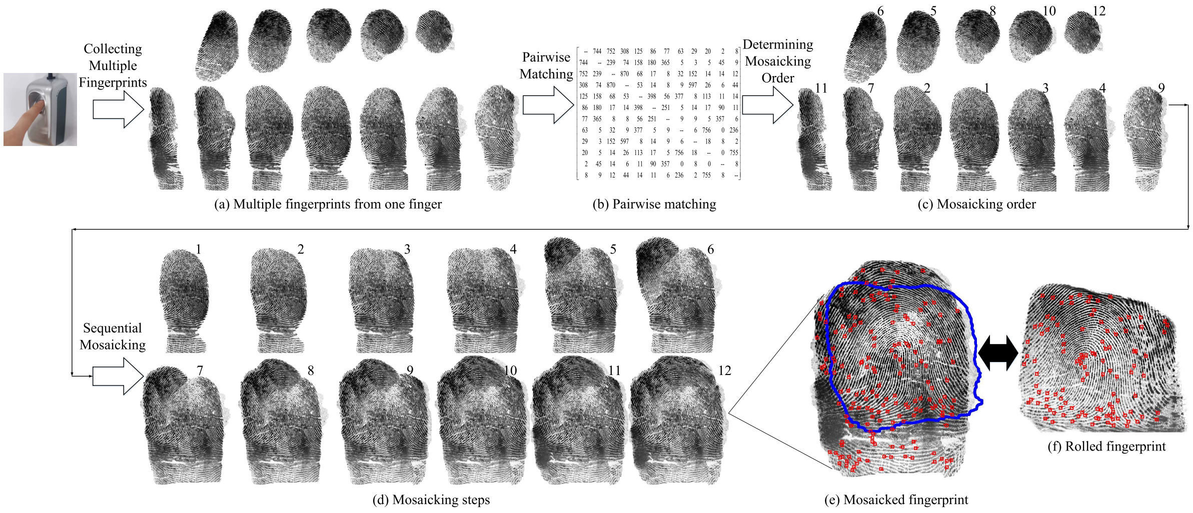
$$\begin{aligned}\mathcal{P}(p) = |I_1(p) - I_2(p)| + \lambda_1 |O_1(p) - O_2(p)| \\ + \lambda_2 \exp(-distance(p, edges)) \qquad (5)\end{aligned}$$

**Step 3** determines the optimal seam after settling endpoints of the possible seam and constructing the graph. To combine reference fingerprint and aligned input fingerprint into a larger fingerprint image, we need to find an optimal separation line. The optimal separation line, or optimal seam, is defined as minimizing the differences between two sides across this seam. Then aligned input and reference fingerprints are cut according to the selected seam, and they are piecing together to get the mosaicked result.

Equation 5 defines a pixel-wise penalty function, and the total penalty of a candidate seam is the sum of penalties of the pixels going through by this seam. Therefore, finding optimal separation line can be viewed as finding an optimal path with minimum total penalty, which is a standard *shortest path* issue and can be solved by *Dijkstra* algorithm.

### B. Mosaicking of Multiple Fingerprints

The above mosaicking algorithm can be extended to combine a series of smaller fingerprints into a larger fingerprint. These fingerprints may be gathered by rolling finger on a fingerprint sensor, or just by pressing finger in different angles on the sensor. By pressing finger with various angles, we can get fingerprints from different parts on finger, as shown in Fig. 6. We collect 12 images to cover nearly all areas on finger



(a) Multiple fingerprints from one finger (b) Pairwise matching (c) Mosaicking order (d) Mosaicking steps (e) Mosaicked fingerprint (f) Rolled fingerprint

Fig. 6. Growing process of multiple fingerprint mosaicking. The last two images are a comparison between the mosaicked full fingerprint and a traditional rolled fingerprint, where the blue outline on the mosaicked fingerprint refers to corresponding rolled fingerprint area. Comparing with the mosaicked fingerprint, a single rolled fingerprint cannot get a full fingerprint due to the special shape of finger, and hence contains much fewer minutiae.

in order to reconstruct a full fingerprint as large as possible. Because of the special shape of finger, traditional rolled fingerprint cannot cover all fingerprint area, especially fingertip area. Meanwhile, gathered multiple smaller fingerprints can fully cover whole fingerprint region. In addition, acquisition of rolled fingerprint requires larger sensors, which are more expensive and much less popular than small area fingerprint sensors.

Before combining fingerprints, we need to know the relations between those fingerprints to determine the order of mosaicking, i.e. which fingerprint should be stitched first. An optimal mosaicking sequence should maximize fingerprint similarities between adjacent fingerprints to reduce probable registration and mosaicking errors. Therefore, we run pairwise fingerprint matching by VeriFinger to compute similarity scores between all fingerprints. The first fingerprint is chosen as the one with the largest total similarity scores with all other fingerprints. The remaining fingerprints are mosaicked onto the first one sequentially in a descending order of their matching scores to it.

Because of the special shape of finger, just rolling finger once cannot cover the whole fingerprint area, especially for fingerprint tip region. Mosaicking multiple fingerprints from different acquisition sessions is needed to obtain a full fingerprint. Fig. 6 also shows each step of mosaicking multiple fingerprints. A new fingerprint is first registered to the mosaicked result of last step, then they are stitched together to make a larger fingerprint. The last two fingerprint images in Fig. 6(e)(f) show mosaicked full fingerprint and a conventional rolled fingerprint for comparison. Comparing with rolled fingerprint, the mosaicked fingerprint has larger area and more minutiae (169 vs. 114), which is beneficial for fingerprint recognition.

## V. Experiment

### *A. Network and Implementation Details*

*1) Network Details:* As shown in Fig. 3, the network consists of two parts: siamese branches for feature embedding and encoder-decoder for displacement estimation. Each siamese network has four convolution layers with ReLU activation and 7×7 kernel size. The encoder-decoder is a U-Net [15] structure with four down-convolution blocks, one normal convolution block, four up-convolution blocks, and a single output convolution layer in order. Each convolution block has two convolution layers. Down-convolution block contains a max-pooling at the end, while up-convolution block has an upsampling layer at the beginning. There are four skip connections from each down-convolution block to the corresponding up-convolution blocks. Detailed parameters of convolution blocks and layers are listed in Table II.

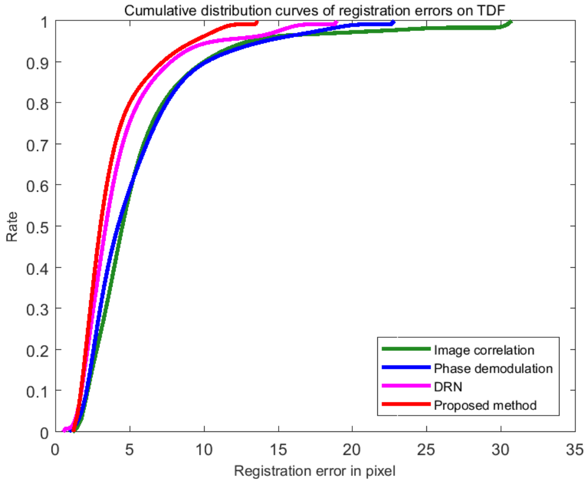


Fig. 7. Cumulative distribution curves of registration errors on TDF database using image correlation (green), phase demodulation (blue), DRN (purple), and the proposed method (red).

*2) Implementation Details:* The network for displacement field estimation in the proposed method was trained on 2 Nvidia 1080Ti GPUs using Adam [32] optimizer with default parameters and mean square error with smoothness loss described in Section III-B. The network was first trained on rolled fingerprint and latent fingerprint database in Table I,

TABLE II
DETAILS OF CONVOLUTION BLOCKS AND LAYERS

| Block or Layer | Kernel | Output Size | Activation |
|---|---|---|---|
| Siamese block | $7 \times 7$ | $M \times N \times 32$ | ReLU |
| Down-convolution block 1 | $7 \times 7$ | $\frac{M}{2} \times \frac{N}{2} \times 64$ | ReLU |
| Down-convolution block 2 | $7 \times 7$ | $\frac{M}{4} \times \frac{N}{4} \times 128$ | ReLU |
| Down-convolution block 3 | $5 \times 5$ | $\frac{M}{8} \times \frac{N}{8} \times 256$ | ReLU |
| Down-convolution block 4 | $5 \times 5$ | $\frac{M}{16} \times \frac{N}{16} \times 512$ | ReLU |
| Convolution Block | $3 \times 3$ | $\frac{M}{16} \times \frac{N}{16} \times 1024$ | ReLU |
| Up-convolution block 4 | $5 \times 5$ | $\frac{M}{8} \times \frac{N}{8} \times 512$ | ReLU |
| Up-convolution block 3 | $5 \times 5$ | $\frac{M}{4} \times \frac{N}{4} \times 256$ | ReLU |
| Up-convolution block 2 | $7 \times 7$ | $\frac{M}{2} \times \frac{N}{2} \times 128$ | ReLU |
| Up-convolution block 1 | $7 \times 7$ | $M \times N \times 64$ | ReLU |
| Convolution Layer | $1 \times 1$ | $M \times N \times 2$ | Linear |

and then fine-tuned on wet fingerprint training data. 20,000 pairs from 20,918 pairs of rolled fingerprints, 20,000 pairs from 20,918 pairs of latent fingerprints, and 9,000 pairs from 9,600 pairs of wet fingerprints were used in training, and the rest pairs were used for validation. Training process including fine-tuning took about 20 hours with batch size 32 on 2 Nvidia 1080Ti GPUs.

For testing, as the network is end-to-end for arbitrary input size, we can directly get a dense displacement field from two input fingerprints. Then the input fingerprint is transformed using nearest neighbor interpolation according to the output displacement field. Finally, matching scores are computed between transformed input fingerprint and reference fingerprint.

*3) Datasets Used in Experiments:* We evaluate the proposed algorithm and other dense registration methods [6], [7], [18] on multiple databases from aspects of registration accuracy, matching accuracy, mosaicking accuracy, and computation efficiency. Table III provides a description of these databases. These databases were captured using different acquisition techniques (optical, capacitive, inking, and latent fingerprint development) and contain flat, rolled, and latent fingerprints. Various challenges such as distortion, low quality, and small area, are present in these databases.

### *B. Registration Accuracy*

We evaluate registration accuracy from two aspects: deviation of labeled corresponding points, and image correlation score. The former is a comparison between registration result with ground truth, while the latter can be tested on large dataset without labeled corresponding points. Finally, we show some good and poor registration examples.

*1) Evaluation by Deviation:* As a quantitative evaluation of registration accuracy, the registration errors are measured by calculating the deviation between labeled point correspondences. For example, for a pair of marked points $p_1$ in image 1 and $p_2$ in image 2, the algorithm translates $p_2$ in image 2 to a new position $\widetilde{p}$ after registration with image 1. Ideally, $\widetilde{p}$ should be equal to $p_1$. But in fact, there exists deviation between $\widetilde{p}$ and $p_1$, and $\|\widetilde{p} - p_1\|_2$ is the registration error.

By calculating and analyzing the registration errors, we can get the cumulative distribution function (CDF) curves of registration errors in pixels on TDF database. Although TDF is involved in generating training data, it's used only for acquiring displacement fields, and the fingerprint images in TDF are not included during training. Additionally, the displacement field are automatically generated from distorted fingerprint video instead of manual labeling. Therefore, this evaluation is fair for different dense registration algorithms.

We use a subset of 120 pairs of fingerprints from TDF with manually marked corresponding points, which are also used for evaluating registration accuracy in previous dense registration methods [6], [7], [18]. As Fig. 7 shows, the CDF curve of our method reaches 1 faster than previous methods. The cut-off error at rate 1 of our method is 13.5 pixels, which is smaller than image correlation (30.7), phase demodulation (22.7), and DRN(18.9). The mean registration errors of three dense registration methods are 5.65, 5.36, 4.34, 3.75 pixels respectively, and our method reaches the minimum registration error.

*2) Evaluation by Image Correlator:* Successful registration reduces the geometric difference between different images from same fingerprint. In order to evaluate the registration accuracy on a larger scale, we conduct registration on the 2,800 pairs of genuine matching fingerprints from FVC2004 DB1_A, and use the correlation coefficient to judge the registration accuracy. During the collection of three datasets of FVC2004, the impressions were required to vary in pressing pressure, skin distortion, and fingerprint quality [2]. Therefore, we can further study the registration performances on different categories of fingerprints. Three types of fingerprints are selected by authors manually, including 1) 61 fingerprints with large distortion, 2) 109 dry fingerprints, and 3) 140 wet fingerprints. The division of the subsets is publicly available.[1] If a pair of genuine matching fingerprints involves one subset, its matching score is included in this subset. Some fingerprint pairs may involve two subsets, and their matching scores are counted into both subsets.

Fig. 8 shows the CDF curves of genuine matching scores using image correlator on different subsets on FVC2004 DB1_A. The CDF curve of genuine matching score is also termed as False Non-Match Rate (FNMR) curve. At a fixed correlation score, a lower CDF value (cumulative probability) by image correlator indicates better registration result as well as lower FNMR. Comparing with DRN [18], the cumulative probability (or FNMR) at correlation score 0.5 by the proposed method is much lower: declined by 32.8% on distorted fingerprints (subset 1), 55.6% on dry fingerprints (subset 2), 23.9% on wet fingerprints (subset 3), and 37.0% on all fingerprints (whole set). Compared to image correlation method [6] and phase demodulation method [7], the proposed method produces even larger reduction in FNMR.

Fig. 8 shows that our method obviously outperforms previous dense registration methods within a wide range of correlation score ($[0.3, 0.8]$). The CDF curves of three dense registration methods only overlap at very low and very high correlation scores. But these two intervals are not important to compare the performances of three dense registration methods

[1] https://github.com/cuiz101417/Dense-Registration

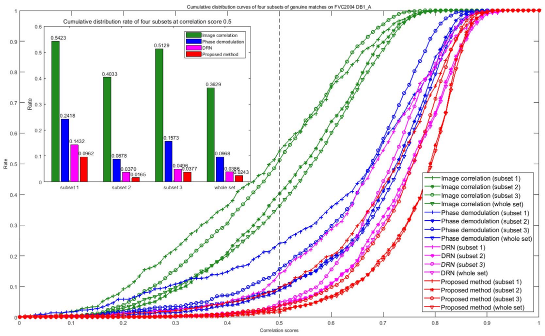


Fig. 8. Cumulative distribution curves by image correlation scores on four subsets of genuine matching pairs on FVC2004 DB1_A. Matching score is computed as image correlation score between fingerprints aligned by each of the four different dense registration algorithms, using image correlation (green), phase demodulation (blue), DRN (purple), and the proposed method (red). The four subsets are distorted fingerprints, dried fingerprints, moistened fingerprints, and all fingerprints, respectively. The subfigure in the upper left area detailedly displays the cumulative probabilities on four subsets by dense registration algorithms at correlation score 0.5.

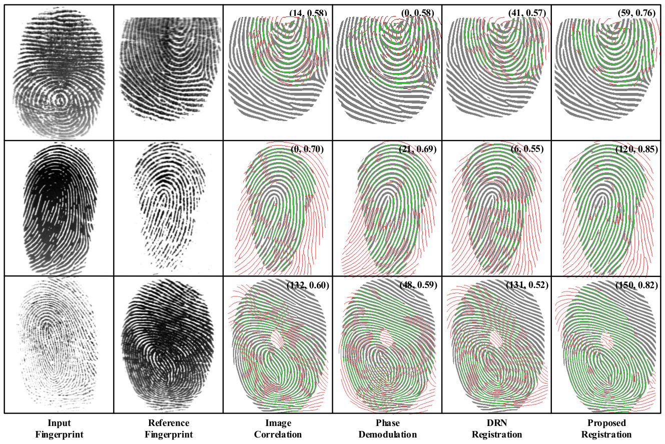


Fig. 9. Registration examples of different dense registration methods for genuine matching fingerprints. The numbers in the brackets are matching scores by VeriFinger and image correlator.

because: 1) A very low correlation score does not make sense because of the existence of very few failures on initial registration that all three dense registration methods will fail; 2) A very high correlation score also makes no sense because registration algorithm cannot eliminate the gray scale difference caused by image noise. That is to say, even if a pair of fingerprints are perfectly registered, their correlation coefficient cannot reach 1. Additionally, Fig. 8 shows that the proposed method successfully improves registration performances on distorted fingerprints and low quality fingerprints, which validates the purpose of this study to improve registration performance on fingerprints with distortion and low quality.

*3) Registration Success and Failure:* Some registration examples on FVC2004 DB1_A by dense registration methods are displayed in Fig. 9. Each row indicates a pair of input and reference fingerprints, as well as their registration results. Four dense registration methods are compared: 1) image correlation, 2) phase demodulation method, 3) DRN, and 4) our method. Registration examples are displayed by overlapping reference fingerprint and aligned input fingerprint in colors to be easily understood. Green lines mark the well-registered areas, and red lines refer to registration failure regions or non-overlapping regions. Numbers inside the brackets are matching scores by VeriFinger matcher [27] and image correlator. Higher scores mean better registration performances, and the scores by image correlator directly reflect the accuracy of dense registration.

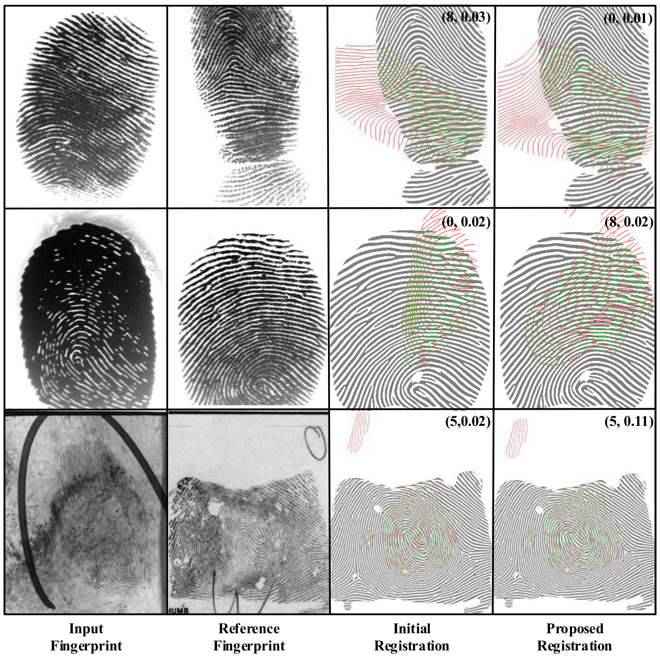


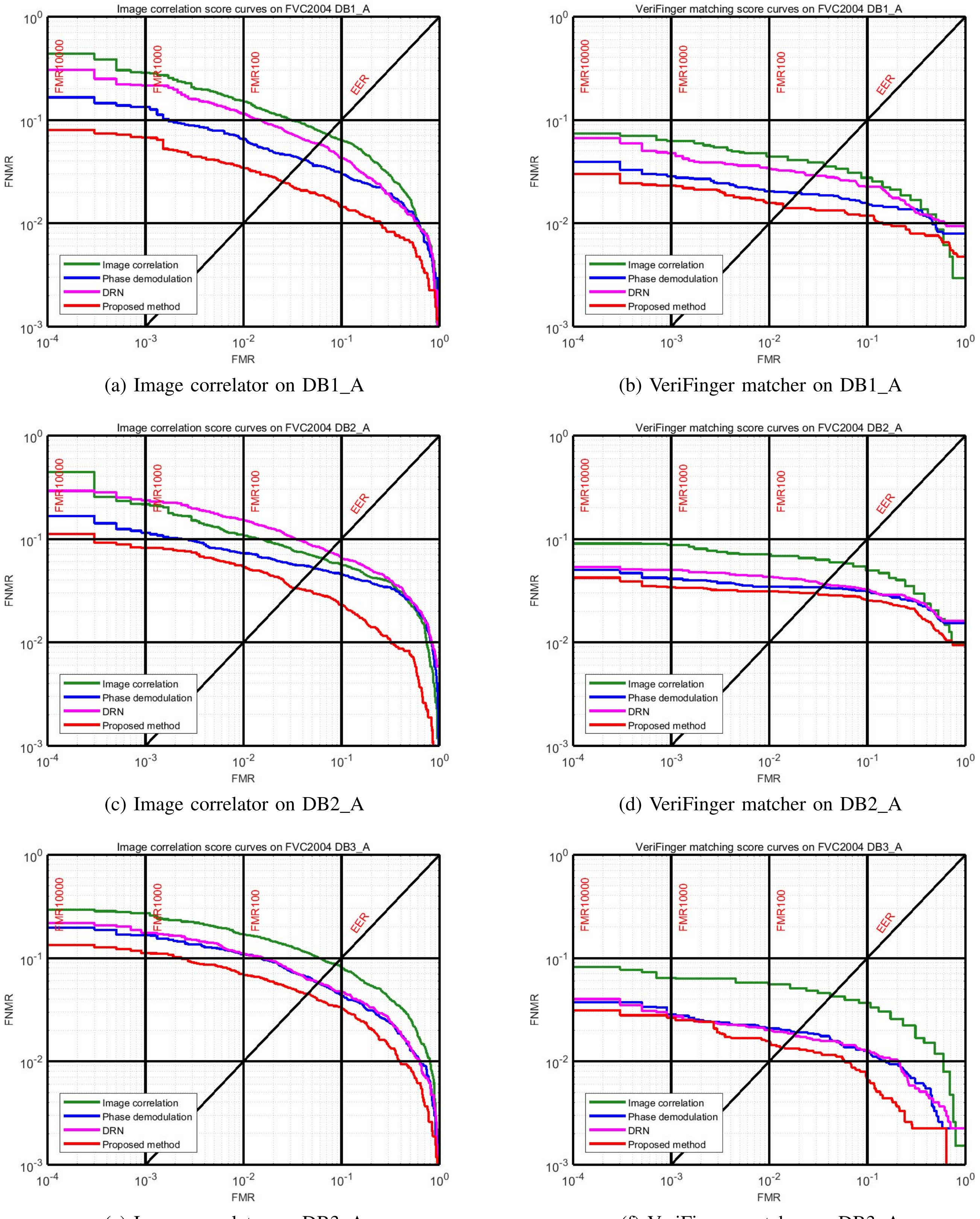
Fig. 10. Bad registration examples by the proposed dense registration method. Due to distortion and small overlapping region (row 1), bad quality and distortion (row 2), and extremely low quality (row 3), minutiae-based initial registration is far from correct, which is beyond the capability of dense registration method.

Those genuine matching examples in Fig. 9 show that the proposed method improves matching results with registration challenges. The first row is a pair of distorted fingerprints, the second row is an example of poor fingerprint quality, and the third row meets difficulties in aligning neighboring ridges in regions outside the fingerprint center because of lacking minutiae and self-similarity of ridge structures. But our method deals with those challenges well.

Meanwhile, as Fig. 10 shows, our method still generates some bad registration results. These results mainly happen in situations where multiple challenges coexist, like small overlapping area, distortion, and low quality. Under these circumstances, the extracted minutiae are of small number or bad quality, and the minutiae-based initial registration result is far away from correct spatial transformation, which is beyond the range that dense registration method can handle. Previous dense registration methods [6], [7], [18] are also not able to solve this problem as they utilize the same minutiae-based initial registration as our method.

### C. Matching Accuracy

Matching experiments are conducted on FVC2004 DB1_A, DB2_A, DB3_A, and NIST SD27 latent fingerprint data-



(a) Image correlator on DB1_A

(b) VeriFinger matcher on DB1_A

(c) Image correlator on DB2_A

(d) VeriFinger matcher on DB2_A

(e) Image correlator on DB3_A

(f) VeriFinger matcher on DB3_A

Fig. 11. DET curves by image correlator and VeriFinger matcher with dense registration algorithms on FVC2004 DB1/2/3_A.

base to show how registration process improves matching accuracy. Same as previous dense registration studies [6], [7], [18], we use VeriFinger and image correlator to compute matching scores. Although there are many fingerprint

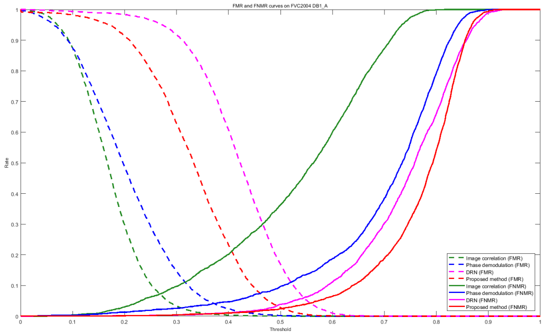


Fig. 12. FMR and FNMR curves versus threshold by image correlator scores on FVC2004 DB1_A using image correlation (green), phase demodulation (blue), DRN (purple), and the proposed method (red).

matching methods [1], we select VeriFinger and image correlator because 1) they stand for minutiae-based matcher and image-based matcher respectively, 2) we are interested in relative increase in matching accuracy caused by registration algorithm, rather than the absolute performance, 3) they are easy to implement or access, and 4) using same matchers allows fair comparison with previous dense registration methods.

Fig. 11 shows the Detection Error Tradeoff (DET) curves on three FVC2004 subsets of dense registration methods: image correlation [6], phase demodulation [7], DRN [18], and the proposed method. Here DRN refers to DRN with single scale in [18]. DRN with multiscale score fusion in [18] combines matching scores from different smoothing scales, thus has better matching performances. Since multiscale DRN outputs multiple registration results, while the other algorithms output a single registration, we report single scale DRN here for a fair comparison of registration performance..

The subfigures (a)(c)(e) in Fig. 11 display the DET curves by image correlator, and (b)(d)(f) display the DET curves by VeriFinger matcher. It can be observed from these plots that our method exceeds other dense registration methods on matching performances.

To further study the matching performance in detail, we draw the False Match Rate (FMR) and False Non-Match Rate (FNMR) curves of image correlator scores of genuine and impostor matching by dense registration methods as Fig. 12. As we can see, the proposed method increases correlation scores of genuine matching, which indicates a better registration performance. Additionally, the curves of impostor matching by our method has the smallest intersection region with genuine matching curves among all dense registration methods, which is beneficial for matching performance.

To further evaluate the proposed method on very low quality fingerprints, we performed matching experiments on NIST SD27. NIST SD27 contains 258 pairs of latent and mated rolled fingerprints, generating 258 × 258 pairs of possible matching. The latent fingerprints are first enhanced by FingerNet [11], then they are registered to rolled fingerprints by different dense registration algorithms. We compare our method with image correlation [6], phase demodulation [7], and DRN [18]. The registered results are evaluated by VeriFinger and image correlator matcher. Fig. 13 shows the results of Cumulative Match Characteristic (CMC) curves. Our method outperforms other methods according to both matchers.

### D. Mosaicking Accuracy

*1) Minutiae Accuracy:* A convenient way of quantifying the performances of fingerprint mosaicking is to evaluate minutiae extraction quality. The quality level of mosaicked fingerprint can be judged by counting ‘fake’ and missing minutiae, i.e. the wrong minutiae caused by discontinuity from mosaicking. If the registration result is poor and misaligns some minutiae or ridges, the mosaicking result would have apparent discontinuity near the seam. Therefore, some fake minutiae appear at the location of ridge disconnection or displacement.

Fig. 14 (c) shows an example with fake minutiae in the overlapping area of two fingerprints. Minutiae in red color refer to the minutiae extracted from the left part fingerprint, and minutiae in green color refer to the minutiae from the right part fingerprint. Minutiae in blue color are extracted from mosaicked fingerprint, and some fake minutiae exist at seam location due to discontinuity. A fake minutia is defined as existing only on mosaicked fingerprint and neither of two fingerprints.

To examine the performance of the proposed fingerprint mosaicking as well as registration, we conduct minutiae extraction accuracy test on 2,800 genuine matching pairs from FVC2004 DB1_A, and counts minutiae extraction errors resulted from mosaicking discontinuity on each pair. For a pair of genuine matching fingerprints, let $I_1$, $I_2$ and $I_M$ denote the input fingerprint, reference fingerprint, and mosaicked fingerprint, $\mathcal{R}_1$ and $\mathcal{R}_2$ are the overlapping region divided by the mosaicking seam. The minutiae extraction error is defined as:

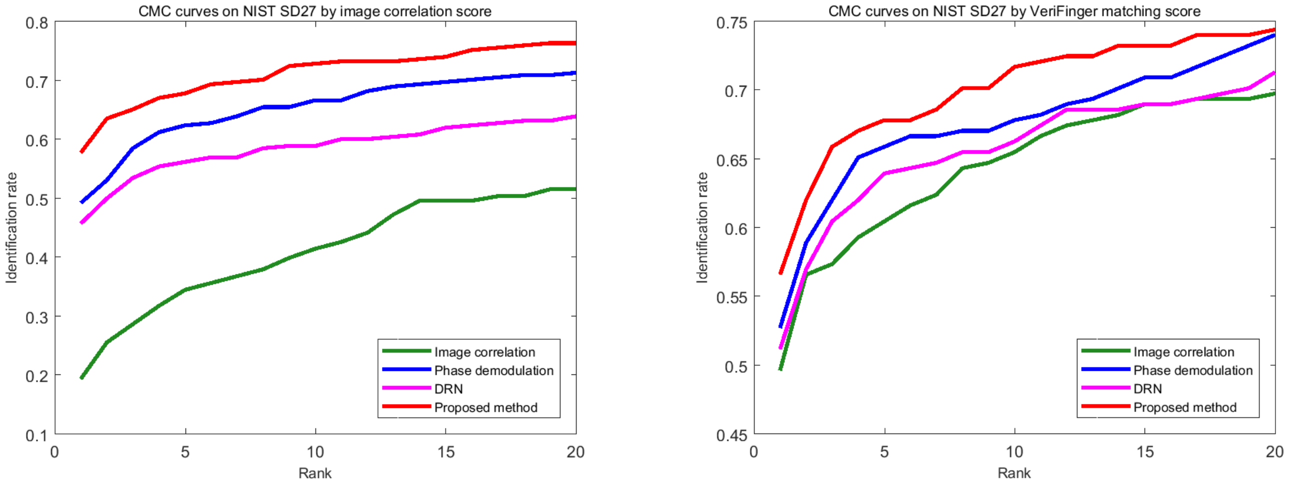
$$e = |\widetilde{n}_1 - n_1| + |\widetilde{n}_2 - n_2|, \tag{6}$$

where

- $n_1$ = minutiae number in $I_1(\mathcal{R}_1)$;
- $n_2$ = minutiae number in $I_2(\mathcal{R}_2)$;
- $\widetilde{n}_1$ = minutiae number in $I_M(\mathcal{R}_1)$;
- $\widetilde{n}_2$ = minutiae number in $I_M(\mathcal{R}_2)$.

Because the fingerprints are already registered, it is reasonable to assume that all ‘real’ minutiae of two fingerprints are aligned and matched. Therefore, directly counting minutiae numbers is enough to judge minutiae errors. In addition, we only conduct minutiae extraction on the region close to the mosaicking seam for efficiency. Because minutiae extraction error is resulted from bad mosaicking result, it only happens near mosaicking seam.

The purpose of evaluating fingerprint mosaicking performances in this study is to judge the performances of dense registration algorithms. Therefore, we apply the proposed fingerprint mosaicking method after dense registration algorithms. We use the registration results of different dense registration methods to conduct fingerprint mosaicking. Fig. 15 shows the cumulative curves of minutiae errors on FVC2004



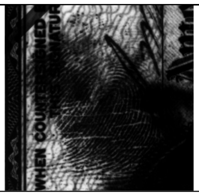
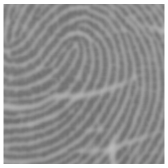
(a) Image correlator on NIST SD27

(b) VeriFinger matcher on NIST SD27

Fig. 13. CMC curves by image correlator and VeriFinger matcher with dense registration algorithms on NIST SD27.

TABLE III

FINGERPRINT DATABASES USED IN EXPERIMENTS

| Database | FVC2004 DB1_A | FVC2004 DB2_A | FVC2004 DB3_A |
|---|---|---|---|
| Image | | | |
| Sensor | Optical | Optical | Thermal sweeping |
| Description | 8 images×100 fingers. Large distortion and various fingerprint quality | 8 images×100 fingers. Large distortion and various fingerprint quality | 8 images×100 fingers. Large distortion and various fingerprint quality |
| Experiments | Registration accuracy<br>Matching accuracy<br>Mosaicking accuracy | Matching accuracy | Matching accuracy |
| Database | NIST SD27 | Tsinghua Distorted Fingerprint Database (TDF) | Small Fingerprint Database |
| Image |  | |  |
| Sensor | Inking, latent | Optical | Capacitive |
| Description | 258 pairs of rolled and mated latent fingerprints of very low quality, small area, and distortion | 320 pairs of highly distorted fingerprints, 120 pairs with manually marked corresponding points | 20 images×48 fingers of size 160×160. |
| Experiments | Matching accuracy | Registration accuracy | Mosaicking accuracy |

DB1_A caused by fingerprint mosaicking by image correlation [6], phase demodulation [7], DRN [18], and the proposed method. Clearly, mosaicking results for our dense registration algorithm are better than previous methods, as more mosaicked fingerprints have no minutiae errors.

*2) Matching Accuracy:* We further conduct matching experiments to evaluate the contribution of mosaicking. 960 small fingerprints of size 160 × 160 from 48 different fingers were used in this experiment. This database is from a cell phone company and is not publicly available. Each finger has 20 impressions from different regions of finger. For each finger, the first 10 fingerprints are used as the template fingerprints, and each of the rest 10 fingerprints is used as the input fingerprint. In the matching experiment without mosaicking, the matching score is computed as the maximum of 10 matching scores between an input fingerprint and 10 template fingerprints. In the matching experiment with mosaicking, the matching score is computed between the input fingerprint and a larger template fingerprint, which is obtained by mosaicking the 10 template fingerprints. An example of mosaicking 10 small fingerprints into a larger template fingerprint is shown in Fig. 16. In total, there are $48 * 1 * 10 = 480$

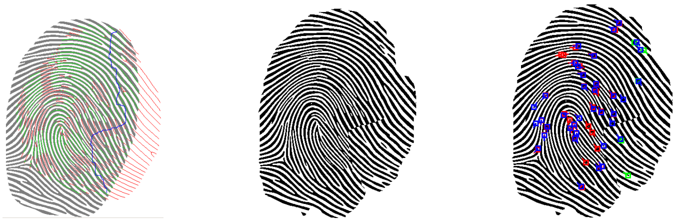

(a) Registration Result (b) Mosaicking Result (c) Minutiae

Fig. 14. Fake minutiae from mosaicking error. The minutiae in red color are extracted from left part fingerprint before mosaicking, minutiae in green color are extracted from right part fingerprint before mosaicking, and minutiae in blue color are extracted after mosaicking. Several blue colored minutiae appear on mosaicking seam none-overlapping with red or green minutiae, and they are classified as 'fake' minutiae.

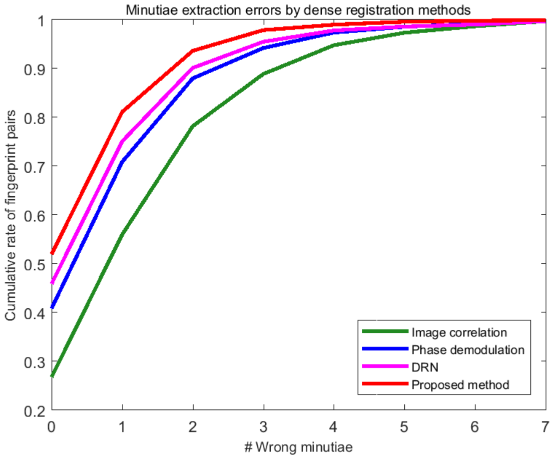


Fig. 15. Minutiae extraction errors on 2,800 genuine matching pairs on FVC2004 DB1_A by dense registration algorithms.

genuine matching pairs and $48*1*10*47 = 22,560$ impostor matching pairs.

Fig. 17 shows the matching results by image correlation score and VeriFinger score. The mosaicked fingerprints have better matching performances than without mosaicking according to both matchers.

### *E. Ablation Study*

*1) Data Augmentation:* Fig. 18 shows the ablation study result of different data augmentation methods. Since three data augmentation approaches are mentioned in Section III-C, there are eight combination strategies in total out of these three methods. For efficiency, the training data to be augmented are only from latent fingerprint database in Table I, and the trained networks are evaluated on FVC2004 DB1_A. Meanwhile, as different augmentation methods vary in magnifications, the training data sizes of different augmentation methods are set to have the same size 20,000 after augmentation. For example, the magnification time of method "Swap+Rotate" is 8, therefore 2,500 pairs of training data are randomly selected for augmentation and training.

TABLE IV

MATCHING ACCURACY OF DIFFERENT DATA AUGMENTATION METHODS ON FVC2004 DB1_A

| Method | EER | FMR100 | FMRZero |
|---|---|---|---|
| No augmentation | 3.94 | 6.54 | 14.79 |
| Swap | 3.96 | 5.61 | 11.39 |
| Flip | 4.93 | 7.46 | 12.61 |
| Rotate | 3.96 | 5.50 | 11.75 |
| Swap+Flip | 4.43 | 6.75 | 10.82 |
| Swap+Rotate | 5.52 | 3.33 | 10.46 |
| Flip+Rotate | 4.43 | 6.96 | 11.14 |
| Swap+Flip+Rotate | 3.46 | 4.50 | 9.54 |

TABLE V

AVERAGE TIME COSTS (IN SECONDS) OF DIFFERENT DENSE REGISTRATION ALGORITHMS FOR PROCESSING A PAIR OF FINGERPRINTS IN FVC2004 DB1_A

| Methods | Image Correlation | Phase Demodulation | DRN | The Proposed Method |
|---|---|---|---|---|
| Time/s | 3 | 1.99 | 1.15 | 0.53 |

As shown in Fig. 18, all augmentation methods success in improving matching accuracy comparing with no augmentation, which testifies the effectiveness of data augmentation methods. These augmentation methods vary a little on matching performances, and generally the order of single augmentation methods is *Swap*, *Rotate*, and *Flip* according to matching performances in Fig. 18. The result by using two augmentation methods is better than using one single method, and using all three methods gets the best result. The detailed error rates of different data augmentation methods are displayed in Table IV.

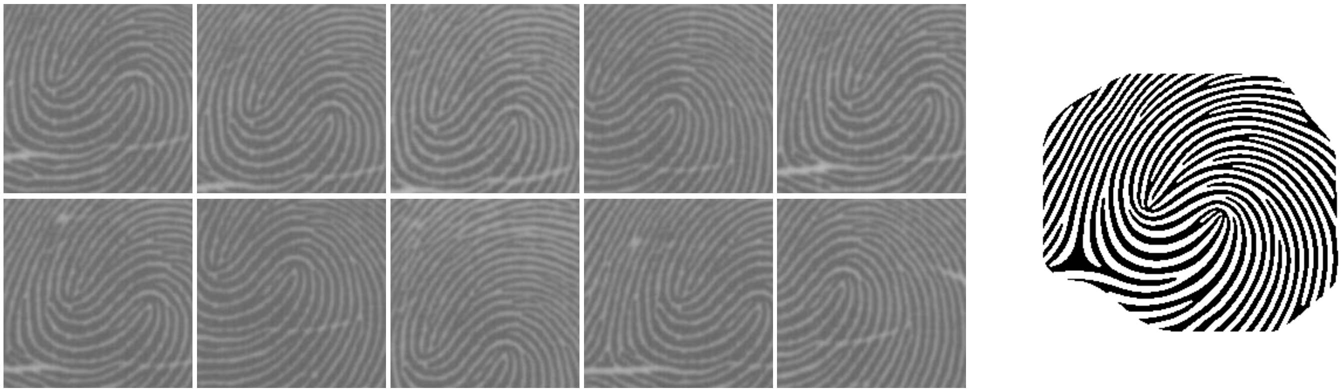
*2) Smoothness Loss:* The effectiveness of smoothness loss is studied and shown in Fig. 19. Three different values of smoothness weight $\lambda$ are compared, $\lambda = 0$ (i.e. no smoothness constraint), and $\lambda = 0.4, 0.8$. Three networks with different $\lambda$ are trained and then evaluated to get registration and matching performances. For efficiency, the networks are trained only on latent fingerprint database in Table I, and evaluated on FVC2004 DB1_A.

As shown in Fig. 19(a), adding smoothness constraint successfully improves genuine matching scores and decreases impostor matching scores. Therefore, it leads to an improvement in matching performance as shown in Fig. 19(b).

### *F. Efficiency*

Table V shows the average time cost to register a single pair of fingerprints by different dense registration algorithms on FVC2004 DB1_A. The first two methods, image correlation [6] and phase demodulation [7], are implemented in C and MATLAB, and tested on a CPU server equipped with Intel Xeon E5-2640 2.5GHz CPU.

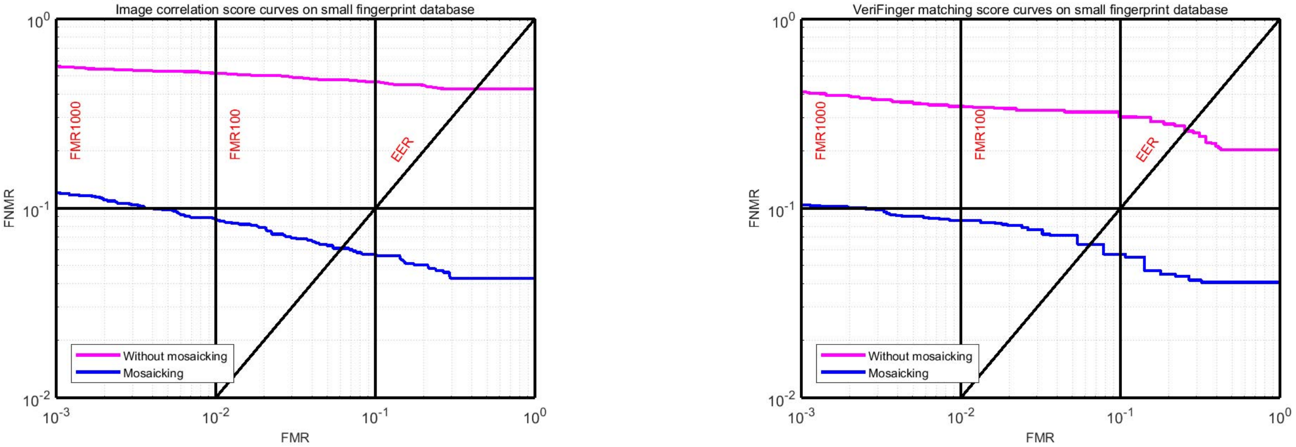
The average time cost of our method is 0.53s, including 0.38s for initial registration on CPU server which is the same as in [6] and [7], 0.15s for the dense registration on a GPU server with two Nvidia 1080Ti, and 2ms for nearest neighbor interpolation also on GPU, which is almost negligible. The dense registration part of our method is much faster than the counterpart of the other two methods. Meanwhile, different from previous dense registration methods that sample displacement field and fit a TPS transformation in the final step,



(a) Small fingerprints used in mosaicking

(b) Mosaicked result

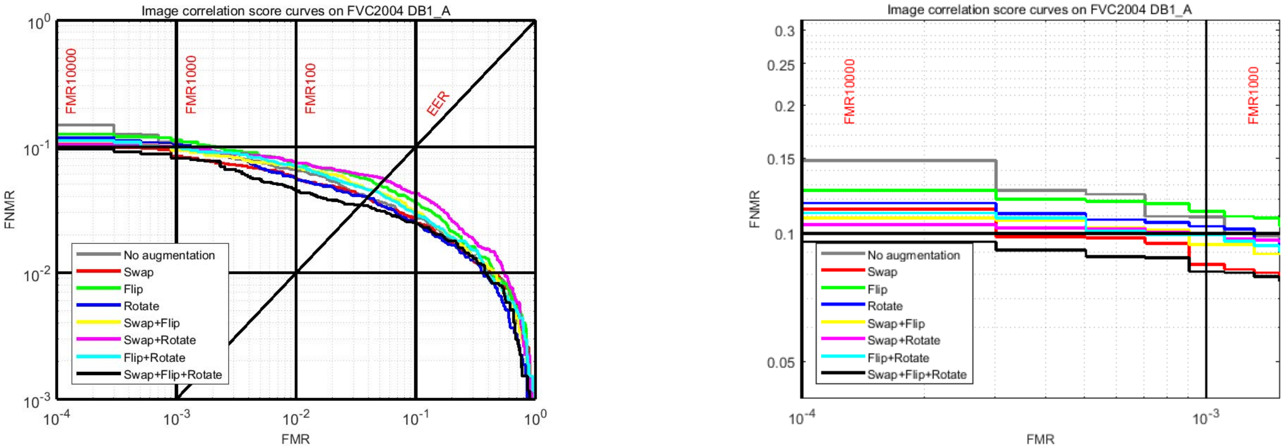
Fig. 16. Example of mosaicking small fingerprints.



(a) DET curve by image correlator

(b) DET curve by VeriFinger

Fig. 17. DET curves by image correlator and VeriFinger matcher on small fingerprint database with and without mosaicking.



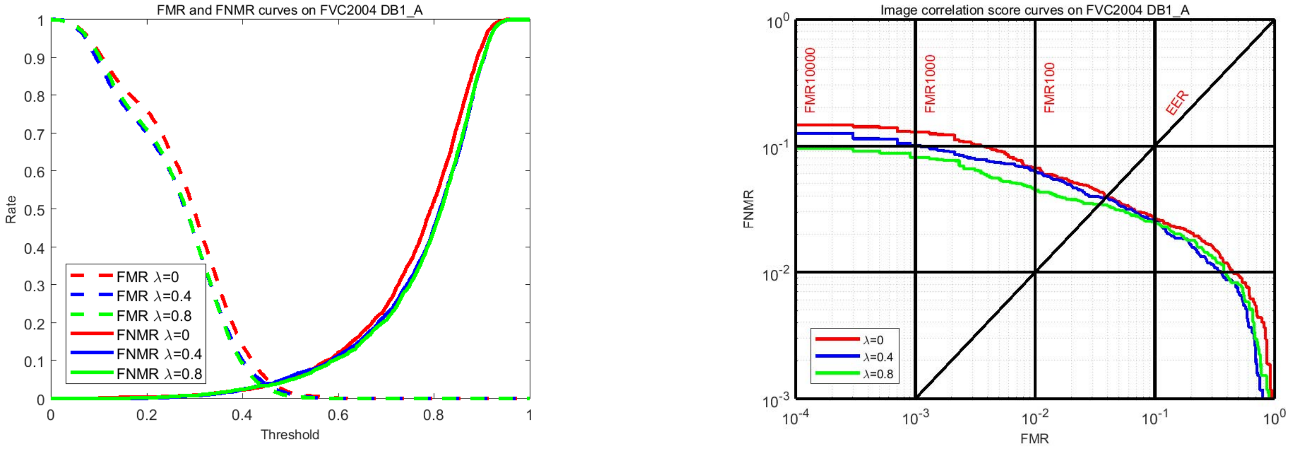
(a) DET curve by image correlator

(b) Enlarged image of (a)

Fig. 18. Studies of data augmentation. All the network models are trained only on latent fingerprint data for efficiency. (a) DET curve by image correlator scores on FVC2004 DB1_A by different data augmentation methods. (b) Zoomed in partial image of (a).

our method directly interpolates to get a transformed image according to displacement field, which is more efficient.

Our method is also faster than [18], which is also implemented on GPU. Comparing with [18], our method is more



(a) FMR and FNMR curves by image correlator scores

(b) DET curve by image correlator

Fig. 19. Studies of smoothness loss hyperparameter $\lambda$. All the network models are trained only on latent fingerprint data for efficiency. (a) FMR and FNMR curves versus threshold by image correlator scores on FVC2004 DB1_A. (b) DET curve by image correlator scores on FVC2004 DB1_A.

complicated on network structure, but costs nearly half the time. It is mainly because that [18] samples displacement field and fits a TPS transformation on final step, but our method directly interpolates to get a transformed image according to displacement field.

## VI. Conclusion

In this article, we propose an end-to-end network to register fingerprints. Input fingerprints are first aligned by minutiae matching, then they are sent into the network to get a pixel-wise dense displacement field. Therefore, the input fingerprints are finely registered by outputted displacement field. We run registration and matching experiments on several databases and prove our registration method outperforms state-of-the-art dense registration method.

Comparing with previous dense registration methods of fingerprints, our method has two main advantages:

1) By collecting and building training data from distorted and latent fingerprints, our method reaches the best registration and matching performances on various types of fingerprints.
2) By utilizing deep learning and training an end-to-end network to directly output displacement field, our method costs much less computation time than previous dense registration methods.

We also develop a fingerprint mosaicking method after registration by computing an optimal seam to stitch two fingerprints. Experiments on mosaicking accuracy and matching accuracy testify our mosaicking method's performance, which is also an evidence of superiority of the proposed registration method over other dense registration methods.

Meanwhile, the bad registration examples in Fig. 10 suggest that our method still needs improvement on registration accuracy. The current algorithm is a fine registration process which makes relatively small adjustment on initial registration result. Therefore, it suffers from very poor initial registration result.

Future work will explore a more powerful algorithm that can handle long-range displacements and is able to correct initial registration errors. The computation speed of the proposed method also needs further improvement to be used in large-scaled fingerprint identification system.

## References

[1] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition*. London, U.K.: Springer, 2009.

[2] R. Cappelli, D. Maio, D. Maltoni, J. L. Wayman, and A. K. Jain, "Performance evaluation of fingerprint verification systems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 1, pp. 3–18, Jan. 2006.

[3] A. M. Bazen and S. H. Gerez, "Fingerprint matching by thin-plate spline modelling of elastic deformations," *Pattern Recognit.*, vol. 36, no. 8, pp. 1859–1867, Aug. 2003.

[4] M. Tico and P. Kuosmanen, "Fingerprint matching using an orientation-based minutia descriptor," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 8, pp. 1009–1014, Aug. 2003.

[5] A. Ross, S. Dass, and A. Jain, "A deformable model for fingerprint matching," *Pattern Recognit.*, vol. 38, no. 1, pp. 95–103, Jan. 2005.

[6] X. Si, J. Feng, B. Yuan, and J. Zhou, "Dense registration of fingerprints," *Pattern Recognit.*, vol. 63, pp. 9252–9260, Mar. 2017.

[7] Z. Cui, J. Feng, S. Li, J. Lu, and J. Zhou, "2-D phase demodulation for deformable fingerprint registration," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 12, pp. 3153–3165, Dec. 2018.

[8] D. C. Ghiglia and M. D. Pritt, *Two-Dimensional Phase Unwrapping: Theory, Algorithms, and Software*. Hoboken, NJ, USA: Wiley, 1998.

[9] K. Cao and A. K. Jain, "Latent orientation field estimation via convolutional neural network," in *Proc. Int. Conf. Biometrics (ICB)*, May 2015, pp. 349–356.

[10] P. Schuch, S.-D. Schulz, and C. Busch, "Deep expectation for estimation of fingerprint orientation fields," in *Proc. IEEE Int. Joint Conf. Biometrics (IJCB)*, Oct. 2017, pp. 185–190.

[11] Y. Tang, F. Gao, J. Feng, and Y. Liu, "FingerNet: An unified deep network for fingerprint minutiae extraction," in *Proc. IEEE Int. Joint Conf. Biometrics (IJCB)*, Oct. 2017, pp. 108–116.

[12] A. Dabouei, H. Kazemi, S. M. Iranmanesh, J. Dawson, and N. M. Nasrabadi, "Fingerprint distortion rectification using deep convolutional neural networks," in *Proc. Int. Conf. Biometrics (ICB)*, Feb. 2018, pp. 1–8.

[13] K. Cao and A. K. Jain, "Automated latent fingerprint recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 4, pp. 788–800, Apr. 2019.

[14] A. Dabouei, S. Soleymani, J. Dawson, and N. M. Nasrabadi, "Deep contactless fingerprint unwarping," in *Proc. Int. Conf. Biometrics (ICB)*, 2019, pp. 1–8.

[15] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent. (MICCAI)*, 2015, pp. 234–241.

[16] A. Jain and A. Ross, "Fingerprint mosaicking," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, May 2002, p. IV-4064.

[17] *FVC2004: The Third International Fingerprint Verification Competition*. Accessed: Aug. 24, 2020. [Online]. Available: http://bias.csr.unibo.it/fvc2004/

[18] Z. Cui, J. Feng, and J. Zhou, "Dense fingerprint registration via displacement regression network," in *Proc. Int. Conf. Biometrics (ICB)*, Jun. 2019, pp. 1–8.

[19] X. Si, J. Feng, J. Zhou, and Y. Luo, "Detection and rectification of distorted fingerprints," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 555–568, Mar. 2015.

[20] R. Cappelli, M. Ferrara, and D. Maltoni, "Minutia cylinder-code: A new representation and matching technique for fingerprint recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 12, pp. 2128–2141, Dec. 2010.

[21] X. Cheng, S. Tulyakov, and V. Govindaraju, "Minutiae-based matching state model for combinations in fingerprint matching system," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2013, pp. 92–97.

[22] A. A. Paulino, J. Feng, and A. K. Jain, "Latent fingerprint matching using descriptor-based Hough transform," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 1, pp. 31–45, Jan. 2013.

[23] A. Ross, S. C. Dass, and A. K. Jain, "Fingerprint warping using ridge curve correspondences," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 1, pp. 19–30, Jan. 2006.

[24] K. Choi, H. Choi, S. Lee, and J. Kim, "Fingerprint image mosaicking by recursive ridge mapping," *IEEE Trans. Syst. Man, Cybern. B, Cybern.*, vol. 37, no. 5, pp. 1191–1203, Oct. 2007.

[25] H. Choi, K. Choi, and J. Kim, "Fingerprint matching incorporating ridge features with minutiae," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 2, pp. 338–345, Jun. 2011.

[26] S. Lan, Z. Guo, and J. You, "A non-rigid registration method with application to distorted fingerprint matching," *Pattern Recognit.*, vol. 95, pp. 48–57, Nov. 2019.

[27] Neurotechnology Inc. *VeriFinger SDK 6.2*. Accessed: Aug. 24, 2020. [Online]. Available: http://www.neurotechnology.com

[28] M. Leordeanu and M. Hebert, "A spectral technique for correspondence problems using pairwise constraints," in *Proc. 10th IEEE Int. Conf. Comput. Vis. (ICCV)*, vol. 1, Oct. 2005, pp. 1482–1489.

[29] F. L. Bookstein, "Principal warps: Thin-plate splines and the decomposition of deformations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 6, pp. 567–585, Jun. 1989.

[30] N. K. Ratha, J. H. Connell, and R. M. Bolle, "Image mosaicing for rolled fingerprint construction," in *Proc. Int. Conf. Pattern Recognit. (ICPR)*, vol. 2, 1998, pp. 1651–1653.

[31] K. Choi, H. Choi, and J. Kim, "Fingerprint mosaicking by rolling and sliding," in *Proc. Int. Conf. Audio Video-Based Biometric Person Authentication (AVBPA)*, 2005, pp. 260–269.

[32] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, vol. 5, 2015, pp. 1–15.

**Zhe Cui** (Graduate Student Member, IEEE) received the B.S. degree from the Department of Physics, Tsinghua University, Beijing, China, in 2014, and the M.Eng. degree from the Department of Automation, Tsinghua University, in 2017, where he is currently pursuing the Ph.D. degree with the Department of Automation. His research interests include fingerprint recognition, computer vision, and pattern recognition.

**Jianjiang Feng** (Member, IEEE) received the B.Eng. and Ph.D. degrees from the School of Telecommunication Engineering, Beijing University of Posts and Telecommunications, China, in 2000 and 2007, respectively. From 2008 to 2009, he was a Post-Doctoral Researcher with the PRIP Laboratory, Michigan State University. He is currently an Associate Professor with the Department of Automation, Tsinghua University, Beijing. His research interests include fingerprint recognition and computer vision. He is an Associate Editor of the *Image and Vision Computing*.

**Jie Zhou** (Senior Member, IEEE) received the B.S. and M.S. degrees from the Department of Mathematics, Nankai University, Tianjin, China, in 1990 and 1992, respectively, and the Ph.D. degree from the Institute of Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology, Wuhan, China, in 1995. From 1995 to 1997, he served as a Post-Doctoral Fellow with the Department of Automation, Tsinghua University, Beijing, China. Since 2003, he has been a Full Professor with the Department of Automation, Tsinghua University. In recent years, he has authored over 100 articles in peer-reviewed journals and conferences. Among them, over 40 articles have been published in top journals and conferences, such as the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE (PAMI), the TRANSACTIONS ON IMAGE PROCESSING (TIP), and CVPR. His current research interests include computer vision, pattern recognition, and image processing. He received the National Outstanding Youth Foundation of China Award. He is an Associate Editor of the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, the *International Journal of Robotics and Automation*, and two other journals.