Estimating Fingerprint Pose via Dense Voting

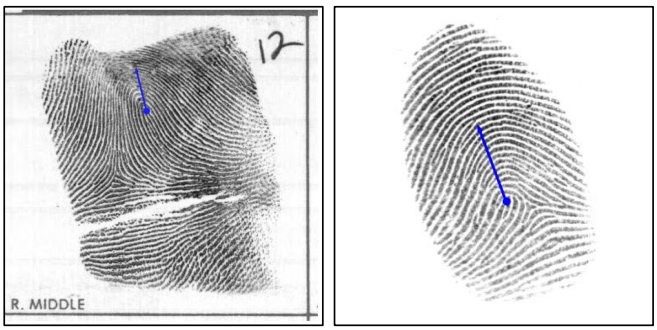
Yongjie Duan, Jianjiang Feng, *Member, IEEE*, Jiwen Lu, *Senior Member, IEEE*, and Jie Zhou, *Senior Member, IEEE*

***Abstract*— Aligning fingerprint images to a unified coordinate system defined by fingerprint pose is beneficial for fast and accurate fingerprint matching. Due to poor ridge quality and partial observations, however, performance of the state-of-the-art fingerprint pose estimation algorithms remains unsatisfactory. In this study, we propose to fuse voting strategy and deep network to estimate fingerprint center and direction. Rather than regressing them directly, we predict dense offset maps and vote for the final estimation. Experimental results on ten fingerprint datasets with over 60K fingerprints show that (1) highly consistent fingerprint pose estimations are obtained across different impressions of the same finger, (2) performance of fingerprint indexing and verification is further improved thanks to more accurate fingerprint pose estimation, and (3) the proposed approach is more robust to sensing technologies (optical, capacitive, inking, and direct imaging) and impression types (rolled, plain, latent, and contactless).**

***Index Terms*— Fingerprint recognition, fingerprint pose estimation, dense voting, deep neural network, fingerprint indexing, fingerprint verification.**

I. Introduction

POSE estimation, usually achieved by landmarks detection for biometrics like face and iris, plays a crucial role in biometric recognition [1], [2]. Fixed-length feature representations can be extracted from aligned facial or iris images, thus facilitating fast and accurate biometric recognition on large scale datasets. Over the past few decades, fingerprint has been one of the most important and dominant biometric identifiers in various applications, including national border control, mobile payment, and crime investigation [3]. However, different from face and iris, fingerprints are typically encoded as a set of unordered minutiae with varying numbers, which makes fingerprint matching time-consuming and lack of global constraint. Apart from estimating fingerprint poses for the widely used contact-based fingerprints, fingerprint poses of contactless fingerprints have also attracted increasing attentions [4], [5], [6], [7]. Suppose that accurate fingerprint pose is known, all fingerprints can be aligned to a unified coordinate system, then the search space for minutiae matching is narrowed down and falsely matched minutiae pairs can be rejected owing to location and direction constraints, thus improving the efficiency and accuracy of fingerprint indexing and verification [8], [9], [10], [11], [12], [13], [14], [15] consequently. Besides, extracting fixed-length fingerprint descriptor can also benefit from the fingerprint alignment [12], [15], which reduces variations of impressions from the same finger. Furthermore, template encryption techniques [16], [17] can be further prompted thanks to the reliable fingerprint alignment.



(a) (b)

Fig. 1. Examples of (a) rolled and (b) plain fingerprint poses. The blue solid circle denotes the location of fingerprint center and the blue arrow indicates the fingerprint direction.

Despite the significant potential value of fingerprint pose, estimating fingerprint pose on various types of fingerprints remains a challenging task. Fingerprint pose is defined by fingerprint center (x, y) and direction θ (as shown in Fig. 1). In contrast to other biometrics, such as face whose pose is determined by key-points with explicit physical definitions, stable anatomical landmarks are difficult to define on fingerprints. Considering the shape of fingerprint area, distribution of ridge orientation field, and presence of singular points, various special fingerprint key-points, such as foreground barycenter [18], singularities [19], [20], maximum curvature points [21], [22], and focal points [23], [24], [25], have been proposed as fingerprint center, while ignoring the fingerprint direction information and being sensitive to ridge quality. Area around fingerprint center was also regarded as an object to be detected based on handcraft [9] and deep features [11], [13]. However, these approaches still cannot solve the problems of incomplete finger area, especially when the area relevant to pose is missing or very noisy. Voting-based methods, such as generalized Hough transform [14], [26], were then proposed to determine fingerprint center and direction based on the maximum voting score. While handcraft features, whose representation ability is limited, were utilized in such methods.

Manuscript received 17 September 2022; revised 26 February 2023; accepted 3 April 2023. Date of publication 12 April 2023; date of current version 24 April 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 61976121. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Walter J. Scheirer. *(Corresponding author: Jianjiang Feng.)*

The authors are with the Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: dyj17@mails.tsinghua.edu.cn; jfeng@tsinghua.edu.cn; lujiwen@tsinghua.edu.cn; jzhou@tsinghua.edu.cn).

This article has supplementary downloadable material available at https://doi.org/10.1109/TIFS.2023.3266625, provided by the authors.

Digital Object Identifier 10.1109/TIFS.2023.3266625

In this paper, we proposed a dense voting framework to integrate the strength of voting strategy and deep learning. Different from previous works which regress to single fingerprint pose output directly, in this paper, pose is estimated based on voting strategy by estimating dense maps, i.e. direction-irrelevant and direction-relevant offset maps. Besides, segmentation probability map is estimated to determine which area is responsible for voting. Considering the characteristic of fingerprint direction, an additional attention map is also predicted to determine which area is important for direction estimation. In an integrated way, a deep network is proposed to extract more robust and discriminative features to estimate dense maps, thus achieving better pose estimation performance, especially for fingerprints with poor ridge quality and partial visible area. The codes will be available for non-commercial purposes[1].

Experimental results reveal that the proposed method outperforms the state-of-the-art (SOTA) algorithms on multiple databases containing different types of fingerprint impressions, including rolled (NIST SD4, NIST SD14), plain (FVC2000 DB2A & DB3A, FVC2002 DB1A, FVC2004 DB1A), latent (NIST SD27, Hisign latent), contactless (PolyU [27]), and our newly collected fingerprint dataset which consists of fingerprints produced with diverse 3D finger rotations (Diverse Poses Fingerprint dataset, DPF). The proposed fingerprint pose estimation algorithm was evaluated by the alignment accuracy, fingerprint indexing with pose constraints, and fingerprint verification performance with fingerprint pose alignment. All experimental results demonstrate the superior performance of our proposed approach compared with other methods.

The main contributions of our study can be summarized as:

- Rather than regressing to single fingerprint pose output directly, we proposed to fuse voting strategy and deep network to estimate fingerprint poses on various fingerprint types, which is especially important for fingerprints with low ridge quality and incomplete area;
- We proposed to estimate fingerprint pose by predicting dense maps and voting integration, and developed a direction attention module to focus on the region that is valuable and important for more robust and accurate fingerprint direction estimation;
- The evaluation experiments and datasets are more comprehensive than previous studies. Extensive fingerprint databases were utilized for evaluation, including rolled, plain, latent, and contactless fingerprints. Given the estimated fingerprint poses using our method, more robust and accurate performance on deviations between mated minutiae pairs, fingerprint indexing, and fingerprint verification was achieved, compared to existing SOTA methods. We also evaluated the proposed method on our collected dataset consisting of plain fingerprints with diverse finger poses, which is very challenging for fingerprint pose estimation, and experimental results on this dataset also demonstrate the superiority of our method.

The rest of paper is organized as follows: section II reviewed published works related to fingerprint pose estimation. Then the proposed dense voting method and its implementation details are introduced in section III and IV, respectively. After that section V shows the experimental results and discussions. Finally, our work is summarized in section VI.

[1]https://github.com/keyunj/Fingerprint-2DPose-Dense-Voting.git

II. Related Works

A. Fingerprint Pose Definition

As mentioned above, there is not an explicit definition of fingerprint pose followed by all research due to the lack of anatomical landmarks in fingerprints compared with other biometrics. Several different definitions of fingerprint pose were proposed in previous research. For rolled or plain fingerprint that is complete and presents high ridge quality, barycenter of the foreground area was considered as fingerprint center [18], and fingerprint direction was defined based on the direction of left and right impression boundaries [28]. Obviously, such definitions are sensitive to the shape of fingerprint impressions which relies on ridge quality, and inconsistent results may be obtained across fingerprints with different visible area.

Considering the stability of singularities (core or delta) in most fingerprint patterns, singular points were selected to assist fingerprint registration [19], while not working well for fingerprints without singularities, such as arch type. Instead, based on the pattern of ridge orientation field, points with maximum curvature [21], [22] or distinctive structure [28], [29] were regarded as fingerprint center. Similarly, crossing point of straight lines normal to ridges, also named as focal point, was referred as fingerprint center in [23], [24], and [25]. However, these lines do not always cross at a single point, thus multiple focal points may be detected. Besides, fingerprint poses based on such pose definitions may also not be consistent across different impressions which is not suitable for assisting following fingerprint indexing or verification tasks.

Yang et al. [26] proposed a fingerprint pose definition in an integrated way based on multiple features, including singularities, maximum curvature points, ridge orientation field, and knuckles. They defined fingerprint direction in perpendicular to the ridge orientation around knuckles. Fingerprint center was defined based on different numbers of singular points according to fingerprint patterns, and all singularities were considered. However, there might be confusion in estimating fingerprint pose when singular points, especially for delta points, cannot be detected or are partially detected, which is common in plain and latent fingerprints (as shown in Fig. 2). Si et al. [20] proposed to utilize only the northernmost core point (for fingerprints with singularities) or maximum curvature point (for fingerprints without singularities) as fingerprint center and adopted the same direction definition as [26]. They found that based on this pose definition, ridge orientation field could be separated into two regions: one containing similar ridge orientations for various pattern types while the other very different, which helps to establish a statistical model for distortion rectification. Such definition is shown to be appropriate for consistent and robust pose definition across different fingerprints or the same fingerprint with different visible area. Therefore, we chose to use it in this paper.

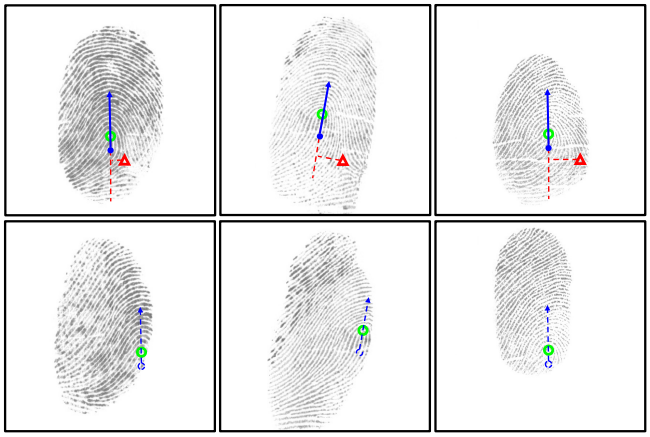


Fig. 2. Examples of fingerprint pose definition in [26] for fingerprint with different visible area. The green circle denotes core singular point, and the red triangle denotes delta singular point. Then the fingerprint pose based on these singularities is represented in blue solid circle and arrow. The first row shows fingerprints with all singularities visible; while the second row shows the corresponding fingerprints with incomplete area where delta singularities are invisible, and the fingerprint pose in blue dotted circle and arrow is directly copied from the first row. It is challenging and confusing to estimate fingerprint pose on partial fingerprints using such fingerprint pose definition.

B. Fingerprint Pose Estimation

Existing fingerprint pose estimation approaches can be classified into three categories, including point detection based, region detection based, and voting based.

1) Point Detection Based: Since distinctive points are utilized in various fingerprint pose definitions, extracting these points from fingerprint is a natural solution. Apart from calculating barycenter of fingerprint foreground [18] which is sensitive to ridge quality and impression shape, ridge orientation field is often employed. Based on the ridge orientation field, singularities [19], maximum curvature points [21], [22], or reference points [29] were detected and referred as fingerprint center. Meanwhile, focal points, which can be calculated as the crossing points of normal lines to ridges, were proposed in several studies [23], [24], [25]. Benefiting from the property of orientation field, such methods can alleviate image noise problem, while it remains challenging when the area around fingerprint center or singular point is incomplete or missing.

2) Region Detection Based: Fingerprint pose estimation can be viewed as an object detection problem where the location and direction of the object (finger) need to be estimated. The whole fingerprint impression or the area around fingerprint center is under consideration in such algorithms. Su et al. [9] trained a set of support vector machine classifiers with different directions, and utilized the histogram of ridge orientations within a fixed-size window as features. Then the center and direction of fingerprint were determined by detecting the maximum response based on overlapped sliding window strategy. Yin et al. [30] utilized the whole ridge orientation field extracted from aligned fingerprints with good quality to create a large orientation dictionary, and exhaustive searching on this orientation dictionary was then performed to estimate fingerprint pose. Ouyang et al. [11] utilized faster R-CNN [31] to determine the location of fingerprint center area and converted fingerprint direction prediction to classification, making the estimation results more accurate and robust compared with handcraft features used in prior works [9], [26]. Engelsma et al. [12] proposed a fixed-length fingerprint descriptor from the aligned fingerprints which is achieved using spatial transformer networks (STN). However, its fingerprint alignment was neither supervised by annotated ground truth annotations, nor evaluated the performance of fingerprint pose estimation. And Yin et al. [13] proposed a multi-task approach to predict fingerprint pose and singular points at the same time inspired by the strong correlation between fingerprint pose definition and singularities. However, singularities might not be helpful when singular points are invisible in many plain or latent fingerprints (as shown in Fig. 2). And the method predicted fingerprint pose by regressing to single value (i.e., three neurons for fingerprint center and direction) directly, making it less reliable to predict poses on fingerprints with low quality or partial area.

3) Voting Based: Rather than estimating fingerprint pose directly, Yang et al. [26] proposed to predict based on Hough voting strategy. They utilized ridge orientations within local patches (96×96 pixels) as features to learn a set of spatial location specified prototype orientation patch dictionaries, and fingerprint center was voted by looking up the learnt localized patch dictionaries. And fingerprint direction was estimated by rotating the original input for several discrete angles to find the maximum Hough voting response. Apart from calculating ridge orientations within a smaller patch (40×40 pixels), Gu et al. [14] also extracted ridge periods as patch features. Fingerprint center was first estimated using Hough forest based on these handcraft features, and fingerprint direction was then determined using a support vector regression method after moving fingerprint to its origin based on the estimated fingerprint center. Benefiting from utilizing voting strategy in these methods, problems of incomplete fingerprints can be mitigated. However, only handcraft features were utilized whose representation ability can be further improved by introducing deep learning. Besides, fingerprint direction was always regressed or classified directly in prior works, where the estimation performance may be limited on fingerprints with low ridge quality or incomplete area.

III. Proposed Method

In this paper, we proposed to estimate fingerprint pose based on dense voting strategy. A deep network was proposed to predict dense maps, i.e. direction-irrelevant and direction-relevant offset maps, segmentation probability map, and direction attention map, and then they were integrated to vote for fingerprint pose. Different from previous works that regress fingerprint pose value directly [9], [11], [12], [13], [14], [26], fingerprint pose is predicted based on dense voting strategy in this study, thus improving algorithm accuracy and robustness on fingerprints with diverse shapes and visible area. The schematic illustration of our method is shown in Fig. 3.

A. Fingerprint Pose Definition

We followed the fingerprint pose definition in [20], which was also adopted in [14] and [15], to make fingerprint pose

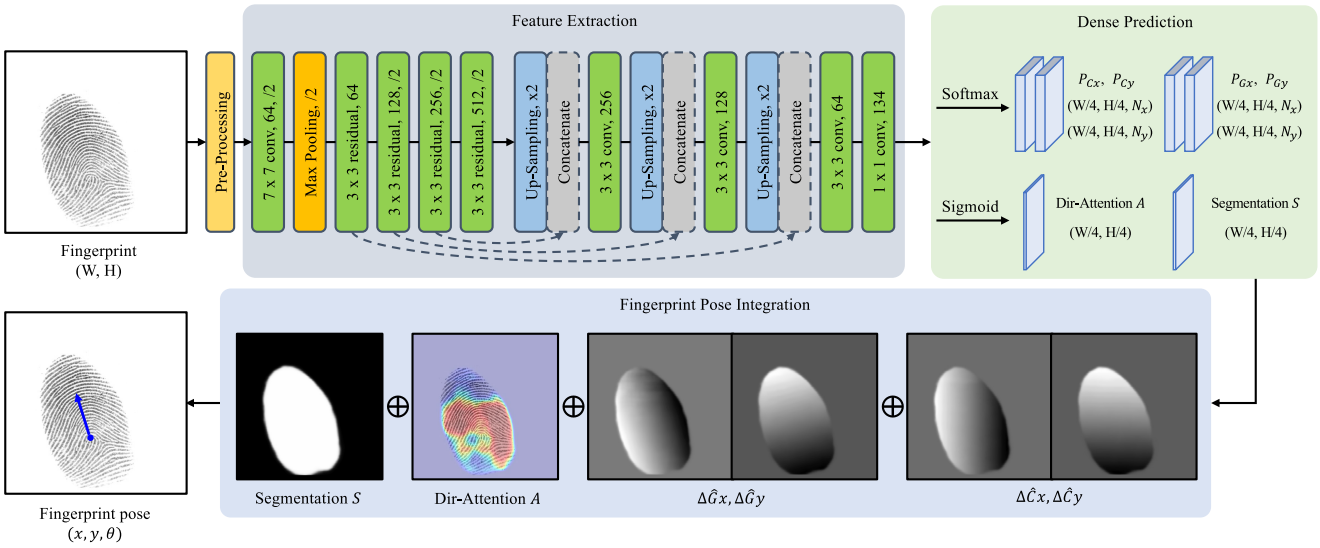


Fig. 3. Overview of the proposed method. We denote "conv" as 2D convolution layer, and "residual" as residual block defined in ResNet-18 [32]. The output of dense prediction module is $1/4$ size of the original input, and N is the number of intervals. Then the estimated dense maps (P_{Cx}, p_{Cy}, P_{Gx}, P_{Gy}, A, and S) are integrated to vote for fingerprint pose. For simplicity, we show the weighted average offset maps ($\Delta\hat{C}x$, $\Delta\hat{C}y$, $\Delta\hat{G}x$, and $\Delta\hat{G}y$) calculated by (4) and (5), where intensity indicates offset value to fingerprint center (white denotes positive offset and dark denotes negative).

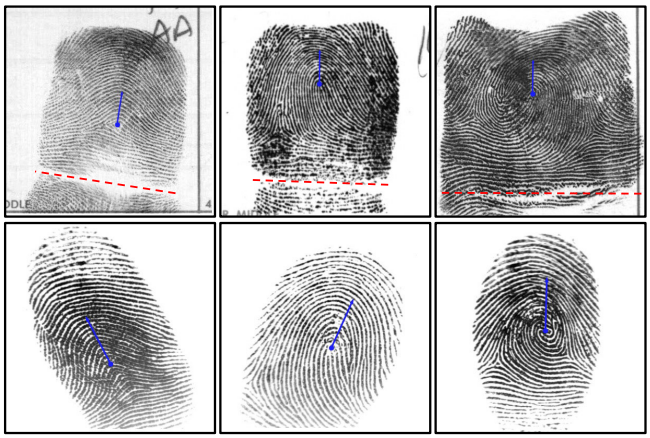


Fig. 4. Examples of fingerprint pose definition. Blue solid circle and arrow denote fingerprint center and direction respectively. The first row shows rolled fingerprints with none, one, and two core singular points. Similarly, the second row shows plain fingerprints with different numbers of core singular points. Red dotted lines indicate the direction of knuckles.

more consistent and robust across different fingerprints with noise and partial visible area. Briefly, fingerprint direction is defined to be perpendicular to ridge orientations around knuckles. Fingerprint center is determined based on the northernmost core points (fingerprints with singularities) or maximum curvature points (fingerprints without singularities). More definition details can be found in [20], and the following fingerprint indexing and verification performance has demonstrated the effectiveness and feasiblity of using this fingerprint pose definition [14], [15], [33]. Examples of fingerprint pose on different fingerprint patterns are shown in Fig. 4.

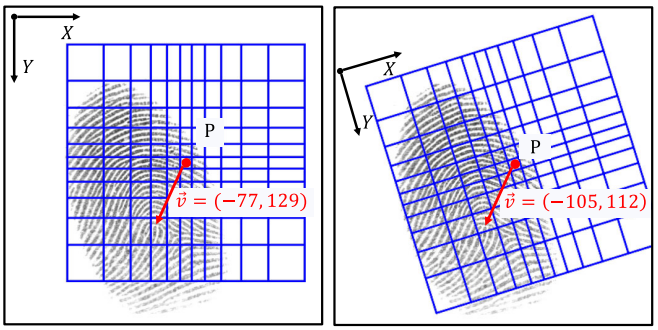
B. Network Architecture

Due to potential missing or noisy area in fingerprints, dense voting strategy based method was utilized in this paper. For each local patch within fingerprint area, offset vectors to fingerprint center are estimated. And fingerprint pose is obtained based on these dense offset maps by voting strategy, thus improving estimation performance on incomplete or noisy fingerprints. Although, voting strategy has been applied for fingerprint center estimation in previous works [14], [26], fingerprint direction was still regressed or classified as a single value directly since rolled fingerprints, which contain complete area, were focused in such methods. However, for plain fingerprints, noisy ridge pattern or partial visible area makes it challenging to estimate accurate and robust fingerprint direction directly. Therefore, we proposed a dense voting network to vote for fingerprint center and direction. Network architecture is illustrated in Fig. 3.

1) Preprocessing Module: To deal with fingerprints with diverse ridge quality, local total variation decomposition [34], [35] is adopted to enhance ridge structure. Ridge orientations in sine and cosine format of double angles are then calculated on the enhanced ridges by gradient based method to integrate ridge orientation information. We concatenate these maps to obtain a three-channel input for following modules.

2) Feature Extraction Module: Since fingerprint pose is classified as a global feature, i.e. level-0 feature [13], ResNet-18 [32] was utilized as our backbone to extract discriminative representations conveying essential multi-scale information. Then a reconstruction module was employed for feature maps up-sampling. Considering compromises between precision and efficiency, the final predicted maps are $1/4$ size of the original input. Skip connections were also adopted to integrate coarse and fine information [36].

3) Dense Prediction Module: To alleviate poor ridge quality and partial visible area in diverse fingerprints, we proposed to utilize voting method for fingerprint pose estimation. Different from previous methods, we also applied voting strategy in fingerprint direction estimation by predicting absolute location



(a) (b)

Fig. 5. Examples of (a) direction-irrelevant and (b) direction-relevant offset maps. The attached corresponding coordinate systems are drawn in black. The blue grids are examples of distance aware offset interval boundaries on point P. Offset vectors ground truth are shown in red under different coordinates.

of each patch within fingerprint area, named as direction-relevant offset vector in this paper. Considering the general size of fingerprint, the maximum offset is assumed to be ± 256 for both axes. Besides, rather than value regression, offset was estimated by interval classification to reduce the difficulty of continuous value regression.

We denote the number of intervals along X and Y axis as N_x and N_y. Intuitively, the uncertainty increases with the distance to fingerprint center, making it less reliable to predict offset vectors on regions away from fingerprint center. Therefore, instead of dividing intervals equally, we applied a distance aware division (as shown in Fig. 5), i.e. interval width is small when the distance to fingerprint center is small and vice versa. In this paper, a proportional mapping function

$$x' = \frac{x}{2 - |x/256|}, \tag{1}$$

where x is the interval boundary of equal division, was utilized to determine the new interval boundaries x'. An example of interval division is shown in Fig. 5. Besides, considering that not all pixels are valuable for fingerprint direction estimation, e.g. area around fingerprint center, a direction attention map was also predicted to focus on the important area. Therefore, totally six output maps were obtained, including two direction-irrelevant offset maps, two direction-relevant offset maps, a segmentation probability map, and a direction attention map. Details of these dense maps are described as following:

- Direction-irrelevant offset maps denote offsets probabilities from each patch center to fingerprint center along image X and Y axis respectively. The offset vectors share the same coordinate system of the original image (as shown in Fig. 5a);
- Similarly, direction-relevant offset maps denote offset probabilities from each patch center to fingerprint center along fingerprint X and Y axis respectively. In fact, the offset vectors indicate the absolute locations of patches within the aligned fingerprint (as shown in Fig. 5b);
- A segmentation map indicates foreground area;
- A direction attention map indicates area that provide effective information for fingerprint direction estimation.

Fig. 6. Two examples of fingerprint center heatmap voted by different samples. The left column shows heatmap voted by samples away from fingerprint center, thus obtaining less reliable estimation. The middle column shows samples close to fingerprint center, and the right column is the combination result. Samples for voting are indicated by green cross.

C. Fingerprint Pose Integration

Given the estimated offset probability maps, fingerprint center and direction can be calculated by voting method. The predicted direction-irrelevant offset along image X axis at pixel (i, j) can be represented as a unit-length N_x dimensional vector $P_{Cx} = \{p^k_{Cx} = P(\Delta Cx^k), k = 1, 2, \ldots, N\}$, where the kth element p^k_{Cx} indicates the probability of offset being the center of the kth offset interval ΔCx^k. Following the similar definition, P_{Cy} and ΔCy are utilized for direction-irrelevant offset along image Y axis, P_{Gx} and ΔGx for direction-relevant offset along fingerprint X axis, and P_{Gy} and ΔGy for direction-relevant offset along fingerprint Y axis. The estimated segmentation probability map is denoted as S and the direction attention map as A. By integrating these estimated dense maps, fingerprint center and direction can be calculated as follows.

1) Fingerprint Center: For fingerprint center estimation, only direction-irrelevant offset maps were utilized. It is a rational assumption that regions have a high degree of confidence in voting for adjacent area, therefore, we applied scaling for estimated offset probabilities

$$g(p^{ijk}) = p^{ijk}/\delta^k, \tag{2}$$

where δ^k indicates the width of the kth interval. Then the estimated fingerprint center $(\hat{x}, \hat{y})$ can be calculated as

$$\begin{aligned} \hat{x} &= \frac{\sum_{ijk}(i + \Delta Cx^{ijk}) \cdot g(p^{ijk}_{Cx}) \cdot S^{ij}}{\sum_{ijk} \cdot g(p^{ijk}_{Cx}) \cdot S^{ij}}, \\ \hat{y} &= \frac{\sum_{ijk}(j + \Delta Cy^{ijk}) \cdot g(p^{ijk}_{Cy}) \cdot S^{ij}}{\sum_{ijk} \cdot g(p^{ijk}_{Cy}) \cdot S^{ij}}, \end{aligned} \tag{3}$$

where $(i, j) \in \Omega$ indicates the coordinate across the output map. Examples of fingerprint center heatmap voted by different samples are shown in Fig. 6.

2) Fingerprint Direction: For fingerprint direction estimation, different from conventional approaches where direction is regressed directly, we make the prediction using dense voting

strategy. Apart from direction-irrelevant offset maps, direction-relevant offset maps, which indicate the absolute locations of local patches in the aligned fingerprint, i.e. centered and upright fingerprint, are also estimated.

For each pixel (i, j) within the output map, we first calculated the potential offset vectors $(\Delta\hat{C}x, \Delta\hat{C}y)$ and $(\Delta\hat{G}x, \Delta\hat{G}y)$ by weighted averaging,

$$(\Delta\hat{C}x, \Delta\hat{C}y) = \left(\sum\nolimits_k p^k_{Cx}\Delta Cx^k, \ \sum\nolimits_k p^k_{Cy}\Delta Cy^k, \right) \quad (4)$$

$$(\Delta\hat{G}x.\Delta\hat{G}y) = \left(\sum\nolimits_k p^k_{Gx}\Delta Gx^k, \ \sum\nolimits_k p^k_{Gy}\Delta Gy^k.\right) \quad (5)$$

Then, the estimated fingerprint direction $\hat{\theta}$ can be calculated using $(\Delta\hat{C}x, \Delta\hat{C}y)$ and $(\Delta\hat{G}x, \Delta\hat{G}y)$ based on the amplitude of inner and cross product:

$$\begin{aligned}\sin\hat{\theta} &= \frac{1}{Z}\sum\nolimits_{ij} (\Delta\hat{G}x\Delta\hat{C}y - \Delta\hat{G}y\Delta\hat{C}x)\cdot S^{ij}A^{ij},\\ \cos\hat{\theta} &= \frac{1}{Z}\sum\nolimits_{ij} (\Delta\hat{G}x\Delta\hat{C}x + \Delta\hat{G}y\Delta\hat{C}y)\cdot S^{ij}A^{ij}, \end{aligned} \quad (6)$$

where trigonometric functions are utilized for angular summation over the output map and Z is the normalization factor.

D. Objective Functions

1) Offset Loss: Generally, cross entropy loss is used in classification task. However, considering the confusion between adjacent intervals when the real offset lies near the interval boundary, in this paper, Gaussian smoothed label strategy [37] was utilized to generate the ground truth of dense offset maps (T_{Cx}, T_{Cy}, T_{Gx}, and T_{Gy}) for supervision, which has the maximum probability ($p = 1$) as the true interval and decreases the probability value as the distance increases. KL (Kullback-Leibler) divergence loss $\mathcal{L}_{\text{KL}}$ was then applied for optimization. Given the estimated direction-irrelevant and direction-relevant offset maps and their ground truth, offset loss can be calculated as

$$\begin{aligned}\mathcal{L}_{\text{offset}} = \frac{1}{|\Omega|}\big(&\mathcal{L}_{\text{KL}}(P_{Cx}, T_{Cx}) + \mathcal{L}_{\text{KL}}(P_{Cy}, T_{Cy})\\ &+ \mathcal{L}_{\text{KL}}(P_{Gx}, T_{Gx}) + \mathcal{L}_{\text{KL}}(P_{Gy}, T_{Gy})\big).\end{aligned} \quad (7)$$

2) Expectation Loss: Apart from offset loss on pixel level, we also constrained the final estimated fingerprint pose using a mean squared error loss

$$\mathcal{L}_{\text{exp}} = \|(\hat{x}, \hat{y}) - (x, y)\|_2^2 + \|\hat{\theta} - \theta\|^2, \quad (8)$$

where (x, y) and θ are the ground truth of fingerprint pose.

3) Segmentation Loss: For segmentation probability map, a weighted binary cross entropy loss was used for optimization

$$\mathcal{L}_{\text{seg}} = -\frac{1}{|\Omega|}\sum\nolimits_{ij}\left(\omega_p \log(S^{ij}) + \omega_n \log(1 - S^{ij})\right), \quad (9)$$

where ω_p and ω_n are weights for positive and negative samples respectively.

4) Overall Loss: Consequently, the proposed network was optimized by minimizing the overall objective function

$$\mathcal{L} = \mathcal{L}_{\text{offset}} + \lambda_{\text{exp}}\mathcal{L}_{\text{exp}} + \lambda_{\text{seg}}\mathcal{L}_{\text{seg}}, \quad (10)$$

where λ_{exp} and λ_{seg} denote the trade-off parameters for expectation loss and segmentation loss respectively, and set to 0.001 and 1 for roughly the same loss decreasing rate.

IV. Implementation Details

We implemented our method with PyTorch on NVIDIA GeForce 3090. Data augmentation was performed during training, including random translation in range of 20% width and height of input size, rotation to follow a uniform distribution $[-45°, 45°]$, horizontal flip, and additive noise sampled from Gaussian distribution $\mathcal{N}(0, 0.1)$. Besides, to improve the generalization of the proposed network, additive natural images were also applied on training fingerprints as background noise.

The maximum offset is set to ± 256 and the number of intervals N_x and N_y as 33. Using the distance aware interval division in (1), the interval width ranges from 8 to 29. Network weights are optimized using an Adam optimizer with initial learning rate of 0.00035. Learning rate decays by 0.1 when the performance on validation dataset does not improved after 10 epochs, and training is stopped after three times decays. L2 regularization with weight of 0.01 is applied to avoid over-fitting. The network is trained from scratch with kaiming weights initialization. There is no limitation on input image size since voting strategy is utilized in our method.

V. Experimental Results

In this section, we conducted several experiments for evaluation. We will first introduce all datasets used in this paper, then the evaluation on minutiae alignment accuracy, fingerprint indexing, and fingerprint verification performance. And we also show the ablation study, qualitative evaluation, and algorithm efficiency of our method. Finally, limitations of the proposed approach were discussed.

A. Datasets

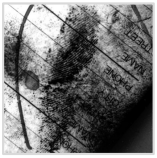
Various fingerprint databases including two rolled, four plain, two latent, one contactless, and our collected DPF (diverse poses fingerprints) datasets were utilized for evaluation. Table I presents the details of all databases and their usage in this paper. Fingerprint examples in each database are also shown in the table. We utilized three types of fingerprint, i.e. rolled, plain, and latent, for network training respectively, due to significant differences in ridge quality and fingerprint area, and no further fine-tuning was performed (e.g. the same network is used for all plain fingerprint datasets).

For rolled fingerprints, the first 3,200 images in NIST SD14 were selected for training. The fingerprint pose ground truth for training were manually labeled. And the "ground truth" of segmentation map was generated by a pre-trained deep network, which was trained based on the segmentation generated by the traditional gradient based method [38]. In fact, the generated "ground truth" is enough since the probability map was utilized to determine the voting area and its estimation accuracy is not focused in this paper.

For plain fingerprints, our collected dataset DPF, consisting of plain fingerprints with arbitrary visible area and their corresponding rolled fingerprints, was used to increase data diversity. A total of 40,112 plain impressions from 776 fingers were collected. We randomly selected 32,676 plain impressions from 633 fingers for training, and the rest 143 fingers for evaluation. Considering the difficulty of pose annotation

TABLE I
ALL FINGERPRINT DATASETS USED IN EXPERIMENTS

Database	NIST SD4	NIST SD14	FVC2000 DB2A	FVC2000 DB3A	FVC2002 DB1A
	Rolled	Rolled	Plain	Plain	Plain
Image					
Sensor	Inking	Inking	Capacitive	Optical	Optical
Description	2,000 pairs	27,000 pairs	100 fingers × 8	100 fingers × 8	100 fingers × 8
Image size	512 × 480	800 × 768	256 × 364	448 × 478	388 × 374
Usage	Testing	Training & testing[1]	Testing	Testing	Testing

Database	FVC2004 DB1A	NIST SD27		PolyU	
	Plain	Rolled	Latent	Contact-based	Contactless[2]
Image					
Sensor	Optical	Inking	-	Optical	Camera
Description	100 fingers × 8	258 pairs latent fingerprints from real crime scenes		336 fingers × 6 contact-based and contactless fingerprints	
Image size	640 × 480	800 × 768	800 × 768	328 × 356	225 × 350
Usage	Testing	Testing[3]		Testing	

Database	DPF		Hisign Latent	
	Rolled	Plain	File (rolled / plain)	Latent
Image				
Sensor	Optical	Optical	Inking / Optical	-
Description	776 rolled 40,112 plain fingerprints with diverse poses		10458 pairs latent fingerprints from real crime scenes	
Image size	800 × 750	512 × 512	640 × 640	512 × 512
Usage	Training and testing[4]		Training and testing[5]	

[1] The first 3,200 rolled fingerprints for training. The last 2,700 pairs for testing.
[2] Original contactless fingerprints are scaled to match the same mean ridge period of contact-based fingerprints [7].
[3] 10,458 file fingerprints from Hisign latent dataset were added as background.
[4] Randomly selected 633 rolled fingerprints and their corresponding 32,676 plain fingerprints for training. The rest 143 rolled fingerprints and 8 randomly selected plain fingerprints per finger (143 × 8) for testing.
[5] The first 8,000 pairs of file and latent fingerprints for training. The last 2,458 pairs for testing.

on plain fingerprints due to partial area, we first manually annotated fingerprint poses on the 633 rolled fingerprints in training set, then plain fingerprint poses were transformed from the corresponding rolled fingerprints based on mated minutiae pairs calculated by the minutiae matcher described in [33]. And segmentation "ground truth" was generated similarly as rolled fingerprints.

For latent fingerprints, the first 8,000 images in Hisign latent dataset were selected for training. Instead of manually annotating on latent fingerprints, similarly, file fingerprint poses were obtained first. Considering the large number of training samples and manually labeled latent minutiae by latent fingerprint examiners, we chose to utilize Yin's method [13], which is the SOTA method for rolled and plain fingerprints with frontal pose, to predict file fingerprint poses. Minutiae of file fingerprints were then extracted using the commercial software VeriFinger SDK 12.0 [39]. Then the latent fingerprint poses can be obtained by calculating mated minutiae pairs between file and latent fingerprints. Selective examinations were performed on the transformed latent fingerprint poses and satisfactory results were observed thanks to the manually labeled minutiae on latent fingerprints. As for segmentation "ground truth", generation steps similar to [40] were utilized, i.e., the "ground truth" was generated by calculating

the convex hull of manually labeled minutiae of latent fingerprints.

FVC2000, FVC2002, FVC2004 are widely used public plain fingerprint databases and contain multiple subsets. We follow the same experimental settings in previous studies [8], [9]: FVC2000 DB2A & DB3A, FVC2002 DB1A, and FVC2004 DB1A were selected for evaluation. Note that plain fingerprints in FVC2000 [41] were collected with limited rotation range $[-15°, 15°]$, we applied additional random rotation in range of $[-30°, 30°]$ on these fingerprints to increase direction diversity for more convincing experiments. Besides, apart from comparison between the same fingerprint types, e.g., NIST SD4 (all rolled fingerprints) and FVC2004 DB1A (all plain fingerprints), we also evaluated between fingerprints with different types, i.e., DPF (plain to rolled), NIST SD27 (latent to rolled), Hisign latent (latent to rolled or plain), and PolyU (contactless to contact-based), which requires more reliable, accurate, and consistent pose estimation.

B. Evaluation on Minutiae Alignment Accuracy

Given the estimated poses, fingerprints from the same fingers can be aligned to a unified coordinate system. Then deviation in location and direction between mated minutiae pairs can be utilized to evaluate the consistency of fingerprint pose estimation algorithms on fingerprints collected under different conditions [9], [13]. First, we extracted minutiae using VeriFinger [39] (minutiae of latent fingerprints in NIST SD27 and Hisign latent datasets were manually labeled). Mated minutiae pairs were then calculated by the minutiae matcher described in [33]. Finally, location and direction deviations between mated minutiae pairs can be calculated after fingerprint alignment.

Quantitative comparison with previous works, including Su's [9], Ouyang's [11], and Yin's [13] method, was performed. Besides, Engelsma et al. [12] proposed a deep network, DeepPrint, which utilized STN to learn fingerprint alignment implicitly. Therefore, we also reimplemented DeepPrint and output the prediction of STN as fingerprint pose estimation. The network was trained using the first 24K pairs of fingerprints, which is much smaller than the private database used in [12] (a large dataset consisting of 445K fingerprints from 38K unique fingers), and data augmentation, including random histogram matching and fingerprint distortion, was employed in training to alleviate the lack of training data.

Note that Su's [9] and Ouyang's [11] method were designed for rolled fingerprints and thus not suitable for plain or latent [13]. And Engelsma's method [12] is also not suitable for latent fingerprints since it was trained for complete fingerprints (e.g., rolled and frontal plain fingerprints). For a fair comparison, Yin's model [13] were re-trained using the same training data in this paper. And for latent fingerprints, FingerNet [40] was first adopted for segmentation and enhancement, and the predicted segmentation was utilized for image pre-cropping in Yin's method [13], then fingerprint poses were estimated based on the enhanced fingerprints for all algorithms.

Fig. 7 and Fig. 8 present the deviation distribution between mated minutiae pairs on part of fingerprint datasets (more experimental results can be seen in supplementary materials). Location deviation is measured in pixels, and direction in degrees. Due to the large area, sufficient information can be extracted from rolled fingerprints, and the direction of rolled fingerprints is also limited since they were collected in controlled environment. Therefore, direction deviation on rolled fingerprints is limited and there is no significant difference among different algorithms. While for other fingerprints, presence of central area and ridge quality is not guaranteed, thus estimation on these fingerprints is more challenging.

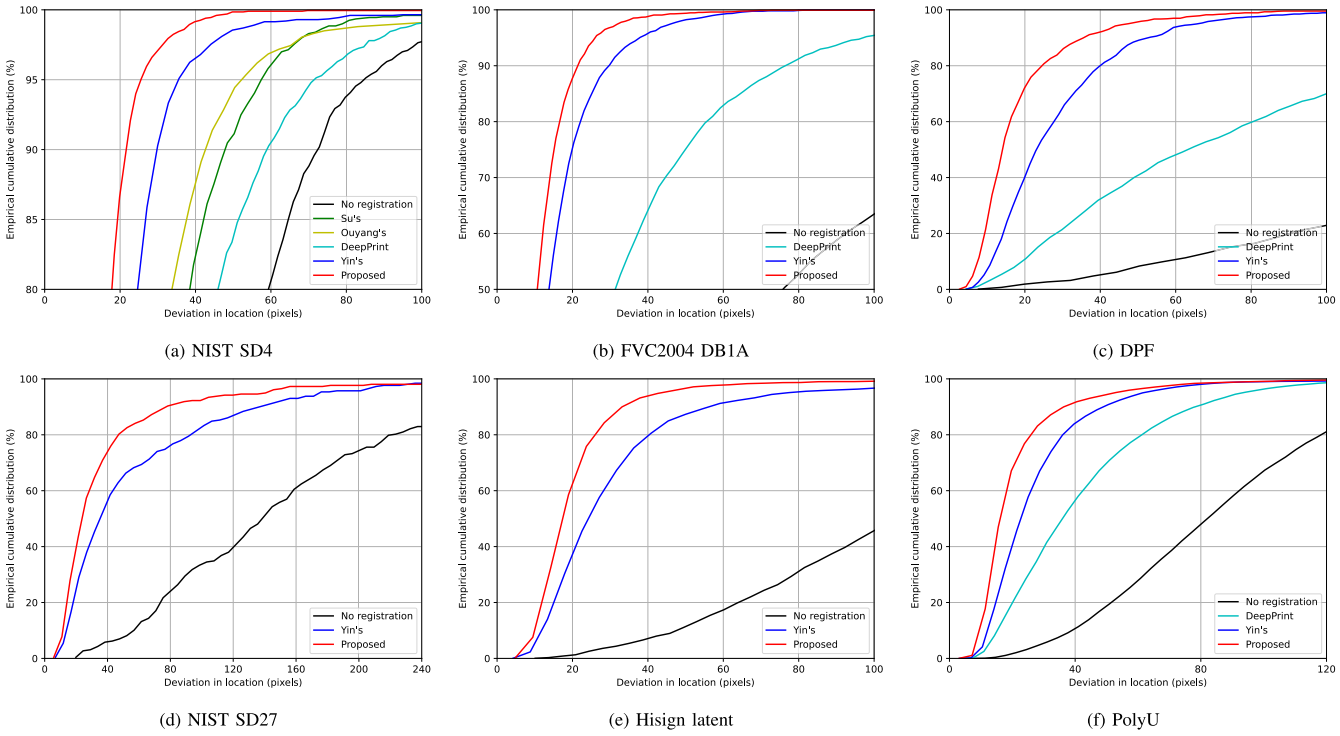
As shown in Fig. 7 and 8, our method outperforms the SOTA methods on various fingerprint datasets. Besides, despite the large diversity of visible fingerprint area and ridge quality, minutiae alignment accuracy of fingerprints with different types can be significantly improved based on the more accurate and consistent fingerprint poses estimated by our method. It is also observed that results of Engelsma's method [12] are unsatisfactory and performance on direction deviation gets even worse on all datasets (estimations on latent fingerprints were not performed as mentioned above). The reason may be that fingerprint alignment in [12] is not supervised by explicit fingerprint pose annotations, but optimized by the following fingerprint matching indirectly. And the matching performance using DeepPrint [12] based on different fingerprint alignment algorithms was also evaluated in the following experiments.

C. Evaluation on Fingerprint Indexing

Given the correctly estimated fingerprint poses, deviation between mated minutiae pairs can be reduced as demonstrated. Therefore, we carried out a minutiae-based fingerprint indexing experiment which incorporates fingerprint pose as constraints to improve indexing accuracy and efficiency [9]. In [9], binary minutiae cylinder-code (MCC) was utilized to calculate matching score of two fingerprints by measuring the similarity between minutiae pairs. After fingerprint alignment, only minutiae pairs, whose deviations in location and direction are lower than the defined threshold, are considered in similarity calculation, thus reducing the searching space and wrong matched minutiae. In this paper, different thresholds were determined based on the previous minutiae alignment accuracy, and detailed settings for all datasets are shown in supplementary materials.

For plain fingerprint datasets, the first impression from each finger was selected as gallery, and the rest fingerprints as query. For DPF dataset, rolled fingerprint was set as gallery and the corresponding 8 randomly selected plain fingerprints were utilized as query. For NIST SD27 dataset, to increase the size of background dataset, additional 10,458 file fingerprints from Hisign latent dataset were added as gallery. For PolyU dataset, the first contact-based fingerprint from each finger was selected as gallery, and all contactless fingerprints as query.

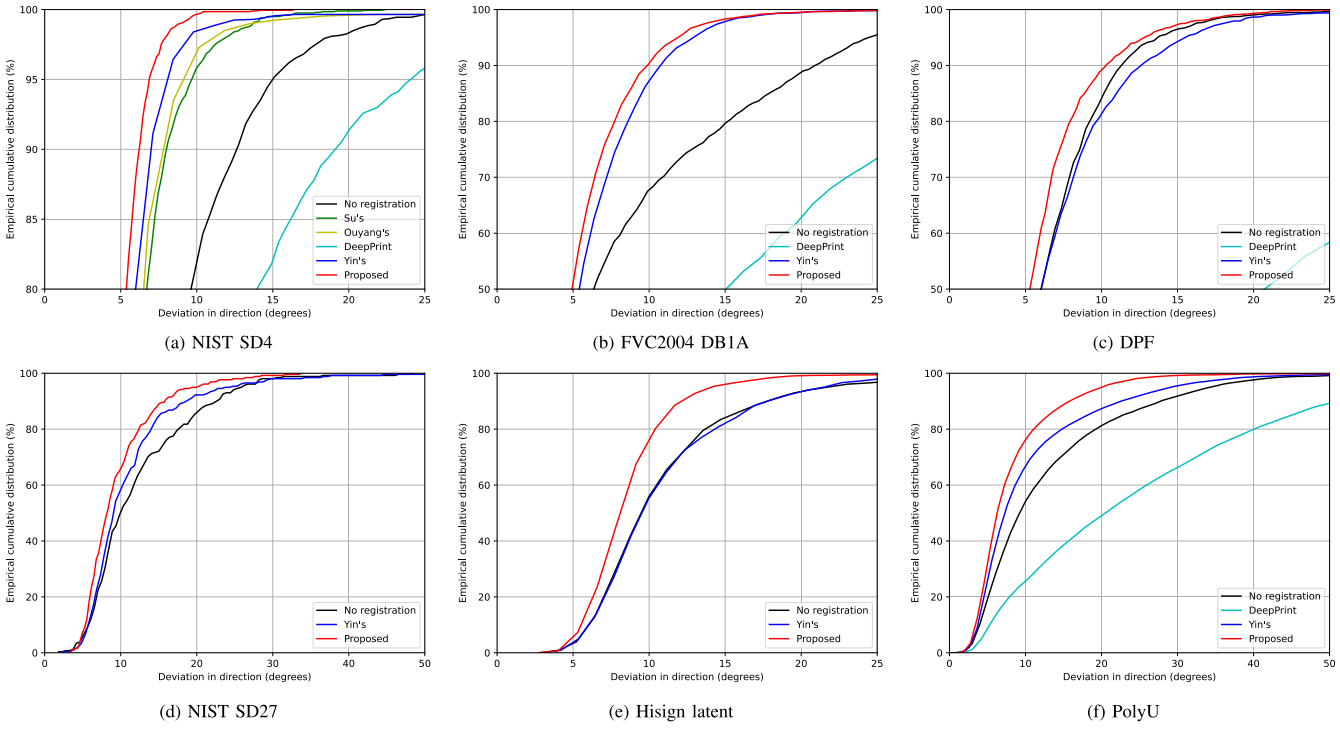
Fig. 9 shows binary MCC [9] based indexing performance on part of datasets (see more results in supplementary materials). It can be observed that with loose constraints,



(a) NIST SD4 (b) FVC2004 DB1A (c) DPF

(d) NIST SD27 (e) Hisign latent (f) PolyU

Fig. 7. The empirical cumulative distribution curve of location deviations between mated minutiae pairs on various fingerprint databases.

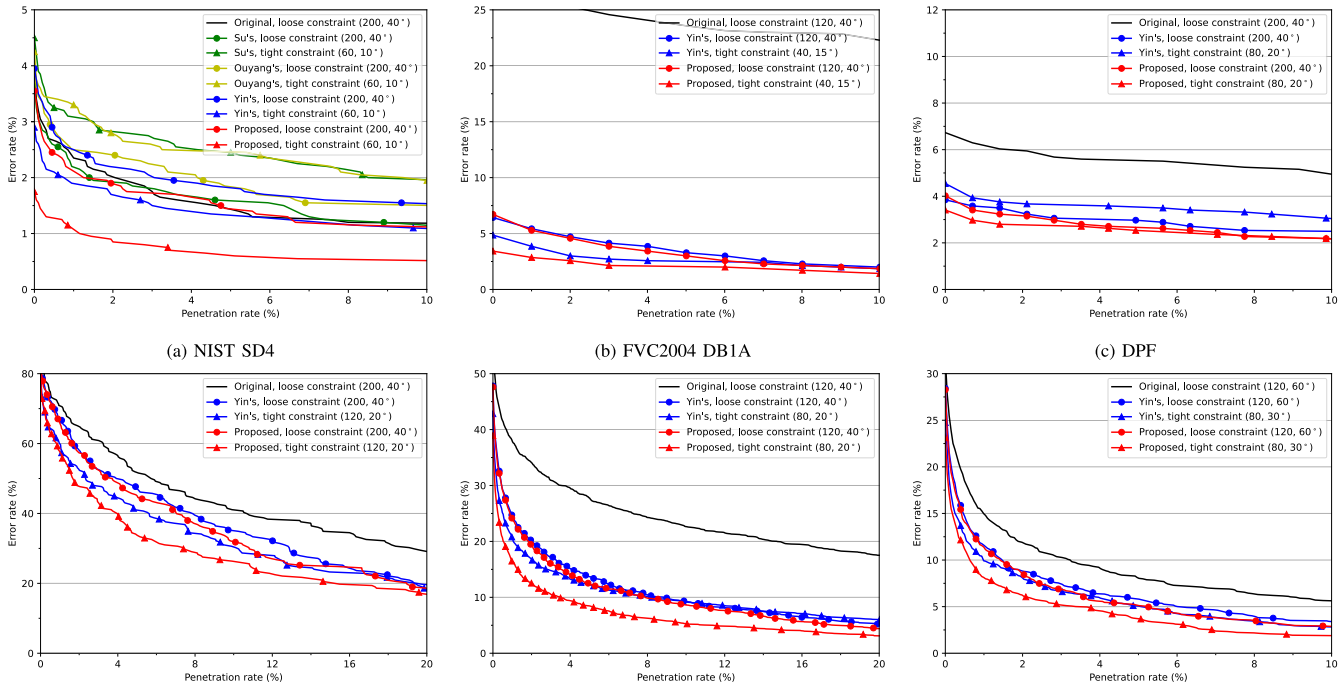


(a) NIST SD4 (b) FVC2004 DB1A (c) DPF

(d) NIST SD27 (e) Hisign latent (f) PolyU

Fig. 8. The empirical cumulative distribution curve of direction deviations between mated minutiae pairs on various fingerprint databases.

similar indexing performance is obtained for different fingerprint pose estimation algorithms. While with tight constraints, our method performs better than the SOTA methods on various fingerprints, especially for NIST SD4 and Hisign latent datasets. Besides, performance on FVC2000 DB3A and two latent dataset is significantly lower compared to other datasets due to poor fingerprint quality (the first row in Fig. 10). In FVC2004 DB1A and DPF datasets, incomplete fingerprint area and large range of fingerprint poses also make fingerprint indexing more challenging (the second row in Fig. 10), but better performance can still be achieved using our method.



(a) NIST SD4 (b) FVC2004 DB1A (c) DPF

(d) NIST SD27 (e) Hisign latent (f) PolyU

Fig. 9. The binary MCC [9] based fingerprint indexing performance with loose and tight constraints on various fingerprint databases.

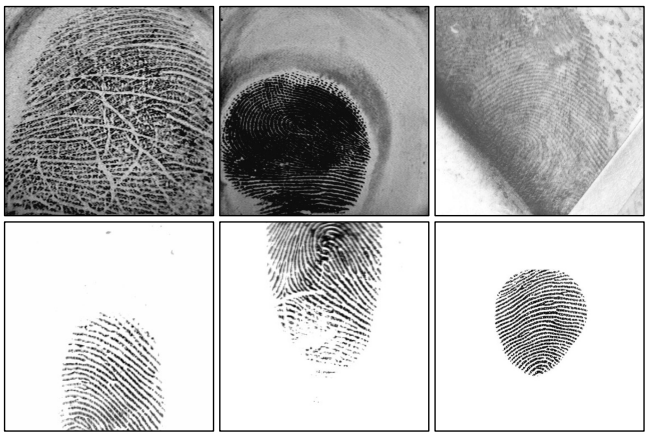


Fig. 10. Examples of fingerprints with poor ridge quality (the first row) and fingerprints with incomplete area and diverse finger poses (the second row).

D. Evaluation on Fingerprint Verification

Apart from fingerprint indexing, fingerprint pose can also be applied in fingerprint verification task. By introducing accurate and consistent fingerprint poses, performance of fingerprint verification (1-to-1 matching) can be improved since only features within appropriate spatial deviations are considered after fingerprint alignment, and most non-mated features can be rejected due to incompatible location and direction.

Instead of testing with commercial matchers, e.g., VeriFinger [39], in which custom modifications cannot be performed and matching performance hardly changes after minutiae or fingerprint alignment (probably because it contains its own internal relative fingerprint pose alignment), MCC [42], a well-designed local minutiae-only matcher with superior performance compared to other open source algorithms, was utilized to explore the effectiveness of incorporating fingerprint pose. Besides, the SOTA deep learning based fingerprint matcher, DeepPrint [12], was also utilized for evaluation further.

1) MCC [42] Based Matching: Considering that modifications cannot be performed directly in the released MCC SDK [42], we reimplemented MCC creation and four types of matching algorithms as described in [42], including local similarity sort (LSS), local similarity assignment (LSA), local similarity with relaxation (LSS-R), and local similarity assignment with relaxation (LSA-R). Only local similarities are considered in the first two types of matching algorithms (LSS and LSA). The same minutia is not considered more than once in LSA, which is not guaranteed in LSS. Then, by incorporating global spatial relationships, higher performance can be achieved in the other relaxation matching algorithms (LSS-R and LSA-R).

Similarly with previous experiments, fingerprint pose was introduced as constraints. To be specific, minutiae were first normalized to a unified standard coordinate system given the estimated fingerprint poses. It is a rational assumption that mated minutiae pairs have a small deviation after alignment, which can be integrated into the local similarity calculation. Therefore, before calculating local similarity scores between cylinder features from different fingerprints, we calculated the deviation score for each minutiae pair (m_i, m_j):

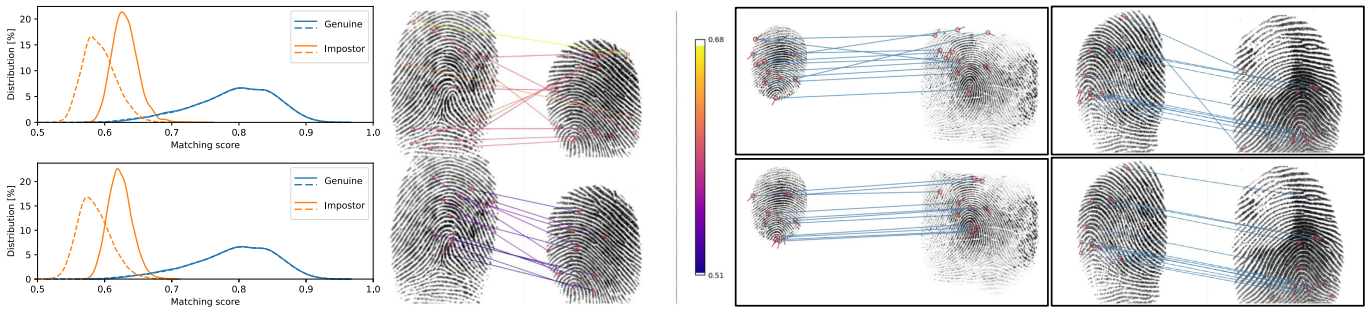
$$S_D = Z(d_l(m_i, m_j), \mu_l, \tau_l) * Z(d_\theta(m_i, m_j), \mu_\theta, \tau_\theta), \quad (11)$$

where d_l and d_θ indicate the deviation between location and direction respectively, and

$$Z(d, \mu, \tau) = \frac{1}{1 + e^{-\tau(d-\mu)}} \quad (12)$$

TABLE II
MCC [42] VERIFICATION PERFORMANCE (%) USING DIFFERENT FINGERPRINT POSES. FNMR@FMR = 0.1% IS REPORTED

Method	Fingerprint Pose	FVC2000 DB2A	FVC2000 DB3A	FVC2002 DB1A	FVC2004 DB1A	DPF	PolyU
LSS-R	-	0.61	10.36	0.25	10.36	4.20	52.36
LSA-R	-	0.54	10.11	0.25	8.86	3.85	48.07
LSS	-	2.79	34.61	1.68	14.89	8.92	96.16
	Yin's pose [13]	1.25↓55%	13.46↓61%	2.57↑ 53%	6.04↓59%	8.04↓10%	33.76↓65%
	proposed pose	0.79↓72%	12.11↓65%	0.54↓68%	5.93↓60%	6.64↓26%	29.79↓69%
LSA	-	0.93	18.14	0.43	6.68	6.12	71.50
	Yin's pose [13]	1.00↑ 7%	12.86↓29%	2.71↑ 530%	4.57↓32%	7.34↑ 20%	29.46↓59%
	proposed pose	0.71↓24%	11.00↓39%	0.32↓26%	4.71↓29%	5.77↓6%	25.40↓64%



(a) (b) (c)

Fig. 11. Incorporating fingerprint pose estimated by our method as constraints in MCC based matching algorithms (LSS and LSA). (a) Distribution of matching scores on FVC2004 DB1A before (solid line) and after (dashed line) incorporating fingerprint pose in LSS (top) and LSA (bottom). (b) An example of estimated minutiae pairs on impostor fingerprint pair before (top) and after (bottom) incorporating fingerprint pose in LSA. The line color indicates local similarity score calculated between MCC descriptors proposed in [42]. (c) Examples of estimated minutiae pairs on genuine matches before (top) and after (bottom) incorporating fingerprint pose in LSS (the first column) and LSA (the second column).

is a normalizing function. We set $\mu_l = 60$, $\tau_l = -0.1$, $\mu_\theta = \pi/9$, and $\tau_\theta = -20$ in this experiment. Intuitively, outliers were defined as minutiae pairs whose deviation scores are lower than a threshold, and removing these outliers can decrease potential incorrect matches. In this experiment, the removing deviation score threshold was set to $d_l = 60$ and $d_\theta = \pi/9$. Local similarity scores were then calculated for the remaining minutiae pair candidates. Table II shows the MCC [42] matching performance on different datasets using different fingerprint pose estimations for fingerprint alignment.

As shown in Table II, false non-match rate (FNMR) is decreased significantly by using the proposed fingerprint pose as constraints in LSS and LSA, where only local similarities are considered while ignoring global spatial constraints. And by incorporating fingerprint pose as constraints in the two local matching algorithms, competitive or higher performance (e.g., FVC2004 DB1A and PolyU) can be achieved compared to global matching approaches (LSS-R and LSA-R).

Fig. 11a shows the distributions of matching scores using LSS and LSA. Significant decreases can be observed for impostor matching scores owing to the incorporation of fingerprint poses. Fig. 11b and Fig. 11c present examples of estimated minutiae pairs using LSS or LSA on impostor and genuine pairs, respectively. MCC minutiae descriptor is defined on the local structure around minutia, and the corresponding matcher is expected to find minutiae pairs with the most similar local structure [42]. Therefore, for impostor fingerprint pairs, due to the incorporation of fingerprint pose constraint, minutiae pairs with the most similar local structure but incompatible locations may be rejected, then minutiae pairs with less similarity will be matched. Consequently, impostor score values will decrease significantly as shown in Fig. 11a and 11b. While for genuine fingerprint pairs, much fewer matches are corrected by pose constraint since the most similar minutiae pairs are already correct. Even for the few corrected ones, the similarity after correction is very high since they are the true minutiae pairs. Therefore, the reduction of genuine score values after introducing fingerprint pose as constraint is not so drastic as the decrease of impostor score values, and more rational and better minutiae matches can be obtained (as shown in Fig. 11c).

2) DeepPrint [12] Based Matching: We replaced the fingerprint alignment generated by STN in DeepPrint [12] with estimated fingerprint poses and then extracted fixed-length fingerprint features on the aligned fingerprints. Considering that similar image modality of fingerprints should be guaranteed for DeepPrint based fingerprint matching, comparing between the raw contactless and contact-based fingerprints in PolyU dataset is inappropriate, therefore, FingerNet [40] was first applied for fingerprint enhancement to unify the intensity distribution of input image, then the fixed-length fingerprint features can be extracted on these enhanced contactless and contact-based fingerprints. Table III shows the DeepPrint matching performance on different fingerprint datasets using different fingerprint pose estimations for fingerprint alignment.

As shown in Table III, it is observed that the performance of DeepPrint [12] can be further improved due to more accurate and consistent fingerprint alignment. We found accuracy on

TABLE III
DEEPPRINT [12] VERIFICATION ACCURACY (%) USING DIFFERENT FINGERPRINT POSES. FNMR@FMR = 0.1% IS REPORTED

Method	FVC2000 DB2A	FVC2000 DB3A	FVC2002 DB1A	FVC2004 DB1A	DPF	PolyU
DeepPrint	2.21	10.57	6.25	5.18	35.49	40.77
DeepPrint + Yin's pose [13]	2.29↑4%	9.32↓12%	4.68↓25%	4.39↓15%	33.13↓7%	39.91↓2%
DeepPrint + proposed pose	1.93↓13%	6.68↓37%	4.32↓31%	4.43↓14%	30.86↓13%	39.83↓2%

TABLE IV
ABLATION STUDY RESULTS. MEAN DEVIATIONS BETWEEN MATED MINUTIAE LOCATION (PIXELS) AND DIRECTION (DEGREES) ARE REPORTED

Vote for center	Vote for direction	Direction attention	FVC2002 DB1A	
			Location	Direction
-	-	-	19.35±16.87	5.42±3.86
OR	-	-	14.36±9.41	5.37±2.85
OR	OR	-	11.01±9.83	5.00±2.58
OR	OR	✓	11.20±9.27	4.74±2.41
IC	IC	✓	9.02±6.95	4.34±1.96

DPF dataset is significantly low, since different fingerprint types were involved, i.e., matching between rolled and plain fingerprints with diverse visible area. Besides, due to the perspective distortion and different intensity distribution in contactless fingerprints, verification between contactless and contact-based fingerprints is also a challenging task, but more consistent fingerprint alignment estimated by the proposed method still helps to improve matching performance.

E. Ablation Study

Finally, an ablation study is performed to demonstrate the importance of introducing dense voting strategy and direction attention module in fingerprint pose estimation. In this study, we evaluated the performance based on mean deviations in location and direction between mated minutiae pairs of fingerprints in FVC2002 DB1A. For brevity, we denote "OR" as estimating offset by regression, and "IC" as interval classification. Note that when the fingerprint center or direction is not predicted by voting method, the network (excluding reconstruction module) will predict directly, i.e., two neurons for fingerprint center (x, y) and one neuron for direction θ.

As shown in Table IV, we found that utilizing voting strategy improves accuracy of fingerprint pose estimation, and superior performance can be observed in converting offset regression to interval classification. Estimating fingerprint direction using dense voting strategy also provides auxiliary information for more consistent fingerprint pose estimation. Besides, due to the incorporation of direction attention module, better performance can be observed in fingerprint direction estimation, which demonstrates the effectiveness of the proposed direction attention module.

F. Qualitative Evaluation and Algorithm Efficiency

We found that there are still nonnegligible minutiae deviations in location and direction after fingerprint alignment, especially for comparisons between fingerprints with difference types, e.g., plain to rolled and latent to rolled. However, improvements of using fingerprint pose can still be observed in the following indexing and verification tasks since minutiae deviations are quite large when there is no alignment (as shown in Fig. 7 and 8). And from the view of information theory, the uncertainty of matching is reduced greatly thanks to the fingerprint pose alignment, which is beneficial to the following fingerprint matching. Then based on the performance of fingerprint pose estimation, appropriate threshold can be determined to penalize impostor pairs as much as possible, while avoiding negative impact for genuine pairs. Therefore, given the more accurate and consistent fingerprint pose estimations predicted by our method, more performance improvement can be achieved.

Fig. 12 shows several typical estimations on different types of fingerprints. More robust estimation results are observed on fingerprints with different visible area and noisy ridge structures using our method, benefiting from dense voting strategy, which utilizes as much information within fingerprint area as possible. While for Yin's method [13], the estimation accuracy decreases significantly on incomplete fingerprints (especially for fingerprints where some singularities cannot be observed) since the fingerprint poses were regressed directly which is sensitive to incomplete fingerprint area, and singularities, the auxiliary information in Yin's method [13], are also not helpful on these partial fingerprints. Considering the shallow network of STN and indirect supervision, it is reasonable to obtain the inaccurate fingerprint pose estimations using DeepPrint [12]. Besides, as described in [12], only frontal plain fingerprints (e.g., the most fingerprints in FVC2004 DB1A dataset) are considered in the training strategy, while fingerprints in DPF are quite different from the frontal plain fingerprints, which contains many partial fingerprints (especially for fingerprints produced by fingertips). Fig. 13 also shows estimations on different impressions from the same finger, and more consistent estimations can be observed using the proposed algorithm, which further demonstrated the superiority of our method.

Fig. 14 shows several examples of direction attention map in our method. Obviously, in rolled fingerprints with large and complete fingerprint area, knuckles region plays an important role in fingerprint direction estimation. While for plain fingerprints, due to incomplete area, knuckles cannot be observed reliably, therefore, the proposed direction attention map help to focus on other area that is helpful for direction estimation, such as area above top core or below deltas.

On a computer with a NVIDIA GeForce 3090 GPU and a 3.0GHz CPU, Yin's method [13] takes about 53ms for pose

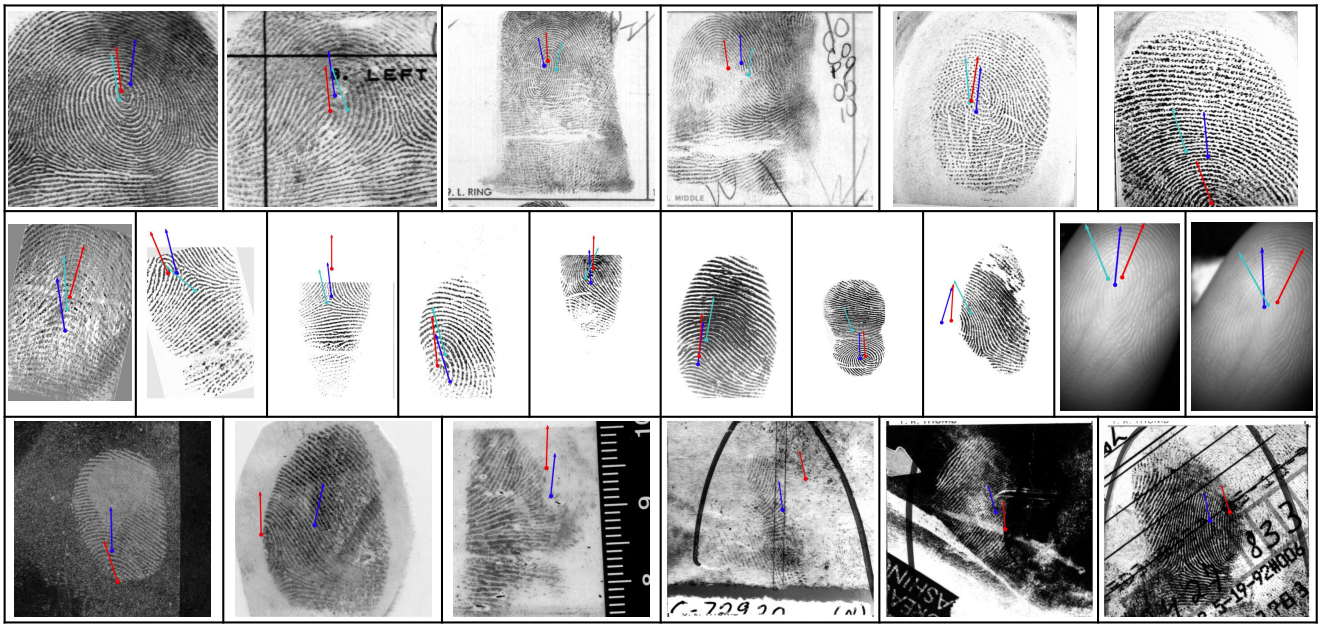


Fig. 12. Examples of fingerprint pose estimation results on different datasets (from left to right and from top to bottom are NIST SD4, NIST SD14, FVC2000 DB3A, FVC2000 DB2A, FVC2002 DB1A, FVC2004 DB1A, PolyU, Hisign latent, and NIST SD27). Cyan arrows indicate results of DeepPrint [12], blue arrows for Yin's method [13], and red arrows for our method.

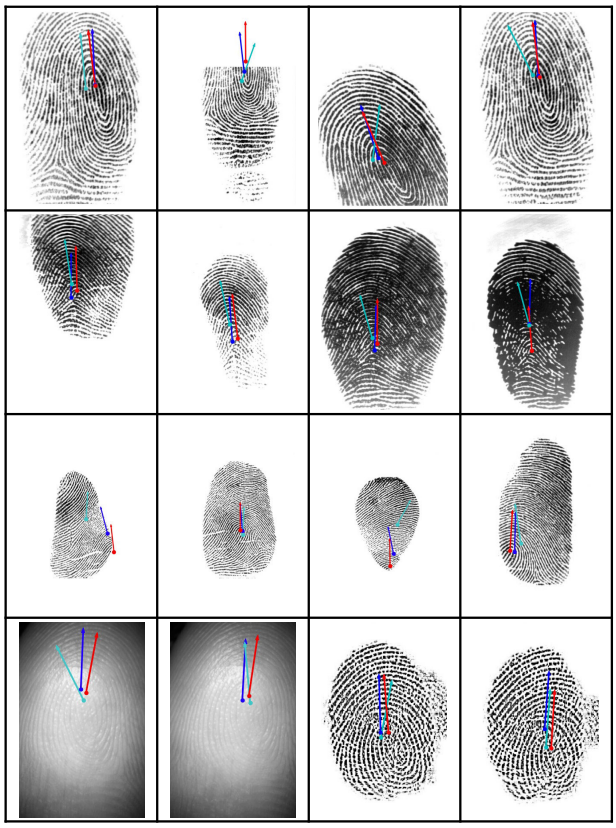


Fig. 13. Fingerprint pose estimation results on different impressions from the same finger (each row). From top row to bottom, examples are selected from FVC2002 DB1A, FVC2004 DB1A, DPF, and PolyU. Cyan arrows indicate results of DeepPrint [12], blue arrows for Yin's method [13], and red arrows for our method.

estimation on an image with size of 512 × 512, and the proposed algorithm takes about 75ms. The comparison result is reasonable since the proposed method contains up-sampling and voting modules which consume more time than predicting fingerprint pose directly.

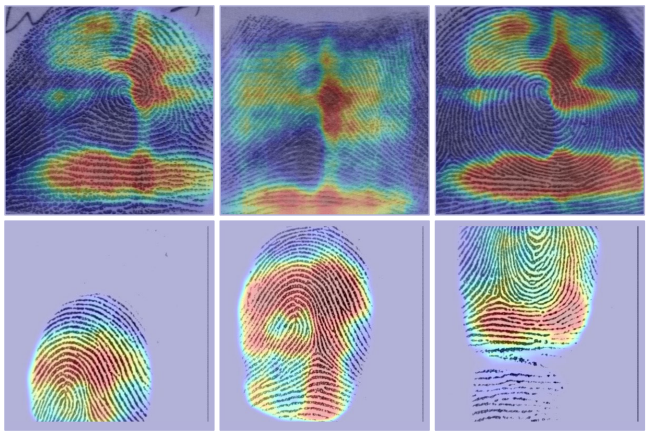


Fig. 14. Examples of direction attention maps for rolled (the first row) and plain (the second row) fingerprints.

G. Limitations

Previous experimental results show our method achieves good performance and increases the following fingerprint indexing and verification accuracy. However, there are still several limitations of our study to be tackled. The estimation of fingerprint center in our method relies on voting area prediction, i.e., segmentation map. But for images with complicated background noise or overlapped fingerprints, inaccurate fingerprint pose may be obtained using our method (the first row in Fig. 15), due to inaccurate prediction of voting area. Besides, deviations may be large even after fingerprint alignment when there exist ridge distortions, and also inaccurate fingerprint pose may be obtained by our method due to strong fingerprint

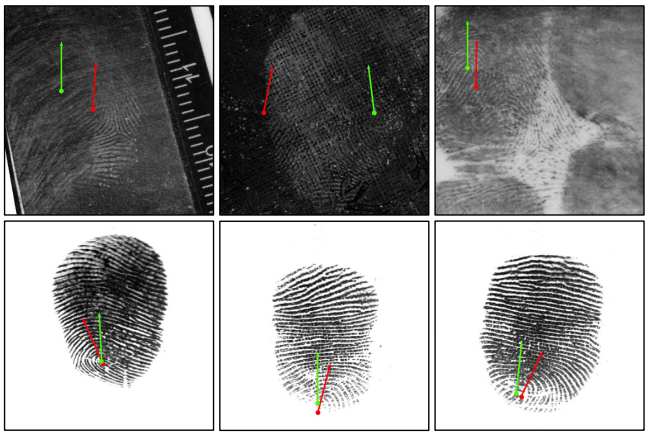


Fig. 15. Failure cases predicted by our method. Green arrows indicate ground truths of fingerprint pose, and red arrows indicate estimations of our method. The first row shows fingerprints with complicated background noise or overlapped area from Hisign latent dataset. And the second row shows fingerprints from FVC2004 DB1A dataset, where inaccurate fingerprint poses were obtained due to severe ridge distortions.

distortion (especially for impressions produced by fingertips), since distortion changes the distribution characteristics of ridge features which are utilized to estimate dense maps in our method (the second row in Fig. 15).

VI. Conclusion

Fingerprint pose can be utilized as constraints to assist fingerprint indexing, verification, and other recognition tasks. However, estimating fingerprint pose precisely is still a challenging task due to various factors such as low ridge quality and partial area. In this paper, we proposed to estimate fingerprint center and direction on various types of fingerprint impressions via dense voting strategy. The proposed approach integrates the strength of deep networks in extracting discriminative features and dense voting strategy to mitigate the impact of noise and incomplete fingerprint area, which is common in plain and latent fingerprints. Besides, fingerprint direction is estimated by dense voting strategy which can increase the robustness of fingerprint direction estimation compared to regressing single direction value directly. Experiments on extensive fingerprint datasets were conducted. Experimental results on these datasets demonstrate the superior performance of our method compared to the SOTA methods, especially for plain and latent fingerprints. Given the fingerprint pose estimations, additional applications of fingerprint pose, e.g., assisting to mitigate fingerprint distortion and extract ridge orientation field, can be further explored in future works. Besides, the proposed method is a relatively general framework, and generalizing the method to other tasks like palmprint or face pose estimation can also be explored in the future.

References

[1] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1701–1708.

[2] J. G. Daugman, "High confidence visual recognition of persons by a test of statistical independence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 11, pp. 1148–1161, Nov. 1993.

[3] D. Maltoni, D. Maio, A. K. Jain, and J. Feng, *Handbook of Fingerprint Recognition*. Cham, Switzerland: Springer, 2022.

[4] R. D. Labati, A. Genovese, V. Piuri, and F. Scotti, "Contactless fingerprint recognition: A neural approach for perspective and rotation effects reduction," in *Proc. IEEE Symp. Comput. Intell. Biometrics Identity Manage. (CIBIM)*, Apr. 2013, pp. 22–30.

[5] C. Stein, C. Nickel, and C. Busch, "Fingerphoto recognition with smartphone cameras," in *Proc. Int. Conf. Biometrics Special Interest Group (BIOSIG)*, Dec. 2012, pp. 1–12.

[6] H. Tan and A. Kumar, "Towards more accurate contactless fingerprint minutiae extraction and pose-invariant matching," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 3924–3937, 2020.

[7] Z. Cui, J. Feng, and J. Zhou, "Monocular 3D fingerprint reconstruction and unwarping," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Jan. 4, 2023, doi: 10.1109/TPAMI.2022.3233898.

[8] R. Cappelli, "Fast and accurate fingerprint indexing based on ridge orientation and frequency," *IEEE Trans. Syst., Man, Cybern., B Cybern.*, vol. 41, no. 6, pp. 1511–1521, Dec. 2011.

[9] Y. Su, J. Feng, and J. Zhou, "Fingerprint indexing with pose constraint," *Pattern Recognit.*, vol. 54, pp. 1–13, Jun. 2016.

[10] K. Cao and A. K. Jain, "Fingerprint indexing and matching: An integrated approach," in *Proc. IEEE Int. Joint Conf. Biometrics (IJCB)*, Oct. 2017, pp. 437–445.

[11] J. Ouyang, J. Feng, J. Lu, Z. Guo, and J. Zhou, "Fingerprint pose estimation based on faster R-CNN," in *Proc. IEEE Int. Joint Conf. Biometrics (IJCB)*, Oct. 2017, pp. 268–276.

[12] J. J. Engelsma, K. Cao, and A. K. Jain, "Learning a fixed-length fingerprint representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 6, pp. 1981–1997, Jun. 2021.

[13] Q. Yin, J. Feng, J. Lu, and J. Zhou, "Joint estimation of pose and singular points of fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 1467–1479, 2021.

[14] S. Gu, J. Feng, J. Lu, and J. Zhou, "Efficient rectification of distorted fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 1, pp. 156–169, Jan. 2018.

[15] S. Gu, J. Feng, J. Lu, and J. Zhou, "Latent fingerprint indexing: Robust representation and adaptive candidate list," *IEEE Trans. Inf. Forensics Security*, vol. 17, pp. 908–923, 2022.

[16] K. Nandakumar, A. K. Jain, and S. Pankanti, "Fingerprint-based fuzzy vault: Implementation and performance," *IEEE Trans. Inf. Forensics Security*, vol. 2, no. 4, pp. 744–757, Dec. 2007.

[17] Y. Sutcu, S. Rane, J. S. Yedidia, S. C. Draper, and A. Vetro, "Feature extraction for a Slepian–Wolf biometric system using LDPC codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Jul. 2008, pp. 2297–2301.

[18] R. Cappelli and D. Maltoni, "On the spatial distribution of fingerprint singularities," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 4, pp. 448–742, Sep. 2008.

[19] K. Nilsson and J. Bigun, "Localization of corresponding points in fingerprints by complex filtering," *Pattern Recognit. Lett.*, vol. 24, no. 13, pp. 2135–2144, 2003.

[20] 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.

[21] M. Liu, X. Jiang, and A. C. Kot, "Fingerprint reference-point detection," *EURASIP J. Adv. Signal Process.*, vol. 2005, no. 4, pp. 1–12, Dec. 2005.

[22] S. Yoon, K. Cao, E. Liu, and A. K. Jain, "LFIQ: Latent fingerprint image quality," in *Proc. IEEE 6th Int. Conf. Biometrics: Theory, Appl. Syst. (BTAS)*, Sep. 2013, pp. 1–8.

[23] K. Rerkrai and V. Areekul, "A new reference point for fingerprint recognition," in *Proc. Int. Conf. Image Process. (ICIP)*, vol. 2, Sep. 2000, pp. 499–502.

[24] V. Areekul, K. Suppasriwasuseth, and S. Jirachaweng, "The new focal point localization algorithm for fingerprint registration," in *Proc. Int. Conf. Pattern Recognit. (ICPR)*, vol. 4, Aug. 2006, pp. 497–500.

[25] C. Deerada, K. Phromsuthirak, A. Rungchokanun, and V. Areekul, "Progressive focusing algorithm for reliable pose estimation of latent fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 1232–1247, 2020.

[26] X. Yang, J. Feng, and J. Zhou, "Localized dictionaries based orientation field estimation for latent fingerprints," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 5, pp. 955–969, May 2014.

[27] C. Lin and A. Kumar, "Matching contactless and contact-based conventional fingerprint images for biometrics identification," *IEEE Trans. Image Process.*, vol. 27, no. 4, pp. 2008–2021, Apr. 2018.

[28] G. T. Candela, *PCASYS—A Pattern-Level Classification Automation System for Fingerprints*. Gaithersburg, MD, USA: National Institute of Standards and Technology, 1995.
[29] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "Filterbank-based fingerprint matching," *IEEE Trans. Image Process.*, vol. 9, no. 5, pp. 846–859, May 2000.
[30] Q. Yin, J. Feng, J. Lu, and J. Zhou, "Orientation field estimation for latent fingerprints by exhaustive search of large database," in *Proc. IEEE 9th Int. Conf. Biometrics Theory, Appl. Syst. (BTAS)*, Oct. 2018, pp. 1–9.
[31] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 1–9.
[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[33] X. Si, J. Feng, B. Yuan, and J. Zhou, "Dense registration of fingerprints," *Pattern Recognit.*, vol. 63, pp. 87–101, Mar. 2017.
[34] K. Cao, E. Liu, and A. K. Jain, "Segmentation and enhancement of latent fingerprints: A coarse to fine ridge structure dictionary," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 9, pp. 1847–1859, Sep. 2014.
[35] 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.
[36] 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)*. Cham, Switzerland: Springer, 2015, pp. 234–241.
[37] H. Law and J. Deng, "CornerNet: Detecting objects as paired keypoints," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 734–750.
[38] J. Bigun and G. H. Granlund, "Optimal orientation detection of linear symmetry," in *Proc. IEEE Int. Conf. Comput. Vis.*, May 1987, pp. 433–438.
[39] *VeriFinger SDK 12.0*. Accessed: Apr. 18, 2023. [Online]. Available: https://www.neurotechnology.com/verifinger.html
[40] 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.
[41] (2000). *FVC 2000: The First International Competition for Fingerprint Verification Algorithms*. [Online]. Available: http://bias.csr.unibo.it/fvc2000/databases.asp
[42] 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.

Yongjie Duan received the B.S. degree from the Department of Automation, Tsinghua University, Beijing, China, in 2017, where he is currently pursuing the Ph.D. degree under the supervision of Prof. Jie Zhou. His research interests include fingerprint recognition, human–computer interaction, 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.

Jiwen Lu (Senior Member, IEEE) received the B.Eng. degree in mechanical engineering and the M.Eng. degree in electrical engineering from the Xi'an University of Technology, Xi'an, China, in 2003 and 2006, respectively, and the Ph.D. degree in electrical engineering from Nanyang Technological University, Singapore, in 2012. He is currently an Associate Professor with the Department of Automation, Tsinghua University, Beijing, China. His current research interests include computer vision, pattern recognition, and machine learning. He has authored/coauthored more than 300 scientific papers in these areas, where more than 100 of them are IEEE TRANSACTIONS articles and more than 100 of them are CVPR/ICCV/ECCV papers. He is a member of the Image, Video and Multidimensional Signal Processing Technical Committee, the Multimedia Signal Processing Technical Committee, and the Information Forensics and Security Technical Committee of the IEEE Signal Processing Society. He is also a member of the Multimedia Systems and Applications Technical Committee and the Visual Signal Processing and Communications Technical Committee of the IEEE Circuits and Systems Society, respectively. He is a fellow of IAPR. He was a recipient of the National Outstanding Youth Foundation of China Award. He serves the Co-Editor-of-Chief for the *Pattern Recognition Letters* and an Associate Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, the IEEE TRANSACTIONS ON BIOMETRICS, BEHAVIOR, AND IDENTITY SCIENCE, and *Pattern Recognition*.

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 was a Post-Doctoral Fellow with the Department of Automation, Tsinghua University, Beijing, China, where he has been a Full Professor since 2003. In recent years, he has authored more than 300 papers in peer-reviewed journals and conferences. Among them, more than 100 papers have been published in top journals and conferences, such as the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON IMAGE PROCESSING, and CVPR. His research interests include computer vision, pattern recognition, and image processing. He is a fellow of IAPR. 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 and two other journals.