

Latent Orientation Field Estimation via Convolutional Neural Network

Kai Cao and Anil K. Jain

Department of Computer Science and Engineering
Michigan State University, East Lansing, Michigan 48824

Email: {kaicao, jain}@cse.msu.edu

Abstract

The orientation field of a fingerprint is crucial for feature extraction and matching. However, estimation of orientation fields in latents is very challenging because latents are usually of poor quality. Inspired by the superiority of convolutional neural networks (ConvNets) for various classification and recognition tasks, we pose latent orientation field estimation in a latent patch to a classification problem, and propose a ConvNet based approach for latent orientation field estimation. The underlying idea is to identify the orientation field of a latent patch as one of a set of representative orientation patterns. To achieve this, 128 representative orientation patterns are learnt from a large number of orientation fields. For each orientation pattern, 10,000 fingerprint patches are selected to train the ConvNet. To simulate the quality of latents, texture noise is added to the training patches. Given image patches extracted from a latent, their orientation patterns are predicted by the trained ConvNet and quilted together to estimate the orientation field of the whole latent. Experimental results on NIST SD27 latent database demonstrate that the proposed algorithm outperforms the state-of-the-art orientation field estimation algorithms and can boost the identification performance of a state-of-the-art latent matcher by score fusion.

1. Introduction

Fingerprint orientation field, which is essential for ridge structure enhancement, feature extraction, pattern type classification, indexing and matching [13], plays an important role in fingerprint identification. However, estimating orientation fields in latent fingerprints (i.e., impressions lifted from the surfaces of objects typically at crime scenes) is a challenging problem due to poor quality of latents in terms of ridge clarity and complex background noise. Consequently, the accuracy of latent identification is significantly lower compared to that of rolled/slapped fingerprints. According to NIST evaluation [11], the best performing latent Automated Fingerprint Identification System (AFIS) achieved only 67.2% rank-1 identification rate in identifying 1,066 latent prints against reference prints from 100,000 subjects.

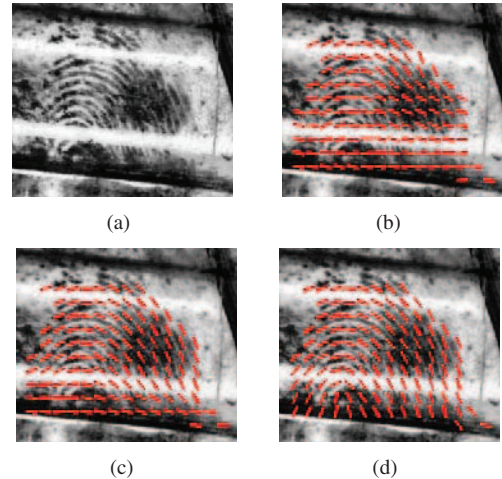


Figure 1. A comparison of latent orientation fields estimated by three different algorithms. (a) A cropped latent fingerprint, (b)-(d) are orientation fields estimated by global orientation patch dictionary [8], ridge structure dictionary [6], and the proposed algorithm on manually marked region of interest (ROI). Note that when a latent overlaps with strong background noise, global orientation patch dictionary and ridge structure dictionary approaches do not work well.

Furthermore, the performance of all the matchers evaluated by NIST decreased consistently as lower quality latents were searched. As demonstrated in [8] [16], better orientation field estimation can improve the identification rate of commercial off-the-shelf (COTS) algorithms. Therefore, extracting accurate latent orientation field is crucial to improve latent identification performance in a lights-out mode¹.

There are a number of approaches [7] [10] for estimation of rolled/slapped fingerprint orientation fields by analyzing information in a local region (e.g., 16×16 or 32×32 pixels). However, these approaches do not work well on latents because (i) the local fingerprint information is sensitive to image quality, and (ii) no prior knowledge of orientation pattern or ridge structure in a larger neighborhood

¹In a lights-out mode, AFIS conducts feature extraction and template comparison automatically, without any human intervention.

is used. Recently, a few dictionary-based approaches have been proposed to improve latent orientation field estimation. In [8], local Fourier analysis was used to estimate the initial orientation field. The orientation patch dictionary (called global orientation patch dictionary compared to localized orientation field dictionary proposed later) was then utilized to correct the noisy initial orientation field. Experimental results on a latent database confirmed that the use of this prior knowledge is helpful. However, the initial orientation field was obtained by a conventional approach which itself is sensitive to noise. Therefore, this approach may not be useful when the latent is very noisy or overlaps with strong background noise (as shown in Fig.1 (b)). Later, a localized dictionary based approach [16] was proposed by first registering the latent and then using localized orientation patch dictionaries to correct the initial orientation field. The resulting orientation field by this approach is much better than that obtained by global orientation patch dictionary. However, the registration step again is based on the initial orientation field which is still not reliable. Instead of using orientation patch dictionaries, the approach in [6] learnt ridge structure dictionaries directly from fingerprint patches for orientation and frequency fields estimation, and quality estimation. Even though a smaller patch (64×64 pixels) was used (compared to 160×160 pixels in [8]), the enhancement of latents based on ridge structure dictionaries was shown to be better than global orientation patch dictionary in terms of identification accuracy. However, all the dictionary based approaches above have a common drawback that the dictionaries were learnt from high quality fingerprints and may not work well on very poor quality latents (see Figs. 1 (b) and (c)). Therefore, an efficient approach which can directly learn orientation fields on large fingerprint patches of poor quality is highly desirable.

Convolutional Neural Networks (ConvNets) have the capability to learn distinctive features directly from the input images. In the past few years, ConvNets have delivered outstanding performance on different challenging classification tasks [12] [14]. Krizhevsky *et al.* [12] achieved the best top-5 error rate of 15.3% on ImageNet 2012 classification benchmark, compared to 26.2% by the second place entry. In [14], the face recognition accuracy outperformed the human-level performance on the challenging Labeled Faces in the Wild (LFW) benchmark. Compared to the first ConvNet proposed in 1990s, the following factors contribute to their remarkable success and reemergence in the past few years [17]: (i) the availability of very large training datasets, (ii) powerful and affordable computational resources, such as GPUs, and (iii) better regularization strategies, such as ReLU nonlinearity and dropout.

Inspired by the success of ConvNets in a variety of classification tasks, we propose to classify the orientation field of a latent patch as one of a set of representative orientation patterns using a ConvNet. To achieve this, a set of learnt orientation patterns (128 used in this paper) is first learnt from orientation patches selected from the NIST

SD4 [3] database which contains 400 rolled prints for each of the five fingerprint types. For each orientation pattern class, 10,000 fingerprint patches are selected from a different rolled fingerprint database, NIST SD14 [2], to train the ConvNet. Thus, for all 128 representative orientation patterns, we have a total of $128 \times 10,000 = 1.28$ million training patches (patch size: 96×96 pixels). Before feeding to ConvNet, these fingerprint patches are corrupted by noise to simulate latents. This is done because of a lack of large public domain latent fingerprint database. Given a latent query image, (i) a preprocessing step is used to remove large scale background noise (also called the cartoon component) and enhance the potential ridge structure of the remaining texture component in the latent; (ii) each latent patch is fed to the trained ConvNet to predict its orientation pattern; (iii) the predicted orientation patterns of all the latent patches are quilted together to construct an orientation field for the whole latent; (iv) directional filtering and gradient based approach are then used to enhance the ridge structure and obtain the final latent orientation field, and (v) Gabor filtering with estimated orientation field and fixed ridge frequency is used to enhance the latent. Experimental results on the NIST SD27 latent database show that the performance of the proposed ConvNet based approach outperforms state-of-the-art algorithms in terms of orientation field estimation and latent identification accuracy.

The contributions of this paper are summarized as follows:

- 1) Posing orientation field estimation of a latent patch to a classification problem,
- 2) training a ConvNet and using the trained network for latent orientation field estimation.
- 3) A method to simulate texture noise present in latents.
- 4) A preprocessing method to enhance the potential ridge structure of poor quality latents.
- 5) An improvement of 7.36% at rank-1 by fusing three comparison scores obtained by an AFIS after inputting the original latent image, enhanced latent images by ridge structure dictionary and the proposed ConvNet based algorithm against 100K background database.

2. Proposed orientation field estimation

The basic idea is to predict the orientation pattern for a given latent patch using a ConvNet. In order to achieve this, a set of representative orientation patterns are first learnt. A large number of fingerprint patches for each orientation pattern are then selected to train the ConvNet. Fig. 2 shows a flowchart of the proposed algorithm.

2.1. Training data preparation

Rolled fingerprint database NIST SD4 [3] is used to learn the representative orientation pattern because it has equally distributed fingerprint types (400 fingerprints for each of the five types, namely arch, tented arch, left loop, right loop and

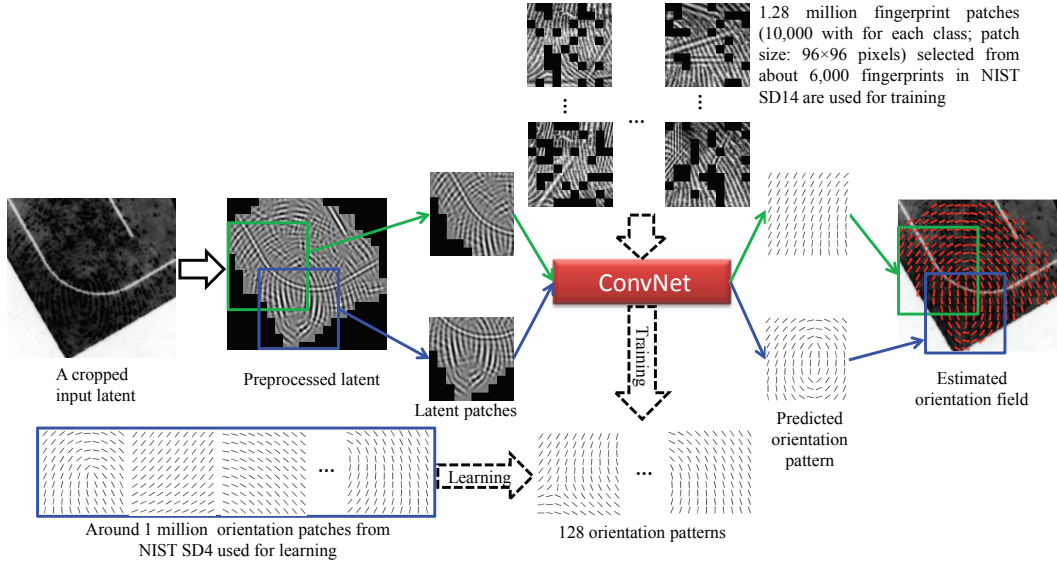


Figure 2. A flowchart of the proposed ConvNet based orientation field estimation algorithm. The dotted arrows show the offline training process, while solid arrows show the online process for latent orientation field estimation.

whorl). The orientation field and quality map² with block size 8×8 pixels of a fingerprint image are obtained using the NIST public domain software [9]; it is downsampled by a factor of 2 because 16×16 block size is the typical size for orientation field estimation [10]. An orientation patch with size of 10×10 blocks is selected if the minimum quality covered by this patch is larger than 1 because we do not want the training patches to include any background. A set of about 1 million orientation patches (S_θ) is extracted from 2,000 file fingerprints in NIST SD4. A fast K-means clustering approach [5] is adopted to cluster S_θ into N ($N = 128$ in this paper) clusters. Fig. 3 shows a subset of the learnt representative orientation patterns (i.e., cluster centers).

To utilize the representation and classification power of ConvNet, a large number of fingerprint patches from another larger rolled print database NIST SD14 [2] are selected for each of the N orientation patterns to train the ConvNet. The orientation fields and quality maps with block size 16×16 pixels of fingerprints from NIST SD14 are obtained using the NIST public domain software and downsampling. A fingerprint patch (160×160 pixels) in the foreground is included into the i^{th} training subset if (i) the minimum quality covered by the patch is larger than 1 and (ii) the most similar representative orientation pattern of its orientation field θ_f is the i^{th} orientation pattern θ_i . The similarity $S(\theta_1, \theta_2)$ between two orientation patches θ_1 and θ_2 is defined as

$$S(\theta_1, \theta_2) = \frac{1}{2} \sum_{k=1}^n \cos^2(\theta_{1,k} - \theta_{2,k}) \quad (1)$$

²A quality value is assigned to each block with 0 being the lowest and 4 the highest.

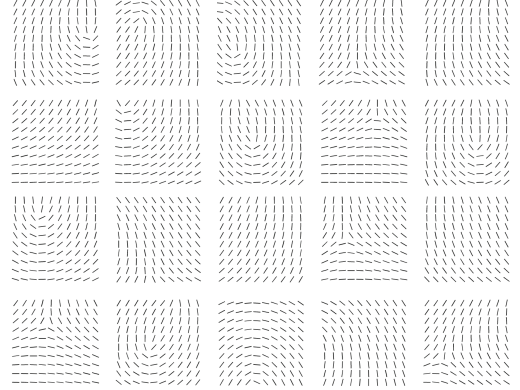


Figure 3. A subset of the 128 representative orientation pattern types learnt from the NIST SD4 database. An orientation pattern contains 10×10 orientation elements, and an orientation element denotes an orientation in a block of 16×16 pixels. So, the patch size is 160×160 .

where n is the number of orientation elements in an orientation patch, and $\theta_{1,k}$ and $\theta_{2,k}$ are the k^{th} orientation elements of θ_1 and θ_2 , respectively. In this way, 1.28 million fingerprint patches (10,000 for each of the $N = 128$ orientation pattern types) from about 6,000 fingerprints of NIST SD14 are selected for ConvNet training.

2.2. Noise simulation

The fingerprint patches used to train the ConvNet are generally of good quality. However, latent patches (input to the trained ConvNet) are likely to be of poorer quality in terms of ridge clarity and background noise. While ridge clarity can be somewhat enhanced by bandpass filtering, directional filtering and root filtering [7], mitigation of

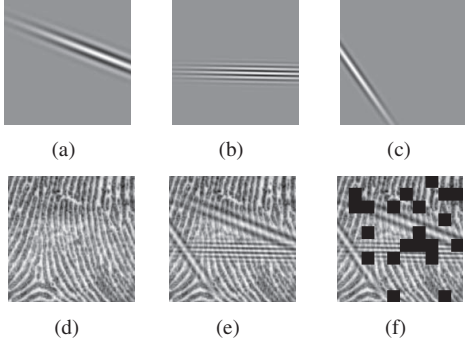


Figure 4. Illustration of the proposed method for latent noise simulation in training patches: (a)-(c) three texture noise images generated by Eq. (3), (d) a fingerprint patch, (e) a corrupted fingerprint patch with texture in (a)-(c), and (f) around 30% blocks in (e) are set to zero.

| Parameters | θ | ψ | f | σ_x | σ_y | x_0, y_0 |
|------------|------------|-------------|---------------|------------|------------|------------|
| Range | $[0, \pi)$ | $[0, 2\pi)$ | $[0.05, 0.2]$ | $[2, 8]$ | $[10, 50]$ | $[0, 160]$ |

Table 1. Ranges of parameters in Gabor function (3).

background noise, especially line-like texture noise, is more challenging. In order to make the training set have similar characteristics as latent patches, simulated line-like texture noise are then added to the training patches. A line-like texture noise image G is defined using a Gabor function as

$$W_{\theta, \sigma_x, \sigma_y, x_0, y_0} = e^{-\left(\frac{x_{\theta}^2}{2\sigma_x^2} + \frac{y_{\theta}^2}{2\sigma_y^2}\right)} \quad (2)$$

$$G_{\theta, f, \sigma_x, \sigma_y, \psi, x_0, y_0} = W_{\theta, \sigma_x, \sigma_y, x_0, y_0} \cos(2\pi f x_{\theta} + \psi), \quad (3)$$

where

$$x_{\theta} = x \cos(\theta) + y \sin(\theta) - x_0, \quad (4)$$

$$y_{\theta} = -x \sin(\theta) + y \cos(\theta) - y_0, \quad (5)$$

and $\theta, f, \sigma_x, \sigma_y, \psi, x_0$ and y_0 are randomly generated parameters (see Table 1).

Suppose that G_i is the i^{th} texture noise image from Eq. (3) and W_i is the weight image from Eq. (2). Given a fingerprint patch P , the corrupted patch P' is obtained by adding N_p texture noise images $\{G_i\}_{i=1}^{N_p}$ by,

$$P' = (P + \sum_{i=1}^{N_p} \lambda G_i) ./ (1 + \sum_{i=1}^{N_p} \lambda W_i) \quad (6)$$

where λ controls the amplitude of the texture noise, and $./$ denotes pixel-wise division. Fig. 4 illustrates this process. For a latent patch, not all the blocks are in the ROI, so, we randomly set about 30% of the blocks to zero. Fig. 4 (f) shows a an example noisy patch that is fed to the ConvNet for training. Note that the simulated noise for the same fingerprint patch is different at each epoch of the ConvNet.

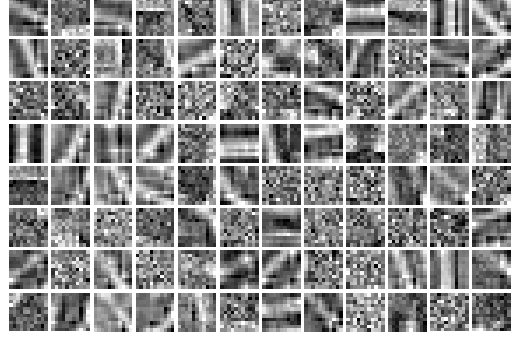


Figure 6. 96 filters in the first convolutional layer of the trained ConvNet.

2.3. ConvNet architecture

The ConvNet is trained to classify a fingerprint patch as one of the N representative orientation patterns. The overall architecture used in this paper is shown in Fig. 5. A fingerprint patch of size 160×160 pixels is input to a convolutional layer (C1) with 96 filters of size 11×11 and stride of 4 (denoted by $96 \times 11 \times 11 \times 1(4)@160 \times 160$). The ReLU activation function ($f(x) = \max(0, x)$), which has been shown to have better fitting abilities than the sigmoid function [12], is adopted to model the output of the neurons. The resulting 96 feature maps (of size 38×38) are then separately fed to a max-pooling layer (M2) which takes the max over 3×3 spatial neighborhoods with a stride of 2. We denote this max-pooling layer by $96 \times 3 \times 3 \times 96(2)@38 \times 38$. A local response normalization [12] is then applied. Similar definitions are used for layers C3, M4, and C5 (see Fig. 5 for additional details).

A fully-connected layer (F6) follows the last convolutional layer (C5). After F6, a dropout regularization method sets the output to zero with probability 0.5 to encourage sparsity of the neurons and to avoid overfitting. The output layer (O7) of the ConvNet, an N -way softmax to predict the probability distribution over N different classes, is defined as

$$y_i = \frac{\exp(x_i)}{\sum_{j=1}^N \exp(x_j)} \quad (7)$$

where x_i and y_i are the input and output, respectively, of the i^{th} neuron at the output layer.

The toolbox MatConvNet [1] is used to implement the ConvNet architecture. Offline training of the ConvNet is conducted on a Linux server with Tesla K20 GPUs and takes 4 days for 20 epochs. Fig. 6 shows the 96 filters at the first convolutional layer of the trained ConvNet.

2.4. Orientation field estimation based on ConvNet

2.4.1 Latent preprocessing

A latent may overlap with other background texture of different scale. The nonlinear decomposition method based on local total variation (LTV) [6] is utilized to remove

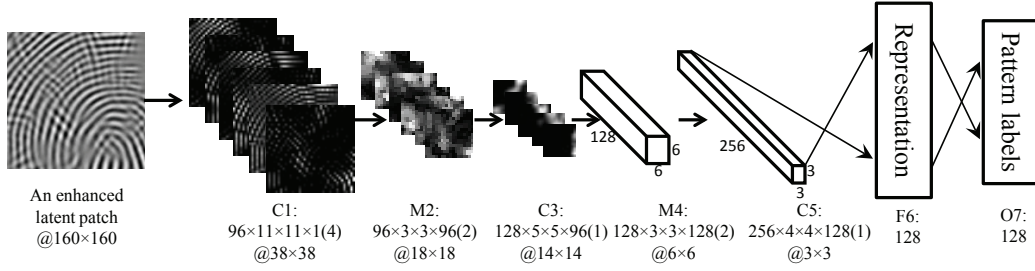


Figure 5. Outline of the ConvNet architecture. There are three convolutional layers (C1, C3 and C5), two max-pooling layers (M2 and M4), a fully-connected layer (F6) and an output layer (O7). The ReLU activation function is used for each layer except the output layer. Dropout regularization is used in F6 to encourage sparsity of the neurons and to avoid overfitting.

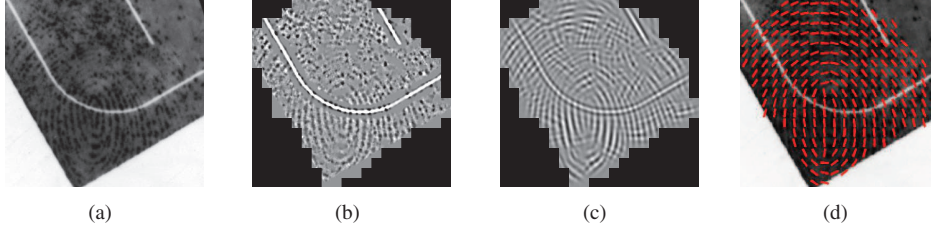


Figure 7. Illustration of the main steps in orientation field estimation: (a) a cropped latent, (b) the texture component of (a), (c) preprocessed latent, and (d) orientation field estimated by ConvNet. Note that only the blocks in the manually marked ROI of the latent are considered.

the cartoon component (or large scale background noise) while keeping the texture component representing the friction ridge information. However, line-like structured noise and other finer texture noise still remain in the texture component (as shown in Fig. 7 (b)). The following steps are executed to enhance the potential ridge structure in manually marked ROI and suppress other textured background noise (Fig.7 (c)):

- (1) The texture component is divided into non-overlapping blocks (of size 16×16 pixels). Each block b is extended to B (of size 64×64 pixels) by placing the local image in the 64×64 window, followed by multiplication by a spectral window (Fig. 8 (b)) [7].
- (2) B is transformed to the Fourier domain and a bandpass filter (Fig. 8 (e)) is used to remove both high frequency and low frequency noise. The response is denoted by \tilde{B} .
- (3) A set of 12 directional filters $\{\psi_i\}_{i=1}^{12}$ (shown in Fig.9 (a)) is used to obtain the responses $\{\tilde{B}_i = \tilde{B}\psi_i\}_{i=1}^{12}$ in the 12 orientations, where ψ_i at orientation ϕ is defined as

$$\psi_i(\phi) = \begin{cases} \cos^2 2(\phi - c_i), & \text{if } |\phi - c_i| < \frac{\pi}{4} \\ 0, & \text{otherwise} \end{cases} \quad (8)$$

where $c_i = \frac{(i-1)\pi}{12}$ is the i^{th} orientation center. Fig. 9 (b) shows the energy magnitude in $\{\tilde{B}_i\}_{i=1}^{12}$.

- (4) The two orientations with the highest responses are selected. Suppose that $\{\tilde{B}_{s_j}\}_{j=1}^2$ are the responses corresponding to the selected orientations. For each of

them, the enhanced $\tilde{B}_{s_j}^e$ is obtained by root filtering. Formally,

$$\tilde{B}_{s_j}^e = \tilde{B}_{s_j} \cdot |\tilde{B}_{s_j}|, \quad (9)$$

where \cdot and $|\cdot|$ are pixel-wise product and absolute value, respectively. Figs. 8 (g) and (h) show the enhanced blocks for these two directions. The enhanced block B' is obtained by

$$B' = FFT^{-1} \left(\frac{\tilde{B}_{s_1}^e}{2E_{s_1}} + \frac{\tilde{B}_{s_2}^e}{2E_{s_2}} \right) \quad (10)$$

where E_{s_1} and E_{s_2} are the energies of $\tilde{B}_{s_1}^e$ and $\tilde{B}_{s_2}^e$, respectively, and FFT^{-1} is the inverse Fourier transform. Fig. 8 (i) shows the final enhanced block. Note that only one direction is selected if there is only one local maximum.

- (5) The enhanced blocks are then combined to generate the whole enhanced latent as shown in Fig. 7 (c). Note that the potential ridge structure of the latent is enhanced while the line-like noise is suppressed.

2.4.2 Orientation field estimation

The preprocessed latent is divided into overlapping patches (160×160 pixels). Each patch is directly fed into the trained ConvNet which outputs the label of its orientation pattern and the associated probability (Eq. (7)) or confidence level. For each block b in the foreground of a latent, let $\{\theta_i, p_i\}$ be the orientation and probability of the i^{th} patch covering

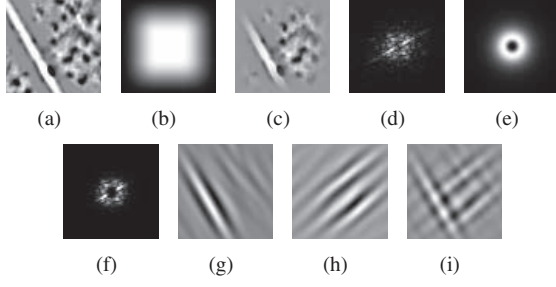


Figure 8. Illustration of latent block enhancement (block size: 64×64 pixels). (a) Texture component of a latent block, (b) spectral window, (c) weighted image of (a), (d) magnitude of the Fourier transform of (c), (e) bandpass filter, (f) magnitude of the Fourier transform after applying (e) to (d), (g) and (h) are two enhanced blocks using directional filtering and root filtering, and (i) fusion of (g) and (h).

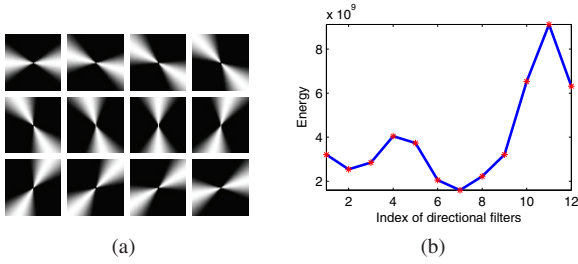


Figure 9. (a) Directional filters along the 12 orientations and (b) energies after applying 12 directional filters in (a) to Fig. 8 (f).

block b . Orientation θ_b is given as:

$$\theta_b = \frac{1}{2} \tan^{-1} \left(\frac{\sum_{i=1}^{n_b} p_i \sin 2\theta_i}{\sum_{i=1}^{n_b} p_i \cos 2\theta_i} \right), \quad (11)$$

where n_b is the number of patches covering block b .

2.4.3 Enhancement and postprocessing

Due to the limited number of orientation pattern classes, the estimated orientation field may not be very accurate. Hence, directional filtering is used to enhance the latent based on the estimated orientation field; the gradient based approach in [10] is adopted to obtain the final orientation field. Fig. 7 (d) shows the final orientation field of an example latent. Gabor filtering [10] tuned by the final orientation field and a fixed ridge frequency (set to 0.12 in this paper) is applied to the texture component of the latent to obtain the enhanced latent which is the input to AFIS.

3. Experiments

We compare the proposed ConvNet based algorithm with other state-of-the-art algorithms in terms of orientation field accuracy and identification accuracy of the enhanced latents resulting from the estimated orientation field. The latent database NIST SD27 [4] (258 latents and mated rolled

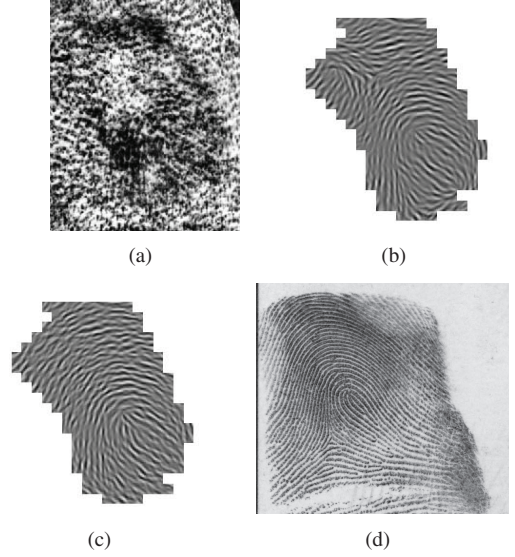


Figure 11. Inputs to COTS matcher. (a) Original latent, (b) and (c) latents enhanced by [6] and the proposed algorithm, respectively, and (d) mated rolled print. The mated print is retrieved at ranks 100,000 (score=0), 68 and 1 when (a), (b) and (c) are fed to the AFIS.

prints) is used in our experiments. According to subjective classification by fingerprint examiner(s), out of the 258 latents, 88 were identified to be of good quality, 85 of bad quality and 85 of ugly quality.

3.1. Accuracy of orientation field estimation

The ground truth orientation fields and manually marked ROI were provided in [8]. The accuracy of an orientation field estimation algorithm is measured by the average Root Mean Square Deviation (RMSD) [15]. Table 2 compares the performance of proposed ConvNet based approach with: (1) global orientation patch dictionary [8] (GlobalDic), (2) localized orientation patch dictionary [16] (LocalizedDic), and (3) ridge structure dictionary [6] (RidgeDic). Fig. 10 compares the orientation fields from different algorithms on three different quality latents visually. We observe that: (i) the proposed algorithm outperforms the other three state-of-the-art algorithms on latents of different quality levels; (ii) it is even slightly better than [16] with manually marked pose; (iii) noise simulation in training set is helpful to improve the accuracy of orientation field estimation, especially for ugly quality latents.

3.2. Identification performance

A state-of-the-art commercial off-the-shelf (COTS) latent matcher (one of the top performing AFIS in the NIST evaluation [11]) is adopted for the identification experiments. In addition to the mated rolled prints of latents in NIST SD27, we also include additional rolled prints to enlarge the background database to 100,000 reference prints. For each latent, we feed three inputs (see Fig. 11) to the

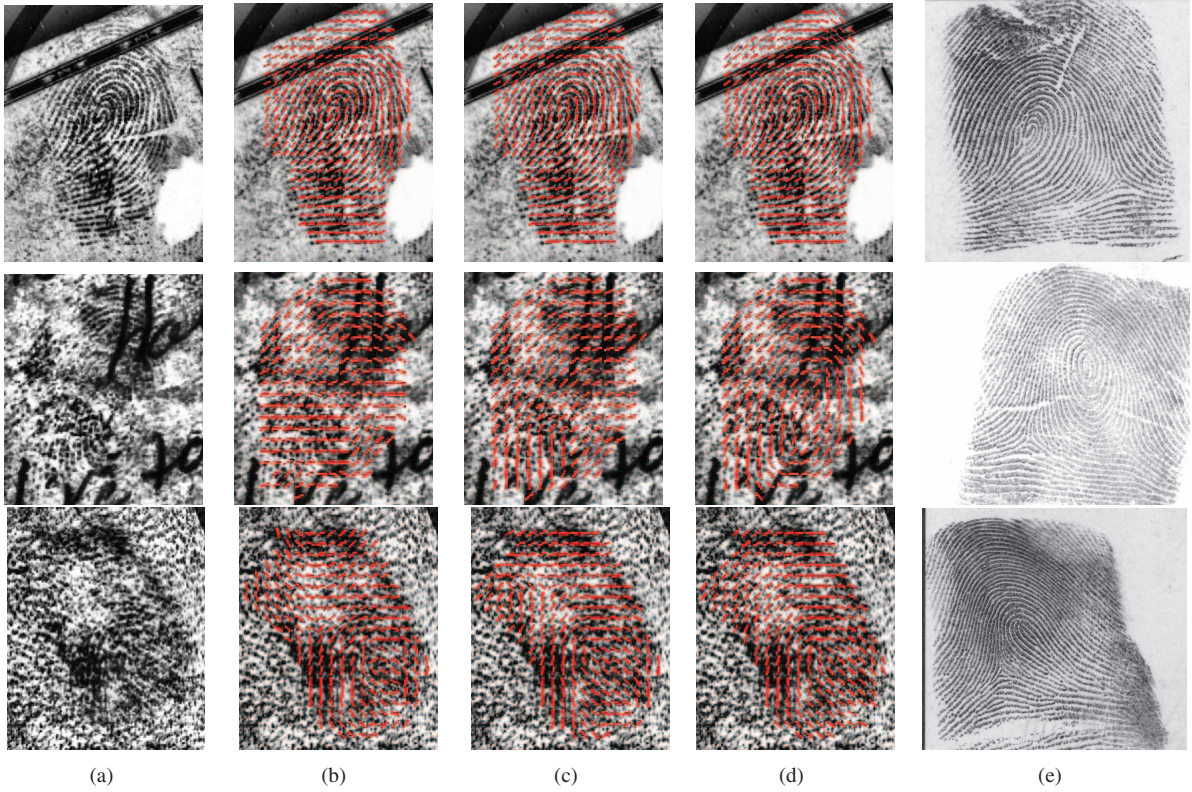


Figure 10. Comparison of orientation fields obtained by three different algorithms on three different cropped latents shown in column (a). (b)-(d) Orientation fields obtained by ridge structure dictionary [6], global orientation patch dictionary [8] and proposed ConvNet based algorithm, respectively, and (e) the mated rolled prints of (a). The contrast of these latents has been enhanced for better visual quality.

| Algorithm | All | Good | Bad | Ugly |
|--|-------|-------|-------|-------|
| LocalizedDict [16] (manually marked pose) | 13.76 | 10.87 | 14.12 | 16.40 |
| LocalizedDic [16] | 14.35 | 11.15 | 15.15 | 16.85 |
| GlobalDic [8] | 18.44 | 14.40 | 19.18 | 21.88 |
| RidgeDic [6] | 19.53 | 15.34 | 20.70 | 22.68 |
| Proposed (without noise simulation) | 14.44 | 11.42 | 14.61 | 17.41 |
| Proposed (with noise simulation) | 13.51 | 10.76 | 13.94 | 16.00 |

Table 2. Average Root Mean Square Deviation (RMSD) of different orientation field estimation algorithms on NIST SD27 latent database.

COTS matcher: (1) original latent image (Original image), (2) latent enhanced using the orientation and frequency fields [6] (Ridge structure dictionary) and (3) latent enhanced using the orientation field by the proposed algorithm and fixed ridge frequency (Proposed). The AFIS scores from these three different inputs are fused using the sum rule (Fusion). Since the proposed algorithm only focuses on orientation field estimation, we use manually marked ROI for (2) and (3) for a fair comparison.

The Cumulative Match Characteristic (CMC) curves of the four scenarios on 258 latents as well as three quality subsets in NIST SD27 are shown in Fig. 12. Even with

a fixed ridge frequency, the proposed algorithm performs consistently better than the ridge structure dictionary based method [6]. The rank-1 identification rate of the proposed algorithm is 6.20% higher than the results from ridge structure dictionary. This improvement is attributed to the accurate orientation field estimated by the proposed ConvNet based algorithm. After fusing the AFIS scores from all the three inputs (see Figs. 11 (a), (b) and (c)), the overall latent identification performance is significantly improved. The rank-1 identification rates on all 258 latents, good latents, bad latents and ugly latents are improved by 7.36%, 4.55%, 7.06% and 10.59%, respectively; there are now 19 additional latents whose true mates can be correctly retrieved at rank-1

4. Summary and future work

Orientation field estimation is a critical step to a robust lights-out latent fingerprint identification system. While dictionary based approaches have shown some success for orientation field estimation, they have the following drawbacks: (1) the initial orientation field used in orientation patch dictionary construction itself is not reliable, (2) extending the ridge structure dictionary with large patch size results in too many dictionary elements and reduces their

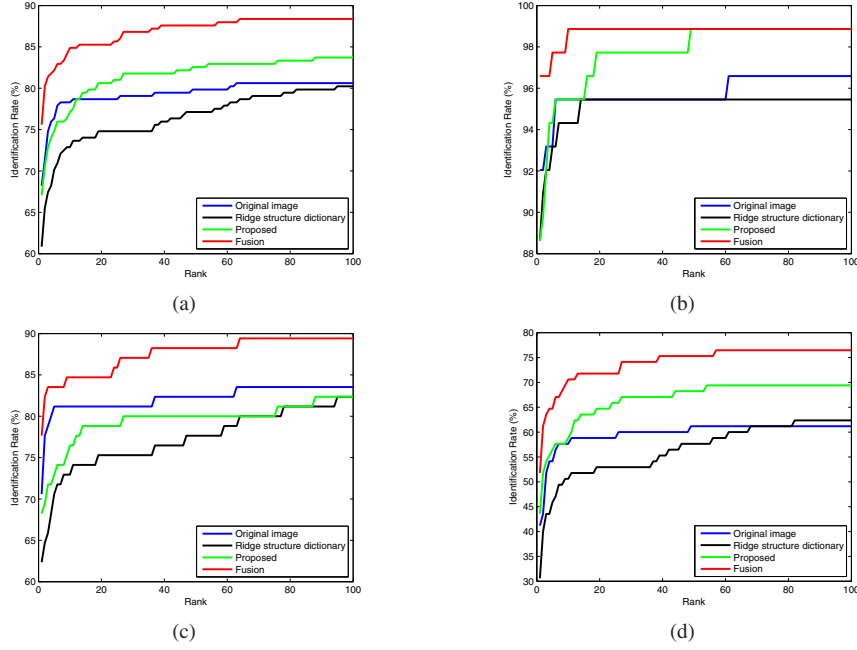


Figure 12. Identification performance (CMC curves) of different algorithms on (a) all 258 latents, (b) 88 good quality latents, (c) 85 bad quality latents, and (d) 85 ugly quality latents in NIST SD27 latent database. Original image (blue): input to AFIS is only the raw latent image; Ridge structure dictionary (black): latent is enhanced by the method in [6]; Proposed (green): latent is processed by the proposed ConvNet based algorithm; Fusion (red): sum score fusion of three scores output by a state-of-the-art AFIS.

efficacy, (3) only a relatively small number of fingerprints have been used for training in dictionary based approaches, and (4) the dictionaries, typically learnt from high quality fingerprints, may not work well on poor quality latents. In this paper, we have proposed a ConvNet based fingerprint orientation field estimation algorithm by posing orientation field estimation in a latent patch as a classification task. Experimental results on NIST SD27 latent database demonstrate the superior performance of the proposed method. This can be attributed to: (1) the inherent power of ConvNet for learning a robust representation for classification, (2) a large number fingerprint patches with simulated noise used for training the ConvNet, and (3) a preprocessing method to enhance potential ridge structure and suppress texture noise.

Future work will include evaluating the effect of (1) orientation pattern size, and (2) noise level added to the training patches, and (3) estimating the ridge frequency field using the same ConvNet based framework.

References

- [1] MatConvNet. <http://www.vlfeat.org/matconvnet/>.
- [2] NIST Special Database 14. <http://www.nist.gov/srd/niststd14.cfm>.
- [3] NIST Special Database 4. <http://www.nist.gov/srd/niststd4.cfm>.
- [4] NSIT Special Database 27. <http://www.nist.gov/srd/niststd27.cfm>.
- [5] D. Cai. Litekmeans: the fastest matlab implementation of kmeans. Available at: <http://www.zjucadcg.cn/dengcai/Data/Clustering.html>, 2011.
- [6] K. Cao, E. Liu, and A. Jain. Segmentation and enhancement of latent fingerprints: A coarse to fine ridgestructure dictionary. *IEEE TPAMI*, 36(9):1847–1859, Sept 2014.
- [7] S. Chikkerur, A. N. Cartwright, and V. Govindaraju. Fingerprint enhancement using STFT analysis. *Pattern Recognition*, 40(1):198–211, 2007.
- [8] J. Feng, J. Zhou, and A. K. Jain. Orientation field estimation for latent fingerprint enhancement. *IEEE TPAMI*, 54(4):925–940, 2013.
- [9] M. D. Garriss, E. Tabassi, C. I. Wilson, R. M. McCabe, S. Janet, and C. I. Watson. NIST fingerprint image software 2. 2004.
- [10] L. Hong, Y. Wan, and A. Jain. Fingerprint image enhancement: Algorithm and performance evaluation. *IEEE TPAMI*, 20(8):777–789, 1998.
- [11] M. D. Indovina, V. Dvornychenko, R. A. Hicklin, and G. I. Kiebusinski. Evaluation of latent fingerprint technologies: Extended feature sets (evaluation 2). *Technical Report NISTIR 7859*, 2012.
- [12] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012.
- [13] D. Maltoni, D. Maio, A. Jain, and S. Prabhakar. *Handbook of Fingerprint Recognition (Second Edition)*. Springer, 2009.
- [14] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In *NIPS*, pages 1988–1996, 2014.
- [15] F. Turrone, D. Maltoni, R. Cappelli, and D. Maio. Improving fingerprint orientation extraction. *IEEE TIFS*, 6(3):1002–1013, 2011.
- [16] X. Yang, J. Feng, and J. Zhou. Localized dictionaries based orientation field estimation for latent fingerprints. *IEEE TPAMI*, 36(5):955–969, May 2014.
- [17] M. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *ECCV*, pages 818–833, 2014.