Automatic Latent Fingerprint Segmentation

Dinh-Luan Nguyen, Kai Cao and Anil K. Jain
Michigan State University
East Lansing, Michigan, USA
nguye590@msu.edu, {kaicao, jain}@cse.msu.edu

Abstract

We present a simple but effective method for automatic latent fingerprint segmentation, called SegFinNet. SegFinNet takes a latent image as an input and outputs a binary mask highlighting the friction ridge pattern. Our algorithm combines fully convolutional neural network and detection-based approaches to process the entire input latent image in one shot instead of using latent patches. Experimental results on three different latent databases (i.e. NIST SD27, WVU, and an operational forensic database) show that SegFinNet outperforms both human markup for latents and the state-of-the-art latent segmentation algorithms. We further show that this improved cropping boosts the hit rate of a latent fingerprint matcher.

1. Introduction

Latent fingerprints, also known as fingermarks, are friction ridge impressions formed as a result of someone touching a surface, particularly at a crime scene. Latents have been successfully used to identify suspects in criminal investigations for over 100 years by comparing the similarity between latent and rolled fingerprints in a reference database [12]. Latent cropping (segmentation) is the crucial first step in the latent recognition algorithm. For a given set of latent enhancement, minutiae extraction, and matching modules, different cropping masks for friction ridges can lead to dramatically different recognition accuracies. Unlike rolled/slap fingerprints, which are captured in a controlled setting, latent fingerprints are typically noisy, distorted and have low ridge clarity. This creates challenges for an accurate automatic latent cropping algorithm.

We map the latent fingerprint cropping problem to a sequence of computer vision tasks as follow: (a) *Object detection* [17] as friction ridge localization; (b) *Semantic segmentation* [14] as separating all possible friction ridge pat-
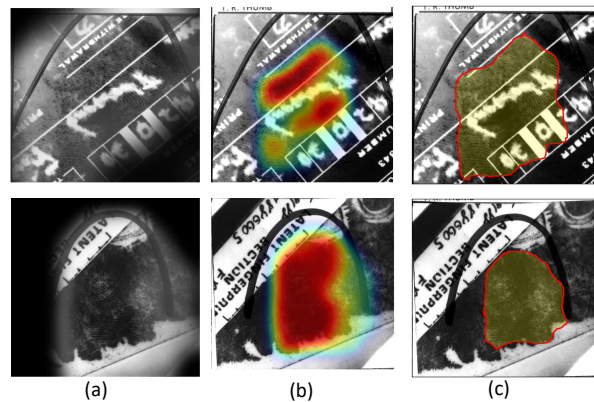
Figure 1. SegFinNet with visual attention mechanism for two different input latents, one per row: (a) Focused region from Visual attention module (section 3.2); (b) Original latents overlaid with a heat map showing the probability of occurrence of friction ridges (from high to low); (c) Binary mask (boundary marked in red) used in subsequent modules: enhancement, feature extraction, and matching.

terns (foreground) from the background; and (c) *Instance Segmentation* [8] as separating individual friction ridge patterns in the input latent by semantic segmentation.

Object segmentation can be based on two different approaches: (i) fully convolutional neural networks (FCN) based [14] and (ii) object detection based [8]. FCN based segmentation consists of a series of consecutive receptive fields in its network and is built on translation invariance. Instead of computing general nonlinear functions, FCN builds its nonlinear filters based on relative spatial information in a sequence of layers. On the other hand, detection based segmentation first finds a core and then branches out in parallel to construct pixel-wise segmentation from regions of interest returned by previous detection.

Our proposed method, called SegFinNet, inherits the idea of instance segmentation and utilizes the advantages of FCN [14] and Mask-RCNN [8] to deal with the latent fingerprint cropping problem. SegFinNet uses Faster RCNN [17] as its backbone while its head comprises of atrous transposed convolution layers [4]. We utilize a com-

Study	Method	Database	Results	Comments
Choi *et al.* [5]	Patch orientation and ridge frequency	NIST SD27 and WVU; Background: 32K images	NIST SD27: 14.78% MDR; 47.99% FDR [+] WVU: 40.88% MDR; 5.63% FDR *Matching:* 16.28% on NIST SD27 and 35.1% on WVU with COTS tenprint matcher [*]	Relies on input image quality and orientation estimation
Zhang *et al.* [19]	Adaptive directional total variance model	NIST SD27 (1,000 dpi); Background: 27K images	14.10% MDR; 26.13% FDR; *Matching:* 2% on NIST SD27 with Verifinger SDK 6.6	Relies on orientation field and orientation coherence estimation
Ruangsakul *et al.* [18]	Fourier Subbands using spatial-frequency information	NIST SD27; Background: 27K images	31.90% MDR; 32.50% FDR; *Matching:* 14% on NIST SD27 with Verifinger SDK 6.6	Handcrafted subband features; dilation and erosion used to fill gaps and eliminate islands
Cao *et al.* [3]	Patch classification based on learned dictionary	NIST SD27 and WVU; Background: 32K images	*Matching:* 61.24% on NIST SD27 and 70.16% on WVU with a COTS matcher [*]	Heuristic patch classification; relies on learned dictionary quality and convex hull to get smooth mask
Liu *et al.* [13]	Linear density on a set of line segments from the texture component of latent images	NIST SD27; Background: 27K images	13.32% MDR; 24.21% FDR; *Matching:* 22% on NIST SD27 with Verifinger SDK 4.3	Use dilation and erosion for post-processing and use convex hull to get smooth mask
Zhu *et al.* [20]	Neural network as binary patch based classifier	NIST SD27; No background reported	10.94% MDR; 11.68% FDR; No matching accuracy reported	Relies on neural network classifier; patch by patch processing is time consuming
Ezeobiejesi *et al.* [6]	Patch-based stack of restricted Boltzmann machines	NIST SD27, WVU, and IIITD; No background reported	NIST SD27: 1.25% MDR; 0.04% FDR [#] WVU: 1.64% MDR; 0.60% FDR; IIITD: 1.35% MDR; 0.54% FDR; No matching accuracy reported	Depends on the stability of classifier; time consuming
Proposed approach	Automatic segmentation based on FCN and detection based fusion	NIST SD27, WVU, and a forensic database; Background: 100K images	MDR, FDR, and IoU metrics; *Matching:* 70.8% on NIST SD27 and 71.3% on WVU with a COTS matcher; *Matching:* 12.6% on NIST SD27 and 28.9% on WVU with Verifinger SDK 6.3 on 27K images	Non-patch based approach; non-warp region of interest; visual attention mechanism; voting masks technique

[+] MDR: Missed Detection Rate; FDR: False Detection Rate; IoU: Intersection Over Union
[*] COTS: Commercial off the shelf; The authors did not identify which COTS was used.
[#] This work used a subset of dataset for training and their metrics are defined on patches.

Table 1. Published works related to latent fingerprint segmentation.

bination of a non-warp region of interest technique, a fingerprint attention mechanism, a fusion voting and a feedback scheme to take advantage of both the deep information from neural networks and the shallow appearance of fingerprint domain knowledge (Figure 1). In our experiments, SegFinNet shows a significant improvement not only in latent cropping, but also in latent search (see Section 4.6 for more details).

2. Related work

In latent fingerprint recognition literature, it is a common practice to use a patch based approach in various modules, (i.e. minutiae extraction [16] and enhancement [19, 3]). In such cases, the input latent is divided into multiple overlapping patches at different locations. The latent segmentation module of latent recognition systems has also been approached in this patch based manner, both with convolutional neural networks (convnet) and non-convnet approaches. Table 1 concisely describes these methods reported in the literature.

Non-convnet patch-based approaches: Choi *et al.* [5] constructed orientation and frequency maps to use as a ref-

erence in evaluating latent patches. Essentially, this is a dictionary look up map aimed at classifying each individual patch into two classes. Zhang *et al.* [19] used an adaptive directional total variance model which also relies on the orientation estimation. From the information in the spatial-frequency domain, Ruangsakul *et al.* [18] proposed a Fourier subband method with necessary post-processing to fill gaps and eliminate islands. Cao *et al.* [3] classified patches based on a dictionary which depends on dictionary quality and needs post-processing to make the masks smooth. Liu *et al.* [13] utilized texture information to develop linear density on a set of line segments but requires a post-processing technique.

The features used in all the above methods are "handcrafted" and rely on post-processing techniques. With the success of deep neural networks in many domains, latent fingerprint cropping has also been tackled using them.

Convnet patch-based approaches: Zhu *et al.* [20] used a classification neural network framework to classify patches. This approach is similar to existing non-convnet methods except it simply replaces hand-crafted features by convnet features. Ezeobiejesi *et al.* [6] used a stack of re-
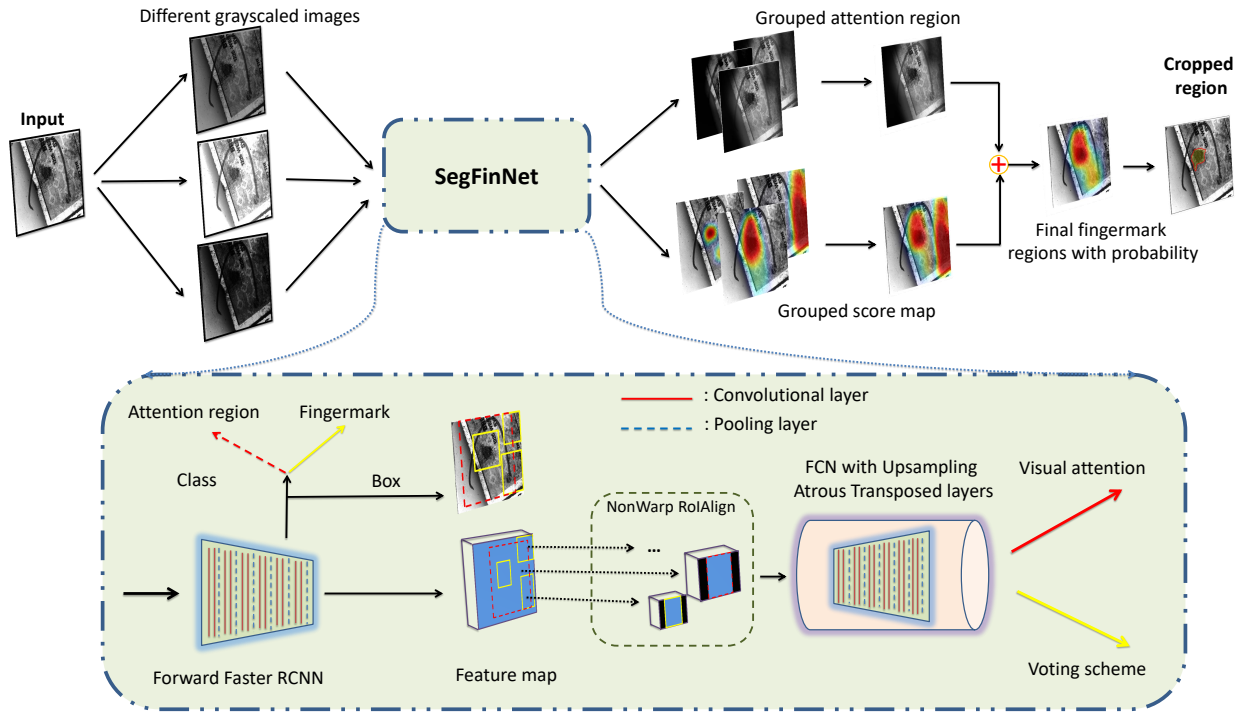
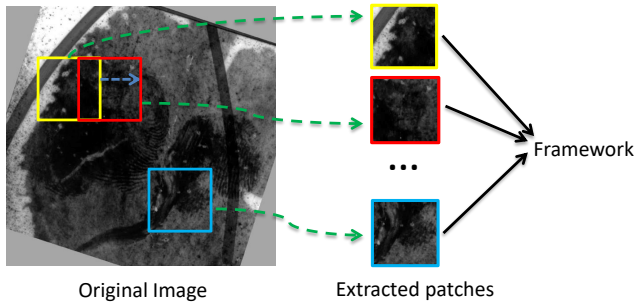Figure 2. SegFinNet architecture.

Figure 3. General pipeline of patch-based approaches.

stricted Boltzmann machines in an idea similar to [20].

There are a number of disadvantages to patch-based approaches. (i) Patch-based methods take significant time to compute since they need to process every patch into the framework (Figure 3). Typically, a patch size is around 96×96. Thus, on average there are, approximately 500 patches in a latent fingerprint in the NIST SD27 dataset. That means patch-based approaches process 500 subimages instead of one. (ii) Patch-based approaches cannot separate multiple instances of friction ridge patterns, i.e. more than one latent (overlapping or non-overlapping) in the input image.

Our work combines fully convolutional neural network and detection based approaches for latent fingerprint segmentation to process the entire input latent image in one shot. Furthermore, it also utilizes a top-down approach (detection before segmentation) which can also be applied to segmenting overlapping latent fingerprints. The main contributions of this paper are as follows:

- A fully automatic latent segmentation framework, called *SegFinNet*, which processes the entire input image in one shot. It also outputs multiple instances of fingermark locations.

- *NonWarp-RoIAlign* is proposed to obtain more precise segmentation while mapping region of interest (cropped region) in the feature map to the original image.

- *Visual attention technique* is designed to focus only on fingermark regions in the input image. This addresses the problem of *"where to look"*.

- *Feedback scheme* with weighted loss is utilized to emphasize the difference in importance of different objective functions (foreground-background, bounding box, etc.)

- *Majority voting fusion mask* is proposed to increase the stability of the cropped mask while dealing with different qualities of latents.

- Experiments demonstrating that the proposed framework outperforms both human latent cropping and published automatic cropping approaches. Furthermore, the proposed segmentation framework, when integrated with a latent AFIS, boosts the search accuracy on three different latent databases: NIST SD27, WVU, and MSP DB (an operational forensic database).

Algorithm 1 SegFinNet latent fingerprint cropping

  Input: Latent fingerprint image
  Output: Binary mask
1: Generate different types of grayscale images.
2: **procedure** PROCESS EACH GRAYSCALE IMAGE
3:     Feed the input image to Faster RCNN to obtain the feature map together with the bounding boxes (coordinates) of fingermarks and attention region candidates.
4:     **for each** box in the candidate list **do**
5:       Regard each box as a friction ridge image to feed to FCN to obtain Visual attention region (section 3.2) and Voting scheme (section 3.4) results.
6:     **end for**
7: **end procedure**
8: Fuse results to get the final fingermark probabilities.
9: Apply a hard-threshold to get binary mask for input latent image.
---

3. SegFinNet

Based on the idea of detection-based segmentation of Mask RCNN [8], we build our framework upon the Faster RCNN architecture [17], where the head is a series of atrous transposed convolutions for pixel-wise prediction.

Unlike previous patch-based approaches which used either handcrafted features [18, 3, 13] or a convnet approach [20, 6], we feed the whole input latent image once to Faster RCNN and process candidates foreground regions returned by SegFinNet. This reduces the training time, and avoids post-processing heuristics to combine results from different patches. Figure 2 and Table 1 illustrate the SegFinNet architecture in details.

3.1. NonWarp-RoIAlign

The RoIAlign module in Mask RCNN can handle the misalignment problem[1] while quantizing the region of interest (RoI) coordinates in feature maps by using bilinear interpolation on fixed point values [10]. However, it warps the RoI feature maps into a square size (e.g. 96×96) before feeding to the upsampling step. This leads to further misalignment and information loss when reshaping the ROI feature map back to original size in image coordinates.

The idea of NonWarp-RoIAlign is simple but effective. Instead of warping RoI feature maps into squared size and then applying multiple deconvolution (upsampling) layers, we only pad zero value pixels to get to a specific size. This can avoid the loss of pixel-wise information when warping regions. We use atrous convolution [4] when upsampling for faster processing and saving memory resources (see Figure 2 for more visualization). The advantage of this

method of upsampling is that we can deal with the multi-scale problem with atrous spatial pyramid pooling properties and weights of atrous convolution can be obtained from the transposed corresponding forward layer.

We also adopt the strategy of combining high-level layers with low-level layers [9, 4, 16] to get finer detail prediction while maintaining high-level semantic interpretation as multi-scale prediction.

3.2. Where to look? Visual attention mechanism

Latent examiners tend to examine fingermarks, directed by the RoI, to identify the region of interest (see Figure 4). Thus, by directing attention to a specific fingerprint, we can eliminate unnecessary computation for low interest regions.

We reuse feature maps returned by Faster RCNN to locate the region of interest. Next, we train SegFinNet to learn two classes: (i) attention region (fingermark region identified by a black marker by the examiner (Figure 2) and (ii) fingermark. In the inference phase, a comparison between returned the fingermarks' location to the attention region is used to decide which one needs to be kept using the following criterion: if the overlapping area between the fingerprint bounding box and the attention region is over 70%, the bounding box is kept.

Our attention mechanism is intuitive, and it helps during matching (see Section 4.6 for more details) because it eliminates background friction ridges which generate spurious minutiae.

3.3. Feedback scheme

One issue in using a detection-based segmentation approach is that it segments objects based on candidates (RoI returned by detector). Thus, a large proportion of pixels in these bounding boxes belong to the foreground (fingermarks) rather than the background. The need to have a new loss function that can handle the imbalanced class problem is necessary. Let $D = \{(x_i, y_i)\}_{i=1,..,N}$ be a set of N
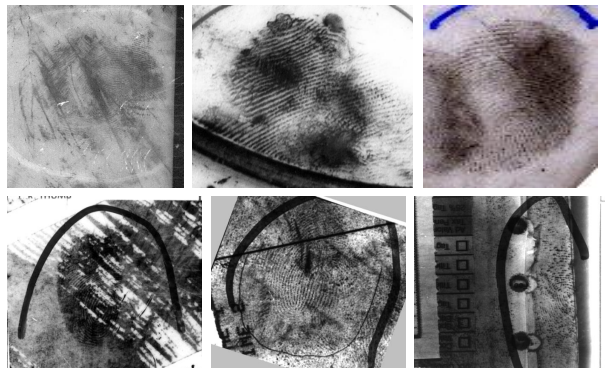

Figure 4. Example images from MSP database (top row) and NIST SD27 (bottom row) with RoI markup by a latent examiner (by colored marker).

---
[1]Due to mapping each point in feature map to the nearest value in its neighboring coordinate grid. We refer readers to [8] for more details.
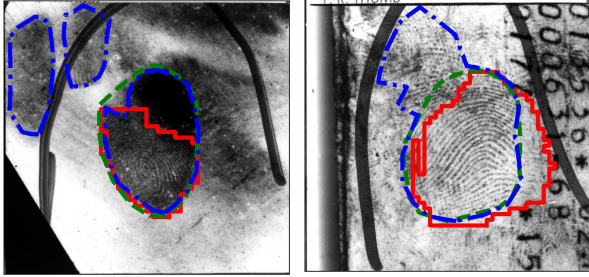
Figure 5. Different groundtruths for two latents in NIST SD27. The groundtruth croppings shown in red, green, and blue are used in Cao *et al.* [3], Ruangsakul *et al.* [18], and Zhu *et al.* [20], respectively.

training samples, where x_i is the input image and y_i is its corresponding groundtruth mask. SegFinNet outputs a set of concatenated C masks $\langle (x_i, \gamma_{i1}), (x_i, \gamma_{i2}), ..., (x_i, \gamma_{iC}) \rangle$ for each input x_i. We create weights (w_j, $j = 1, .., C$) for each loss value to solve this problem.

Unlike most popular computer vision datasets that have a significant number of pixel-wise annotated masks, there is no dataset available in the fingerprint domain that provides pixel-wise segmentation. Hence, different published studies have used different annotations (see Figure 5). Furthermore, since the border of fingermarks is usually not well defined, it leads to inaccuracies in these manual masks and, subsequently, training error. To alleviate this concern, we propose a semi-supervised partial loss that updates the loss for pixels in the same class while discarding other classes except the background.

Combining the two solutions together, let $\mathcal{L}_{\mathcal{M}}$ be the segmentation (mask) loss which takes into account the proportion of all classes in the dataset:

$$\mathcal{L}_{\mathcal{M}} = \sum_{j \in C} (\sum_{i \in N} w_j \mathscr{L}(\gamma_{ij}, y_{ij}) + \lambda l(\gamma_{ij}, y_{ij})) \quad (1)$$

where w_j is the soft-max weight number of pixels on the jth label, y_{ij} is corresponding mask label of the ith sample in the jth class, λ is regularization term, $l(.)$ is cross-entropy loss w.r.t. background, and $\mathscr{L}(.)$ is the per-pixel sigmoid average binary cross-entropy loss as defined in [8].

In the training phase, we consider the loss function as a weight sum of the class, bounding box, and mask loss. Let $\mathcal{L}_{all}, \mathcal{L}_C, \mathcal{L}_B, \mathcal{L}_M$ be the total, class, bounding box, and pixel-wise mask loss, respectively. The total loss for training is calculated as follows:

$$\mathcal{L}_{all} = \alpha \mathcal{L}_{\mathcal{C}} + \beta \mathcal{L}_{\mathcal{B}} + \gamma \mathcal{L}_{\mathcal{M}} \quad (2)$$

We set $\alpha = 2$, $\beta = 1$, $\gamma = 2$ to emphasize the importance of the correctness of predicted class and pixel-wise instance segmentation. We note that the mask loss, $\mathcal{L}_{\mathcal{M}}$ is based on the Intersection over Union (IoU) criteria [14, 8, 4].

3.4. Voting fusion masks

The effect of grayscale normalization. In the computer vision domain, input images usually have intensity values in a specific range. Thus, we can easily recognize, detect or segment objects. However, the fingerprint domain differs from most traditional computer vision related problems. In particular, because of the noisy background and low contrast of the fingerprint ridge structure, mistakes can be easily made in detecting fingermarks. Motivated by the various procedures used by forensic experts to "preprocess" images when examining latent fingerprints, we tried different preprocessing techniques on the original latent such as centered gray scaled, histogram equalized, and inverse image.

Even though the original latent fingerprint is noisy, removing noise via an enhancement algorithm prior to segmentation [3, 19] is not advisable because texture information may be lost. To make sure the segmentation result is reasonable and invariant to the contrast of input image, we propose a simple but effective voting fusion mask technique. Given an input latent, we first preprocess it to generate different types of grayscale images which are subsequently fed into SegFinNet to get the corresponding score maps. The final score map is accumulated over different grayscale inputs. Each pixel in the image has its own score value. We set the threshold to $K = 3$, which means that each pixel in the chosen region receives at least 3 votes from the voting masks.

Although this approach seems to increase the computational requirement, it boosts the reliability of the resulting mask while keeping the running time of whole system efficient (see Section 4.5 for quantitative running time values).

4. Experiments

4.1. Implementation Details

We set the anchor size in Faster RCNN varying from 8×8 to 128×128. Batch size, detection threshold, and pixel-wise mask threshold are set to 32, 0.7, and 0.5, respectively. The learning rate for SegFinNet is set to 0.001 in the first $30k$ iterations, and 0.0001 in the rest of the $70k$ iterations. Mini-batch size is 1 and weight decay is 0.0001. We set the hyper-parameter $\lambda = 0.8$ in Equation 1 .

4.2. Datasets

We have used 3 different latent fingerprint databases: MSP DB (an operational forensic database), NIST SD27 [7], and West Virginia University latent database (WVU) [1]. The MSP DB includes 2K latent images and over 100K reference rolled fingerprints. The NIST SD27 contains 258 latent images with their true mates while WVU contains 449 latent images with their mated rolled fingerprints and another 4, 290 non-mated rolled images.
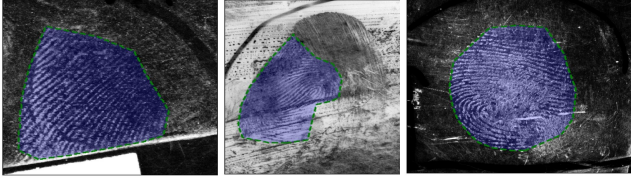
Figure 6. Example images in the MSP database with the corresponding manual ground truth mask overlaid.

Training: we used the MSP DB to train SegFinNet. We manually generated ground truth binary masks for each latent. Figure 6 shows some example latents in the MSU DB with the corresponding groun truth masks. We used a subset of 1000 latent images from MSP DB for training while using a different set of 1000 latents in MSP DB for testing. With common augmentation techniques (e.g. random rotation, translation, scaling, cropping, etc.), the training dataset size increased to $8K$ latent images.

Testing: we conducted experiments on NIST SD27, WVU, and 1000 sequestered test images from the MSP database. To make the latent search to appear more realistic, we constructed a gallery consisting of $100K$ rolled images, including the 258 true mates of NIST SD27, $4,290$ rolled fingerprints in WVU, $27K$ images from NIST14 [2], and the rest from rolled prints in the MSP database. The ground truth masks were obtained from [11].

4.3. Cropping Evaluation Criteria

Published papers based on patch-based approach with a classification scheme reported the cropping performance in terms of MDR and FDR metrics. The lower values of these metrics, the better framework is. Let A and B be two sets of pixels in predicted mask and ground truth mask, respectively. MDR and FDR are then defined as:

$$MDR = \frac{|B| - |A \cap B|}{|B|}; FDR = \frac{|A| - |A \cap B|}{|A|} \quad (3)$$

With the proposed non-patch-based and top-down approach (detection and segmentation), it is necessary to use IoU metric, which is more appropriate for multi-class segmentation [14, 8, 4]. In addition, we report our results in terms of the MDR and FDR metrics for comparison. In contrast to MDR and FDR metrics, a superior framework will lead to a higher value of IoU. The IoU metric is defined as:

$$IoU = \frac{|A \cap B|}{|A \cup B|} \quad (4)$$

We note that the published methods [18, 20, 5] used their own individual ground truth information so comparing them based on MDR and FDR is not fair given these two metrics critically depend on the ground truth. Figure 5 demonstrates the variations in ground truths used by existing works[2]. It is

---

[2]We contacted the authors to get their groundtruths.

important to emphasize that a favorable metric value does not mean that the associated cropping will lead to better latent recognition accuracy. It simply reveals the overlap between predicted mask and its manually annotated ground truth.

4.4. Cropping Accuracy

Table 2 shows a quantitative comparison between SegFinNet and other existing works using MDR and FDR metrics on NIST SD27 and WVU databases. The IoU metric was computed based on masks and the ground truth provided by the authors. Because Ezeobiejesi *et al.* evaluated MDR and FDR on patches, it is not fair to include it for IoU comparison. Table 2 also shows that SegFinNet provides the lowest error rate in terms of MDR and FDR. This is because of our use of non-patch based approach. The table also reveals that the low values of either MDR or FDR only does not usually lead to high IoU value.

4.5. Running time

Experiments are run on a Nvidia GTX Ti 1080 GPU with 12GB of memory. Table 3 shows a comparison with different configurations in computation time on NIST SD27 and WVU. We note that the voting fusion technique takes longer time to process an image because it runs on different inputs. However, its accuracy is better than just using a single image with attention technique.

Figure 7 shows the visualization of SegFinNet compared to existing works on NIST SD27. These masks were obtained by contacting authors.

4.6. Latent Matching

The final goal of segmentation is to increase the latent matching accuracy. We used two different matchers for latent to rolled matching: Verifinger SDK 6.3 [15] and a state-of-the-art latent COTS AFIS. To make a fair comparison to

Table 2. Comparison of the proposed segmentation method with published algorithms using pixel-wise (MDR, FDR, IoU) metrics on NIST SD27 and WVU latent databases.

Dataset	Algorithm	MDR	FDR	IoU
NIST SD27	Choi [5][#]	14.78%	47.99%	43.28%
	Zhang [19]	14.10%	26.13%	*N/A*
	Ruangsakul [18][#]	24.56%	36.48%	52.05%
	Cao [3][#]	12.37%	46.66%	48.25%
	Liu [13]	13.32%	24.21%	*N/A*
	Zhu [20]	10.94%	11.68%	*N/A*
	Ezeobiejesi [6][*]	1.25%	0.04%	*N/A*
	Proposed method	**2.57%**	**16.36%**	**81.76%**
WVU	Choi [5]	40.88%	5.63%	*N/A*
	Ezeobiejesi [6][*]	1.64%	0.60%	*N/A*
	Proposed method	**13.15%**	**5.30%**	**72.95%**

[#] We reproduce the results based on masks and ground truth provided by authors.
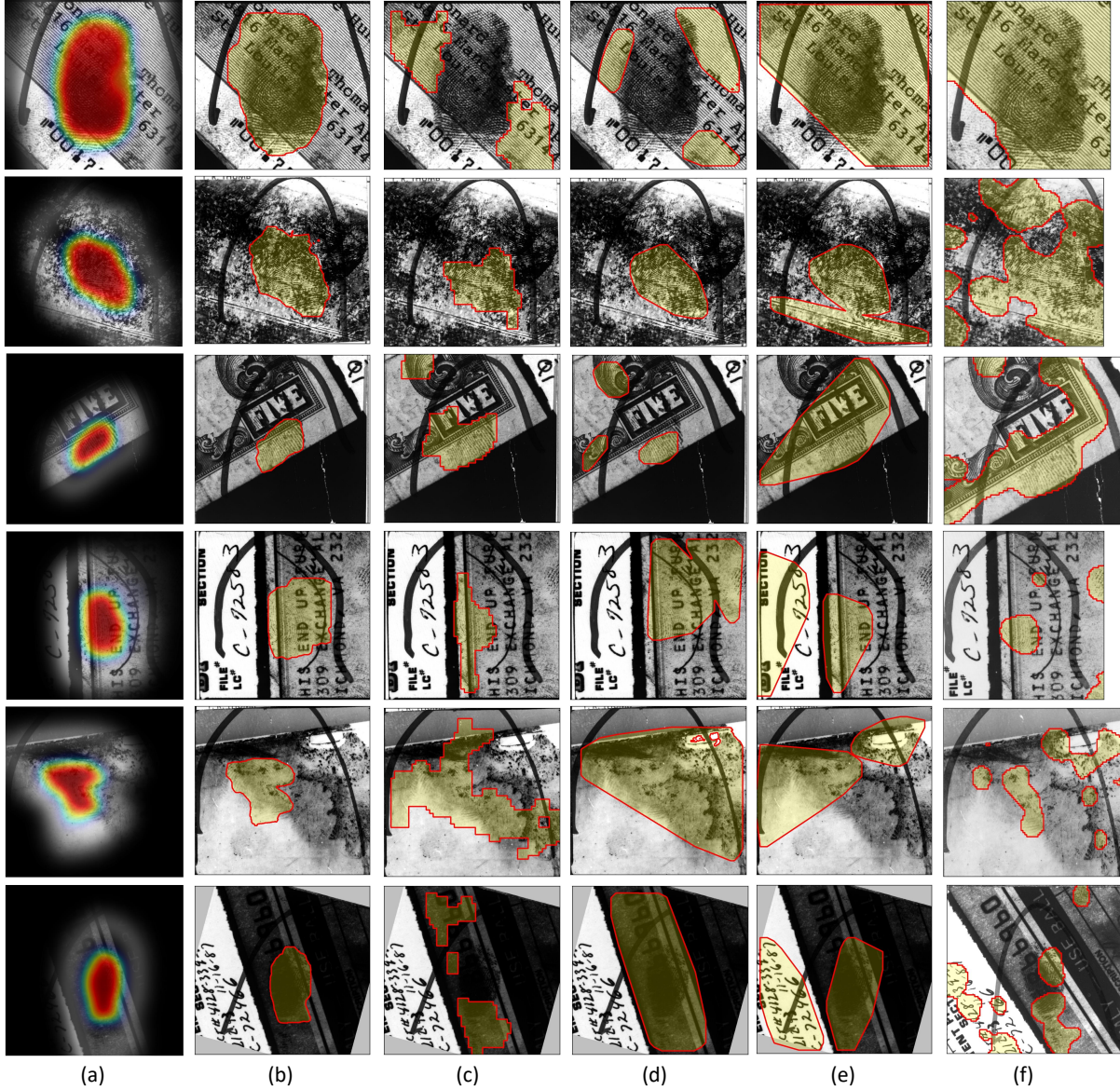[*] Its metrics are on reported patches.

Figure 7. Visualizing segmentation results on six different (one per row) latents from NIST SD27. (a) Our visual attention with heatmap (fingermark probability), (b) Proposed method, (c) Ruangsakul *et al*. [18], (d) Choi *et al*. [5], (e) Cao *et al*. [3], (f) Zhang *et al*. [19]. Images used for comparison vary in terms of noise, friction ridge area and ridge clarity. Note that Zhang *et al*. used 1,000 dpi images while others, including us, used 500 dpi latents.

existing works [19, 18, 13], we also report the matching performance for Verifinger on 27K background from NIST 14 [2]. In addition, we report the performance of COTS on 100K background. To explain the matching experiments, we first define some terminologies.

(a) *Baseline:* Original gray scale latent image.

(b) *Manual GT:* Groundtruth masks from Jain *et al*. [3].

(c) *SegFinNet with AM:* Masked latent images using visual attention mechanism only.

(d) *SegFinNet with VF:* Masked latent images using majority voting mask technique only.

(e) *SegFinNet full:* Masked latents with full modules.

(f) *Score fusion:* Sum of score fusion of our proposed SegFinNet with SegFinNet+AM, SegFinNet+VF, and original input latent images.

Table 4 demonstrates matching results using Verifinger. Since there are many versions of the Verifinger SDK, we use masks provided by the authors in [5, 19, 3, 18] to make a fair comparison. However, the authors did not provide their masks for the WVU database. Note that contrary to popular belief, the manual groundtruth does not always give better results than original images.
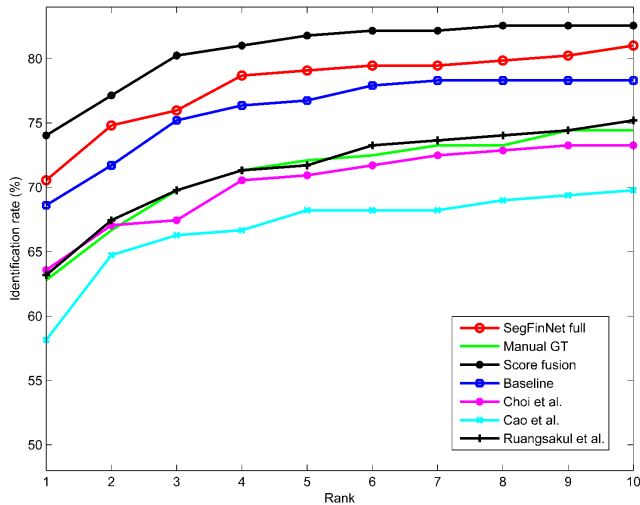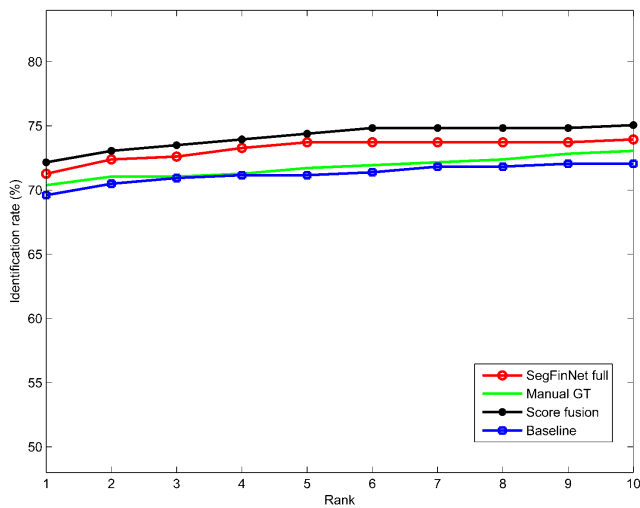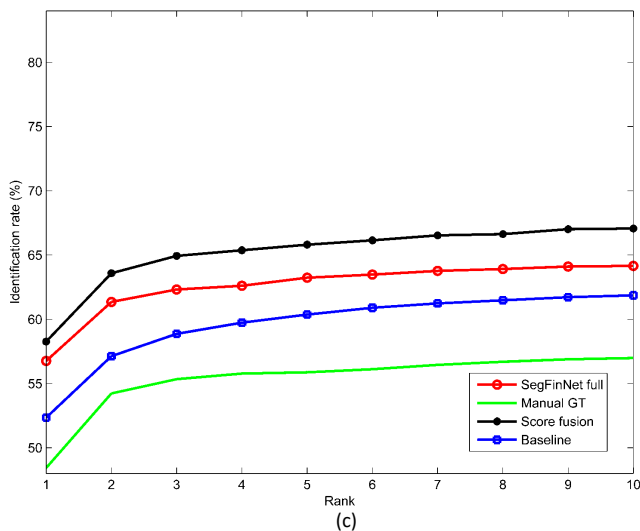
Figure 8. Matching results with a state-of-the-art COTS matcher on (a) NIST SD27, (b) WVU, and (c) MSP database against 100K background images.

Table 3. Performance of SegFinNet with different configurations. AM: attention mechanism (Section 3.2), VF: voting fusion scheme (Section 3.4)

Dataset	Configuration	Time(ms)	IoU
NIST SD27	SegFinNet w/o AM & VF	248	46.83%
	SegFinNet with AM	274	50.60%
	SegFinNet with VF	396	78.72%
	SegFinNet full	**457**	**81.76%**
WVU	SegFinNet w/o AM & VF	198	51.18%
	SegFinNet with AM	212	62.07%
	SegFinNet with VF	288	67.33%
	SegFinNet full	**361**	**72.95%**

Table 4. Matching results with Verifinger on NIST SD27 and WVU against 27K background.

Dataset	Methods	Rank-1	Rank-5
NIST SD27	Choi [5]	11.24%	12.79%
	Ruangsakul [18]	11.24%	11.24%
	Cao [3]	11.63%	12.01%
	Manual GT	10.85%	11.63%
	Baseline	8.14%	8.52%
	Proposed method	**12.40%**	**13.56%**
	Score fusion	13.95%	16.28%
WVU	Manual GT	25.39%	26.28%
	Baseline	26.28%	27.61%
	Proposed method	**28.95%**	**30.07%**
	Score fusion	29.39%	30.51%

Figure 8 shows the matching results using state-of-the-art COTS. We did not use any enhancement techniques like Cao *et al.* [3] in this comparison. The combination between the attention mechanism and the voting technique showed better performance in our proposed method. Besides, highest results of score fusion technique mean that our method can be complementary to using full image in matching.

5. Conclusion

We have proposed a framework for latent segmentation, called SegFinNet. It utilizes a fully convolutional neural network and detection based approach for latent fingerprint segmentation to process the full input image instead of dividing it into patches. Experimental results on three different latent fingerprint databases (i.e. NIST SD27, WVU, and MSP database) show that SegFinNet outperforms both human ground truth cropping for latents and published segmentation algorithms. This improved cropping, in turn, boosts the hit rate of a state of the art COTS latent fingerprint matcher. Our framework can be further developed along the following lines: (a) Integrating into an end-to-end matching model by using shared parameters learned in the Faster RCNN backbone as a feature map for minutiae/non-minutiae extraction; (b) Combining orientation information to get instance segmentation for segmenting overlapping latent fingerprints.

Acknowledgements

References

[1] Integrated pattern recognition and biometrics lab, West Virginia University. http://www.csee.wvu.edu/~ross/i-probe/. 5

[2] NIST Special Database 14. http://www.nist.gov/srd/nistsd14.cfm. 6, 7

[3] K. Cao, E. Liu, and A. K. Jain. Segmentation and enhancement of latent fingerprints: A coarse to fine ridgestructure dictionary. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(9):1847–1859, 2014. 2, 4, 5, 6, 7, 8

[4] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 1, 4, 5, 6

[5] H. Choi, M. Boaventura, I. A. Boaventura, and A. K. Jain. Automatic segmentation of latent fingerprints. In *IEEE International Conference on Biometrics: Theory, Applications and Systems*, pages 303–310, 2012. 2, 6, 7, 8

[6] J. Ezeobiejesi and B. Bhanu. Latent fingerprint image segmentation using deep neural network. In *Deep Learning for Biometrics*, pages 83–107. Springer, 2017. 2, 4, 6

[7] M. D. Garris and R. M. McCabe. NIST special database 27: Fingerprint minutiae from latent and matching tenprint images. *NIST Technical Report NISTIR*, 6534, 2000. 5

[8] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *IEEE International Conference on Computer Vision*, pages 2980–2988, 2017. 1, 4, 5, 6

[9] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 4

[10] M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, pages 2017–2025, 2015. 4

[11] A. K. Jain and J. Feng. Latent fingerprint matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):88–100, 2011. 6

[12] A. K. Jain, K. Nandakumar, and A. Ross. 50 years of biometric research: Accomplishments, challenges, and opportunities. *Pattern Recognition Letters*, 79:80–105, 2016. 1

[13] S. Liu, M. Liu, and Z. Yang. Latent fingerprint segmentation based on linear density. In *IEEE International Conference on Biometrics*, pages 1–6, 2016. 2, 4, 6, 7

[14] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015. 1, 5, 6

[15] Neurotechnology Inc. Verifinger. http://www.neurotechnology.com/verifinger.html. 6

[16] D.-L. Nguyen, K. Cao, and A. K. Jain. Robust minutiae extractor: Integrating deep networks and fingerprint domain knowledge. In *IEEE International Conference on Biometrics*, 2018. 2, 4

[17] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, 2017. 1, 4

[18] P. Ruangsakul, V. Areekul, K. Phromsuthirak, and A. Rungchokanun. Latent fingerprints segmentation based on rearranged fourier subbands. In *IEEE International Conference on Biometrics*, pages 371–378, 2015. 2, 4, 5, 6, 7, 8

[19] J. Zhang, R. Lai, and C.-C. J. Kuo. Adaptive directional total-variation model for latent fingerprint segmentation. *IEEE Transactions on Information Forensics and Security*, 8(8):1261–1273, 2013. 2, 5, 6, 7

[20] Y. Zhu, X. Yin, X. Jia, and J. Hu. Latent fingerprint segmentation based on convolutional neural networks. In *IEEE Workshop on Information Forensics and Security*, pages 1–6, 2017. 2, 3, 4, 5, 6