Enhanced Segmentation-CNN based Finger-Vein Recognition by Joint Training with Automatically Generated and Manual Labels

Ehsaneddin Jalilian and Andreas Uhl

Department of Computer Sciences
University of Salzburg
Austria

ISBA 2019

Outlines

Introduction and problem statement

- Finger-vein recognition is a process in which a person's finger vein patterns are captured using near-infrared (NIR) illumination and used as a basis for biometric recognition.
- Poorly designed scanner devices, poor NIR lighting, varying thickness of fingers, etc., cause the images to contain ambiguous (low contrast) regions between vein and non-vein areas.
- The intensity distributions in these areas can hardly be described by a mathematical model, making it difficult to propose a comprehensive algorithmic solution to extract the actual vein pasterns.
- Deep learning techniques (e.g CNNs), are gaining increasing interest within the biometric community, and application of CNNs for vein extraction has received some research attentions recently.
- Large amounts of high-quality annotated samples (ground-truths) are typically required for CNN training, which is very expensive and time consuming.
- This process gets even more tedious and error-prone, in case the annotators have to deal with ambiguous images (finger-vein images).

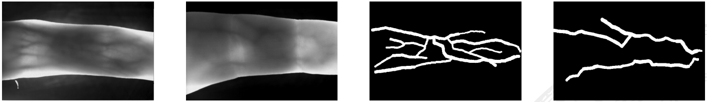
Introduction and problem statement

- We propose a novel approach for finger-vein recognition, focused on direct extraction of actual finger-vein patterns from NIR finger images using semantic segmentation networks.
- We also tray to address problem of the ground-truth labels generation, analyzing the effect of training data quantity on the network performance.
- Also, we propose an efficient training model based on automatically generated labels, aiming to eliminate the need for the required grand-truth labels, and also improve the networks' performance.
  - ✓ We derive a set of features rather than direct use of NIR finger image itself.
  - ✓ Proposing automatic label generation technique, we utilize the advances made in any of the three major steps of traditional biometric systems (pre-processing, feature extraction, alignment and matching).
  - ✓ Finally, our approach allows to employ the generated binary labels in existing multiple features fusion, or multi-sample recognition techniques.

Experimental framework

Databases

- We used the UTFVP database[1] which contains 1440 finger-vein images (672x380) from 60 volunteers, with 2 hands, 3 fingers (index, middle, and ring fingers), 4 images per finger.
- We established an annotation tool to generate the Manual labels for a subset (400 samples) of the dataset (including at least one sample per subject).



(a) (b) (c) (d)

Figure: Sample finger-vein images in the UTFVP database (a, b), and their corresponding generated Manual labels (c, d) respectively

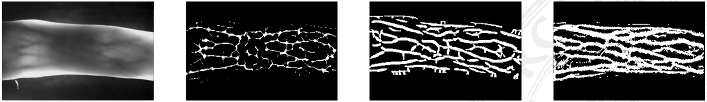
Networks

- CNN networks used: Unet [1], RefineNet [2], SegNet [3].

[1] http://scs.ewi.utwente.nl/downloads.

Automated label generation

- Given the problems with manual data annotation, generating ground-truth labels automatically has been suggested for some CNN-based segmentation tasks in many fields (in special medical imaging).
- Aiming to improve the networks' preference (by improving annotation accuracy and precision in ambiguous areas), and eliminating the need for the Manual labels, we used Maximum Curvature (MC) [4], Repeated Line Tracking (RLT) [5], and Gabor Filter (GF) [6] algorithms to generate the corresponding (to the Manual labels) ground-truth labels.



(a) (b) (c) (d)

Figure: Sample UTFVP image (a) and its corresponding automatically generated labels using MC (b), GF (c), and RLT (d) algorithms respectively

Training

- We divided the dataset into two parts (testing sets), then for he Manual labels, we created two disjoint training sets (containing 180 samples each) within the testing set, then we created 6 disjoint training subsets (containing 180, 140, 100, 60, 20 and 5 labels, respectively) within each training set.
- For the automatically generated labels we performed the same partitioning, while we created 5 disjoint training subsets (containing 200, 160, 120, 80, and 40 labels, respectively) within each training set.
- First, we trained the networks with each subset using Manual labels, and evaluated the them on the oder testing set of the dataset, and vis versa.
- We repeated the same training and evaluation process using the automatically generated labels (keeping only 40 pcs of Manual labels).

Evaluation

- Evaluation: receiver operator characteristic behavior (in particular EER, FMR 1000 (FMR), and ZeroFMR (ZFMR)), using FVC2004 protocol [7].

Matching

- For matching, we calculated the correlation between the binary feature maps.
- The correlation between the input $I(x,y)$ and the reference one is calculated several times while shifting the reference $R(x,y)$, whose upper-left position is $R(c_w,c_h)$ and lower-right position is $R(w-c_w,h-c_h)$, in x- and y-direction.

$$N_m(s,t)=\sum_{y=0}^{h-2c_h-1}\sum_{x=0}^{w-2c_w-1} I(s+x,t+y)R(c_w+x,c_h+y) \tag{1}$$

where $N_m(s,t)$ is the correlation. The maximum value of the correlation is normalized and used as matching score:

$$S=\frac{N_{m_{max}}}{\sum_{y=t_0}^{t_0+h-2c_h-1}\sum_{x=s_0}^{s_0+w-2c_w-1} I(x,y)+\sum_{y=c_h}^{h-2c_h-1}\sum_{x=c_w}^{w-2c_w-1} R(x,y)} \tag{2}$$

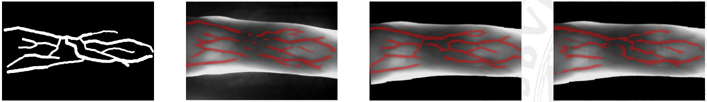
where s_0 and t_o are the indexes of $N_{m_{max}}$ in the correlation matrix $N_m(s,t)$, and S values are: $0 \leq S \leq 0.5$.

First experiential stage using Manual labels

- We analyzed the networks' performance using different quantities of the Manual labels.

Networks	Unet			RefineNet			SegNet		
Labels	EER	FMR	ZFMR	EER	FMR	ZFMR	EER	FMR	ZFMR
180 pcs	0.87	1.85	5.18	2.73	5.83	11.85	2.91	6.75	12.63
140 pcs	1.15	2.08	4.30	2.73	6.62	9.02	3.09	8.79	16.94
100 pcs	1.04	1.88	3.47	3.09	8.61	18.24	**2.21**	6.20	17.03
60 pcs	1.71	3.65	11.52	2.32	6.01	9.39	2.35	6.66	11.25
20 pcs	**0.64**	1.94	6.34	2.26	5.83	8.19	7.26	25.09	53.70
5 pcs	3.80	11.75	24.30	**1.76**	4.12	6.34	9.71	25.69	31.57

Table: Networks' performance (%), trained with different number of Manual labels



(a) (b) (c) (d)

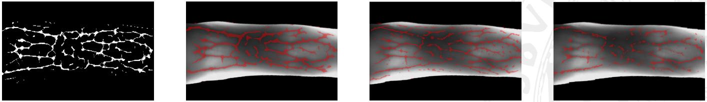
Figure: Sample Manual training label (a) and the corresponding output results using U-net (b), RefineNet (c), and SegNet (d) networks

Second experiential stage using MC labels

- With same objectives in the first experimental stage and also improving the network preference, we curried the same experiments on the networks using labels generated by MC algorithm (along with 40 pcs of Manual labels).

Networks	Unet			RefineNet			SegNet		
Labels	EER	FMR	ZFMR	EER	FMR	ZFMR	EER	FMR	ZFMR
200 pcs	**0.32**	0.60	0.92	**0.28**	0.37	1.57	1.43	2.45	5.64
160 pcs	0.51	1.20	5.18	**0.28**	0.69	1.29	1.34	2.31	3.42
120 pcs	0.41	0.64	1.25	0.36	0.69	1.11	**0.73**	1.66	2.91
80 pcs	0.41	0.55	0.78	0.47	0.97	1.25	1.15	3.47	9.12
40 pcs	1.25	1.38	2.17	1.43	2.50	12.91	4.44	12.96	16.80

Table: Networks' performance (%), trained with different numbers of MC labels



(a) (b) (c) (d)

Figure: Sample MC training label (a) and the corresponding output results using U-net (b), RefineNet (c), and SegNet (d) networks

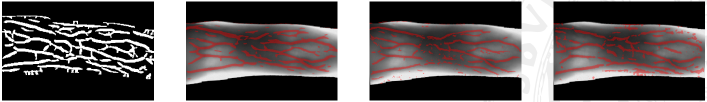
Automatic label training - experiment and results

Second experiential stage using GF labels

- Likewise, we curried the same analytical experiments on the networks using different quantities of labels generated by GF algorithm (along with 40 pcs of Manual labels).

Networks	Unet			RefineNet			SegNet		
Labels	EER	FMR	ZFMR	EER	FMR	ZFMR	EER	FMR	ZFMR
200 pcs	0.79	2.73	3.79	2.13	5.04	8.75	1.20	2.68	5.55
160 pcs	1.02	3.14	12.77	2.77	6.85	10.13	**0.78**	2.59	5.74
120 pcs	3.33	64.30	95.23	3.74	9.35	12.08	1.47	3.37	5.60
80 pcs	**0.74**	2.08	6.71	**1.84**	5.00	8.33	2.21	6.25	12.82
40 pcs	1.61	3.56	6.15	2.36	4.07	7.50	6.57	12.31	17.31

Table: Networks' performance (%), trained with different numbers of GF labels



(a) (b) (c) (d)

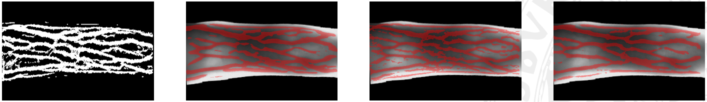
Figure: Sample GF training label (a) and the corresponding output results using U-net (b), RefineNet (c), and SegNet (d) networks

Second experiential stage using RLT labels

- Similarly, we curried the same analytical experiments on the networks using different quantities of labels generated by RLT algorithm (along with 40 pcs of Manual labels).

Networks	Unet			RefineNet			SegNet		
Labels	EER	FMR	ZFMR	EER	FMR	ZFMR	EER	FMR	ZFMR
200 pcs	2.09	11.62	24.86	1.10	2.82	3.75	**1.29**	3.00	7.59
160 pcs	2.45	15.78	23.33	1.61	3.05	5.00	**1.29**	3.14	6.48
120 pcs	**1.16**	5.46	11.20	**0.78**	1.89	3.51	1.43	3.51	5.69
80 pcs	2.36	14.95	35.09	1.38	3.00	4.90	4.87	12.31	19.02
40 pcs	1.57	8.61	17.96	1.80	3.47	6.20	13.41	35.23	43.19

Table: Networks' performance (%), trained with different numbers of RLT labels



(a) (b) (c) (d)

Figure: Sample RLT training label (a) and the corresponding output results using U-net (b), RefineNet (c), and SegNet (d) networks

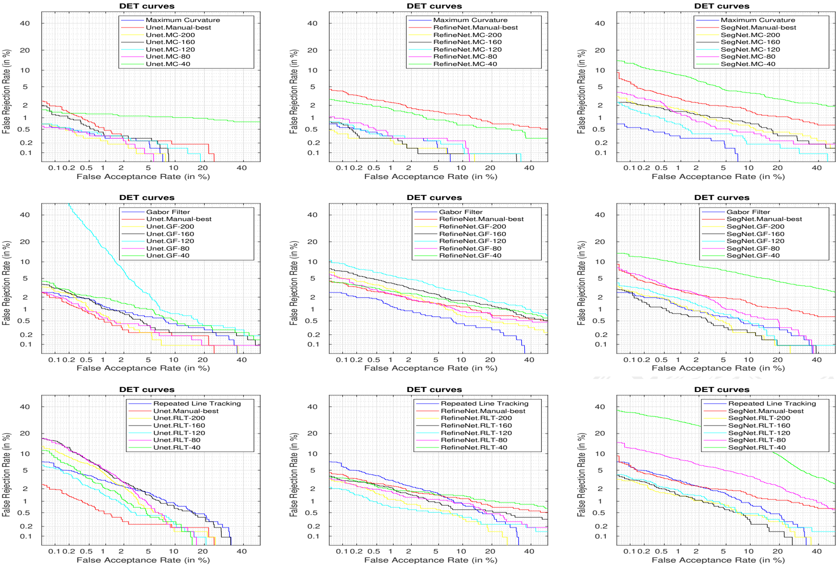


Figure: DET curves for: Unet (left column), RefineNet (middle column), and SegNet(right column) networks, trained whit: MC (first row), GF (second row), and RLT (third row) labels

Evaluating performances

- We compare the obtained recognition performance results to that of the MC, RLT ,GF, and Deformation-tolerant Feature Point (DTFP) [8] algorithms.

Method	MC			GF			RLT			DTFP		
Database	EER	FMR	ZFMR	EER	FMR	ZFMR	EER	FMR	ZFMR	EER	FMR	ZFMR
UTFVP	0.41	0.55	1.29	1.11	2.45	4.12	2.17	5.87	9.35	1.68	2.91	5.18

Table: Classic algorithms' performance (%) on the UTFVP database

- Unet shows better performance than the GF, RLT, and DTFP algorithms (i.e. using 20, 180 labels), while RefineNet outperforms only RLT algorithm (i.e. using 5 labels), and SegNet generally does not perform well on the dataset.
- Using labels generated by MC algorithm:
  - RefineNet clearly outperforms the best classical algorithms result (obtained by MC algorithm), when trained with sufficient (160, 200) pcs of training labels.
  - Unet also outperforms MC algorithm when trained with 200 pcs of automatically generated labels, while generally outperforms GF, RLT, and DTFP algorithms when trained with more than 40 labels.
  - SegNet outperforms GF, RLT, and DTFP algorithms when trained with 120 labels, while increasing the number of training labels of this type (up to 180 pcs) generally erodes its performance.

Evaluating performances

- Using labels generated by GF algorithm:
    - Compared to the results in the second table, only SegNet shows a limited improvement when trained with 160 or more pcs of GF labels.
    - Unet and SegNet outperform the GF, RLT, and DTFP algorithms when trained with distinct number (i.e. 80, 160 receptively) of GF labels, and RefineNet only outperforms the RLT algorithm when trained with 80 pcs of this type of labels.
- Using labels generated by RLT algorithm:
    - Compared to the results in the second table, only SegNet's preference improved slightly when trained with 160 or more pcs of automatically generated labels.
    - Compared to the results in the third table, RefineNet gained up to 50% improvement, but Unet suffered a considerable degradation up to the same order of magnitude in most cases. This is mostly due to the effect of labels quality (accuracy) and the architectural specifications of the networks.

Analyzing the results

- Manual labels are restricted to rather large scale vessels, and the corresponding patterns simply do not contain sufficiently high entropy to facilitate high accuracy recognition. But more accurate labels such as MC labels, and their corresponding outputs of CNNs, exhibit much more fine grained vasculature details.
- RefineNet maintains stable performance, and seems to stay invariant with respect to the quantity of the training labels (using Manual labels), converges well even with a limited number (5) of training labels, and its performance improves introducing higher quantity of automated (more precise) labels.
- U-net converges fast with a limited number of training labels (i.e. 20 labels), and when trained with more precise labels is able to deal well with the ambiguous regions in finger-vein images.
- The SegNet enjoys stable (while not optimal) performance on the dataset, deals better with false positive pixel labels, and seems to be more sensitive to the quantity of the training labels.

Conclusion

- We proposed a new model for finger-vein recognition using fully convolutional deep neural networks, focusing on direct segmentation of actual finger-vein patterns from the finger images, and using them as the binary finger-vein features for the recognition process.
- We showed that automatically generated labels can improve the networks performance in terms of achieved recognition accuracy.
- It also turned out that these improvements are highly dependent on the inter-play between properties of the used labels and the network architecture.
- In any case we demonstrated that, beside improving the network performance, utilizing automatically generated labels to train the networks eliminates the need for the Manual labels, whose generation is an extremely cumbersome, difficult, and error-prone process.

Future work

In future works

- We will assess the strategy to pre-train with Manual labels and then fin-tune the networks with automatically generated ones.
- In addition, we use labels generated by other automated feature extraction techniques in a single training process.
- Also, an evaluation of cross-vessel type (using training data of different vessel types, e.g. retinal vasculature) training will be conducted.
- Finally, we will look into augmentation techniques specifically tailored to the observed problem with the Manual labels, i.e. scaling the data to model also more detailed and finer vessel structures.

Thank you, Remarks?

References

Ronneberger Olaf, Fischer Philipp, and Brox Thomas.
U-net: Convolutional networks for biomedical image segmentation.
In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241, 2015.

Guosheng Lin, Milan Anton, Shen Chunhua, and Ian Reid.
Refinenet: Multi-path refinement networks for high-resolution semantic segmentation.
In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5168–5177, 2017.

Badrinarayanan Vijay, Kendall Alex, and Cipolla Roberto.
Segnet: A deep convolutional encoder-decoder architecture for image segmentation.
IEEE transactions on pattern analysis and machine intelligence, 39(12):2481–2495, 2017.

Miura Naoto, Nagasaka Akio, and Miyatake Takafumi.
Extraction of finger-vein patterns using maximum curvature points in image profiles.
IEICE Transactions on Information and Systems, 90(8):1185–1194, 2007.

Miura Naoto, Nagasaka Akio, and Miyatake Takafumi.
Feature extraction of finger-vein patterns based on repeated line tracking and its application to personal identification.
Machine Vision and Applications, 15(4):194–203, Oct 2004.

Kumar Ajay and Zhou Yingbo.
Human identification using finger images.
IEEE Transactions on Image Processing, 21(4):2228–2244, 2012.

Maio Dario, Maltoni Davide, Cappelli Raffaele, Wayman Jim, and Jain Anil.
Fvc2004: Third fingerprint verification competition.
In *Lecture notes in Biometric Authentication*, pages 1–7. Springer, 2004.

Matsuda Yusuke, Miura Naoto, Nagasaka Akio, Kiyomizu Harumi, and Miyatake Takafumi.
Finger-vein authentication based on deformation-tolerant feature-point matching.
Machine Vision and Applications, 27(2):237–250, Feb 2016.