Pre-Processing Cascades and Fusion in Finger Vein Recognition

Christoph Kauba, Jakob Reissig, and Andreas Uhl

Department of Computer Sciences, University of Salzburg
Jakob-Haringerstraße 2, A-5020 Salzburg, Austria
{ckauba,jreissig,uhl}@cosy.sbg.ac.at

Abstract: Preprocessing and fusion techniques for finger vein recognition are investigated. An experimental study involving a set of preprocessing approaches shows the importance of selecting the appropriate single technique and the usefulness of cascading several different preprocessing methods for subsequent feature extraction of various types. Score level fusion is able to significantly improve recognition results, in particular when combining features describing complementary finger image properties.

1 Introduction

Authentication based on finger veins uses the pattern of the blood vessels inside the fingers of a human. Compared to fingerprint authentication using finger veins has several advantages. The veins are underneath the skin so the vein pattern is resistant to forgery as the veins are only visible in infrared light. Also liveness detection is easily possible. Moreover the vein patterns are neither sensitive to abrasion nor to finger surface conditions, like dryness, dirt, cuts, etc. But there are also some disadvantages. First of all, so far it is not completely clear if vein patterns exhibit sufficiently distinctive features to reliably perform biometric identification in large user groups. With the currently available data sets (see Table 1, the public ones available on-line: ChonbukU[1], PekingU[2], PolyU[3], TwenteU[4]) this issue cannot be clearly answered at present state due to their partially limited size.

Another major disadvantage is that the capturing devices, due to the required transillumination principle, are rather big compared to e.g. fingerprint sensors. Furthermore, the vein structure is influenced by temperature, physical activity and certain injuries and diseases. While the impact of these effects on vein recognition performance has not been investigated in detail so far, consequently it is clear that suitable feature extraction methods should be independent of the vein width to compensate for corresponding variations.

The first issue tackled in this paper is the low contrast and the low overall quality of finger

[1]http://multilab.chonbuk.ac.kr/resources
[2]http://rate.pku.edu.cn
[3]http://www4.comp.polyu.edu.hk/~csajaykr/fvdatabase.htm
[4]http://www.sas.el.utwente.nl/home/datasets

Table 1: Finger vein datasets.

Acronym	Availability	# Images	# Subjects / # Fingers	Publication
ChonbukU	public	6000	100 / 600	[LXW+13]
ChongqingU	private	4260	71 / 426	[QQX+13]
DonggukU	private	4800	60 / 480	[LLP09]
HarbinU	private	100	10 / 10	[PWEL+12]
HITACHI	private	678	339 / 339	[MNM04]
NTUST	private	680	85 / 170	[QQX+13]
PekingU	public	50700	5208 / n.a.	[HDL+10]
PolyU	public	6264	156 / 312	[KZ12]
TwenteU	public	1440	60 / 360	[TV13]

vein images in general. The nature of these images (see e.g. an example of a good and a poor quality image pair in Fig. 1) requires to apply successful preprocessing in order to achieve decent recognition performance, no matter which feature extraction strategy is applied. However, it is difficult to compare the various preprocessing strategies since these are dispersed across literature and are often combined with a single feature extraction technique only (and applied to different, sometimes non-public, datasets). In this paper, we compare a set of preprocessing techniques in a common framework applied with various feature extraction techniques to identify good combinations. We also apply more than a single preprocessing technique, but contrasting to [SPNP14] we do not fuse the results before feature extraction but apply different techniques in a cascaded manner.

In order to improve finger vein recognition results, several fusion techniques have been investigated in literature. Fusion of several fingers' results is an obvious possibility [YYW12, QQX+13], however, sensing is more expensive in this approach of course. Using multiple features extracted from a single image in a score level fusion (by analogy to fusion of e.g. palm vein data [ZK10]) has resulted in first promising results: Fusion of vein pattern structure, LBP, and super-pixel features [LYYX13] as well as fusion of vein-shape, vein orientation, and SIFT features [QQX+13] were able to improve the usage of single features. Especially in the latter paper in case of using high accuracy features improvements have been found to be moderate only. In this work we show that features representing orthogonal finger-vein image properties lead to significantly improved accuracy in a score level fusion scheme, even in case the single techniques involved exhibit partially poor recognition accuracy.

The rest of the paper is organised as follows: Section 2 describes techniques for finger vein recognition (i.e. preprocessing and feature extraction approaches) while Section 3 describes the experimental setting and provides results with respect to the selection of preprocessing strategies and score level fusion. Section 4 concludes this work and gives an outline of future research.

2 Finger Vein Recognition

2.1 Preprocessing Techniques

Finger vein image preprocessing techniques can be grouped into methods for alignment of the finger position and into methods to improve the often uneven and poor contrast of the images. The first preprocessing stage used for all subsequent feature extraction algorithms is to detect the actual region of the finger in the images. Especially for subsequent contrast manipulations this is important, since otherwise background pixels would influence the computation of the parameters for contrast change. **LeeRegion detection** [LLP09] uses the fact that the finger region is brighter than the background and determines the finger boundaries using a simple 20 x 4 mask, containing 2 rows of 1 followed by 2 rows of -1 for the upper boundary and a horizontally mirrored one for the lower boundary. The position at which the masking value is maximal is determined as the finger edge (boundary), background pixels are set to black.

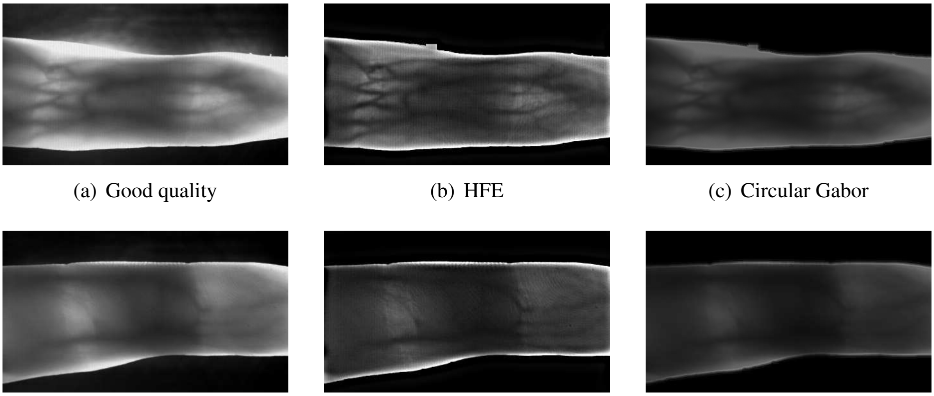
Due to slight variations in positioning of the finger on the capturing device, the orientation of the finger in the image is not always the same. Therefore **Finger Position Normalisation** [HDL+10] aligns the finger to the center of the image, compensating rotations and translations. It uses the finger edges detected by LeeRegion and fits a straight line between the detected edges. The parameters of this line are then used to perform an affine transformation which aligns the finger. In the experiments, this approach is used. A slightly different method is to compute the orientation of the binarised finger ROI using second order moments and to compensate the orientation in rotational alignment [KZ12].

As an alternative, minutiae-based alignment [LLP09] can be applied, however, as this approach requires the extraction of vessel structure *before* geometrical alignment contradicting the cascaded application as done here we do not consider it further. It should be noted that we do not need to apply alignment methodology when applying SIFT and SURF feature extraction as both are designed to be invariant against affine transformations.

The second preprocessing stage is to improve and equalise contrast. **CLAHE** [Zui94] or some other local histogram equalisation techniques [KZ12] are suggested by most authors as a preprocessing step for finger vein images. Due to the unevenly distributed contrast a localised technique like CLAHE is a perfect tool, additionally the integrated contrast limitation avoids amplification of noise.

High Frequency Emphasis Filtering (HFE) was proposed originally for hand vein image enhancement [ZTXL09]. After computing the discrete Fourier transform, a Butterworth high-pass filter of order n is applied. Instead of using global histogram equalisation on the filtered image as the authors suggest, we apply CLAHE afterwards due to superior results. See Figs. 1.b and 1.e for examples when computing HFE on two images from the Twente database.

The **Circular Gabor Filter** [ZY09], being rotation invariant and achieving optimal joint localisation in both spatial and frequency domain, was proposed in combination with grey level grouping for contrast enhancement, especially at the veins edges. As before, instead of grey level grouping we apply CLAHE after filtering due to better results. See Figs. 1.c and 1.f for examples when applying this approach.



(a) Good quality (b) HFE (c) Circular Gabor

(d) Poor quality (e) HFE (f) Circular Gabor

Figure 1: Preprocessing examples when applied to two images from the Twente dataset.

Further (tested) image enhancement techniques include Unsharp Masking, Retinex contrast enhancement [WTZ12] as well as denoising with a median filter (5 x 5) and an adaptive Wiener filter (7 x 7) but as these did not improve the results in experimentation we have not further looked into their details.

2.2 Feature Extraction

We first discuss techniques which aim to explicitly extract the veins from the background using different approaches generating a binary image and then compare the resulting binary images using correlation. Subsequently, key-point based techniques are described. Note that none of the feature extraction approaches followed here relies on finger vein "minutiae" (i.e. vein crossings and endings) as the accuracy of this approach has only been shown on a very small dataset [QQX+13]. A comparison of accuracy results of the described techniques as published in literature is given in Table 2.

Repeated Line Tracking (RLT [MNM04]) is based on dark line tracking starting repeatedly at various random positions. Veins appear as valleys in the cross-sectional profile of the image. The randomly initialised tracking point is moved pixel by pixel along the dark line, where the depth of the valley indicates the movement direction (pixel is moved to where the valley is deepest). If no "valley" (exhibiting a certain width) is detectable a new tracking operation is started. The number of times a pixel is tracked is recorded in a matrix (called locus space). Pixels that are tracked multiple times as being a line statistically have a high likelihood of belonging to a blood vessel (high value in locus space image). Therefore, binarisation is applied to the locus space image to get the final binary output image. See Figs. 2.b and 2.h for an example binarisation of this type.

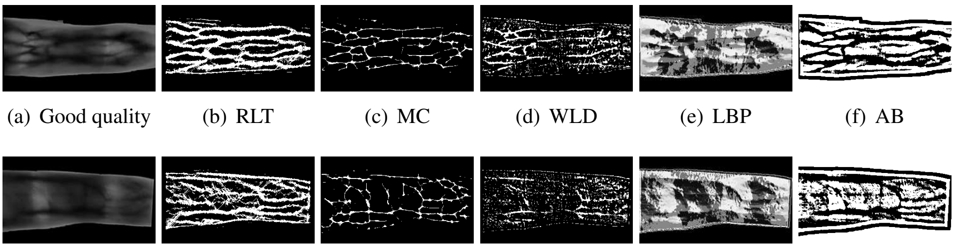
Table 2: Finger vein recognition accuracy comparison in terms of EER (%).

Publication	Dataset	RLT	MC	WLD	LBP	EGM	MF	SIFT
[MNM04]	HITACHI	0.15	-	-	-	-	-	-
[MNM07]	HITACHI	0.01	0.0001	-	-	-	-	-
[TV13]	TwenteU	1.2	0.4	0.9	-	-	-	-
	PekingU	5.9	1.2	2.7	-	-	-	-
[HDL+10]	PekingU	-	2.8	0.87	-	-	-	-
[KZ12]	PolyU	6.54	2.2	-	-	0.43	1.88	-
[LLP09]	DonggukU	-	-	-	0.08	-	-	-
[PWEL+12]	HarbinU	-	-	-	-	-	-	0.46
[QQX+13]	ChongqinU	-	-	-	-	-	-	16.09
	NTUST	-	-	-	-	-	-	10.98

Maximum Curvature (MC [MNM07]) emphasises the center-lines of the veins and is therefore insensitive to changes in the width of the veins. At first the center position of veins is extracted. For this purpose, the local maximum curvature in cross-sectional profiles in four directions, horizontal, vertical and the two oblique directions, based on the first and second derivates, is determined. Then each profile is classified as being concave or convex (curvature positive or negative) where local maxima in concave profiles indicate the center positions of the veins. Each center position is then assigned a score, according to the width and curvature of the region.

Subsequently, the center positions of the veins are connected. Due to noise or other distortions some pixels may not have been classified correctly at the first step, so a filtering operation is applied in all four directions taking adjacent context pixels into account. The last step is the binarisation of the vein pattern using the median of the locus space as a threshold.

The **Wide Line Detector** (WLD [HDL+10]) works similar to adaptive thresholding (using isotropic nonlinear filtering), i.e. thresholding inside a local neighbourhood region. The difference of each pixel inside a circular neighbourhood to the central pixel is determined. Subsequently, the number of pixels inside this neighbourhood which have a difference smaller than a set threshold are determined. This number is again thresholded to get the final binary vein image See Figs. 2.d and 2.j for an example binarisation of this type.



(a) Good quality (b) RLT (c) MC (d) WLD (e) LBP (f) AB

(g) Poor quality (h) RLT (i) MC (j) WLD (k) LBP (l) AB

Figure 2: Feature extraction examples.

Given the title of the paper [KZ12], it is somewhat surprising to find highly competitive finger vein feature extraction techniques: **Gabor Filtering** (EGM) is suggested by taking the maximal response in each pixel by using a convolution of self similar 2D even Gabor filters for different orientations with a zero-mean version of the image. Subsequently, a morphological top-hat operation is performed (i.e. subtracting the Gabor response image from a version that has undergone morphological opening) to generate the binary result. **Matched Filters** (MF [KZ12]) are used to match cross section profiles of finger veins employing a group of 1-D Gaussian functions. Similar to the Gabor case before, the employed Gaussian functions are rotated in different orientations and only the maximum response is utilised. Binarisation in conducted in the same manner.

To assess the advanced binarisation techniques which try to model vein shape in their binarisation strategy as described so far, we have additionally used basic **Adaptive Binarisation** (AB [Ots79]) to see the (eventual) effect of the more advanced schemes. See Figs. 2.f and 2.l for an example binarisation of this type and please note, that contrasting to the other binarisation techniques discussed (i) veins appear black and (ii) also the outline of the finger causes a strong black curve indicating its shape. Thus, in addition to show the vein structure we additionally get finger shape information for free.

As the last binarisation-type feature extraction scheme, we consider a **Local Binary Patterns** (LBP) feature extraction scheme [LLP09]. LBP compare the grey level of a center pixel to its neighbouring pixels. The original LBP is a 3x3 non-parametric operator. It can also be defined as an ordered set of binary values determined by comparing the gray values of a center pixel to its 8 neighbouring pixels. Each image gray scale value is replaced by the corresponding binary code resulting from the binary pattern of the neighbourhood (which results in binary images of 8-fold size of the binarisation results discussed so far).

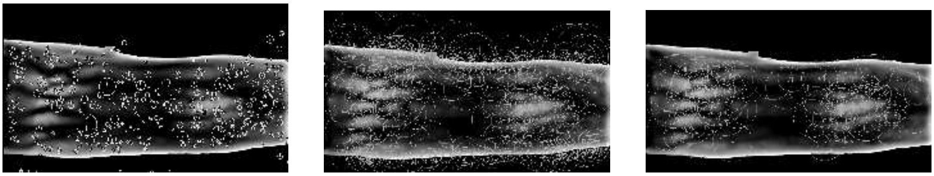
Lee at al. [LLP09] used an LBP approach for finger vein matching, preceded by a minutia-based alignment step and a Hamming distance matching. We followed their approach for feature extraction but moved the alignment into the matching stage and used a multi-scale LBP version instead (due to better results). See Figs. 2.e and 2.k for an example binarisation of this type.

For matching a pair of binarized image features we adopt the approach in [MNM04] and [MNM07]. As the input images are not registered to each other and only coarsely aligned, we simply calculate the correlation between the input image and the reference one while shifting the reference image in x- and y-direction. The maximum value of the correlation is normalised and used as matching score.

While binarisation techniques (with the exception of LBP) compute a single bit of information for each pixel in the original image, the importance of neighbourhood information is already accounted for in how these single bits are computed in most techniques as discussed before as well as in the super-pixel approach of Liu et al. [LYYX13]. As already discussed, an entirely different approach is to use vein minutiae representing the most discriminative local information contained in the images. A different approach combining both ideas, i.e. highlighting the most discriminative points as well as the importance of neighbourhood and context information, is the employment of key-points and their descriptors. In this work we consider SIFT [Low99] and SURF [BETVG08]. See Figs. 3.a and 3.b for a visualisation of extracted keypoints.

SIFT keypoints have been considered in palm vein image recognition [ZK11] with moderate success only. Peng et al. [PWEL$^+$12] propose an approach for finger vein images based on CLAHE and an orientation-selective Gabor filtering (responses of two two Gabor filters with orientations $\frac{2\pi}{8}$ and $\frac{7\pi}{8}$ are fused) with subsequent matching based on SIFT feature extraction on the filtered images. Qin et al. [QQX$^+$13] use SIFT to identify distinctive corresponding sub-regions in finger vein images and apply matching to those regions only.

As can be seen from the SURF feature extraction example (see Fig. 3.b), there are many strong keypoints along the finger boundaries. As descriptors of such keypoint contain irrelevant background information, including them can lead to false or ambiguous matches. Thus we implemented a filtering step within a window along the finger boundaries (whose width can be defined). All keypoints inside the window are discarded (keypoint filtering, see Fig. 3.c). In addition, a minimum number of keypoints can be defined. If too few keypoints are found, feature extraction is re-run with adapted parameters to extract more keypoints until at least the minimum number of keypoints are found or the parameters cannot be changed any more.



(a) SIFT (b) SURF (c) SURF filtered

Figure 3: SIFT & SURF feature extraction.

SIFT and SURF matching is done using the keypoint descriptors – the keypoint with the smallest distance to the reference keypoint is the matched one if the distance is below a threshold, otherwise there is no match. To resolve the problem with ambiguous matches (i.e. one keypoint may have small distances to more than one other point) a ratio threshold is used: A match is only valid if the distance of the best point match is at least k (threshold) times smaller than to all other points.

After matching we get a set of matched keypoints with associated distances. The simplest way (and also best one, at least for SURF) to get a final matching score is to use the number of matched keypoints only. A slightly better way (for SIFT) is to use the ratio of matched keypoints to the maximum possible number of matches (minimum number of keypoints of the two images) – "ratio score calculation". All techniques involving also distances of matched keypoints performed worse and thus have not been considered further.

Comparing the accuracy of the different techniques as given in literature, Table 2 shows that reported EERs (1) highly depend on the dataset (2) are not even consistent among different implementations when applied to the same dataset (3) tend to be lower in the original publications as compared to re-implementations. Thus, at current stage, it is almost impossible to make recommendations concerning the "best" technique based on the data available.

3 Experiments

3.1 Experimental Settings

For our evaluation we use the TwenteU dataset. This dataset (see Table 1) consists of a total of 1440 images, taken from 60 persons, 6 fingers per person (index, ring and middle finger of each hand) and 4 images of each finger. The images were captured in 2 sessions with a time-lag of 15 days between the sessions using a custom designed transillumination device. Each finger was captured twice during one session. 73% of the subjects were male and 82% were right handed. The images have a resolution of 672 x 380 pixels, a density of 126 pixels/cm and are stored in 8 bit grey scale PNG format. The width of the visible blood vessels is 4 - 20 pixels.

RLT, MC, WLD feature extraction and matching are used as provided in a MATLAB implementation by B.T. Ton[5]. SIFT and SURF software is used as provided by OpenCV, custom implementations are used for the preprocessing techniques, LBP, SIFT and SURF keypoint filtering, and AB. Images have been downsampled by a factor of 2 in both dimensions as the MATLAB software is customised to the expected vein width of this image scale.

In order to be able to compare our results to those in [TV13] (compare Table 2 [TV13] to Table 5) we applied the same EER determination procedure: Only one finger of each of the first 35 subjects is used as training set for parameter tuning. The actual EER determination with the rest of the dataset consists of 1950 genuine matches and 842400 impostor matches compared into FAR and FRR, respectively.

Considering score level fusion as used in this work, the simple MinMax normalisation turned out to be the best. We tested different score combinations, from pairs of 2 scores to tuples of all available scores, each one with different fusion schemes (including weighted sum, weighted product, weighted mean/median and weighted minimum/maximum) and systematically assessed all variants with different weight combinations on the parameter optimisation test set.

3.2 Experimental Results

The first results (Tables 3 and 4) are devoted to preprocessing strategies. For these results, computational load has been reduced due to costly exhaustive configuration testing – N preprocessing stages can be combined into N! different combination configurations – by restricting the impostor matches to comparing each image with the corresponding finger/image of each other person only but used the entire dataset instead of 90%. Thus, these results cannot be compared directly to the results in Table 5.

The results in Tables 3 and 4 are cumulative, i.e. for a given line all the above lines were

[5]Publicly available on MATLAB Central: `http://www.mathworks.nl/matlabcentral/fileexchange/authors/57311`

also applied to achieve the final result. An important thing to note is that parameters are not independent from each other, which means if e.g. the feature extraction parameters are changed also the matching parameters have to be re-adjusted.

Table 3: Preprocessing Impact on EER (%) of SIFT.

Without any preprocessing	18.82
Filtering of keypoints along the finger edges	10.66
Feature Extraction parameter adjustments	9.65
Matching parameter adjustments	9.00
CLAHE	8.26
Resizing (0.5)	6.44
HFE Filtering	2.74
Gabor Filter	2.59
Ratio Score Calculation	2.04

As it can be seen easily in Table 3 for SIFT-based recognition, keypoint filtering and HFE filtering exhibit the most significant positive impact on the EER. Interestingly, we are able to improve EER in several successive stages from applying SIFT without any preprocessing at EER of 18.82% to the highly tuned variant at EER of 2.04% involving several preprocessing and matching optimisations and variants.

Table 4: Impact of Preprocessing on EER (%).

Preprocessing Method	MC	WLD	RLT	LBP	AB	SURF
none	6.443	32.19	22.63	34.1	7.223	13.81
Resize (0.5)	2.041	10.18	4.03	32.67	4.872	9.843
LeeRegion	1.989	9.96	4.11	31.45	4.816	9.843
Normalisation	0.7831	4.777	2.04	29.67	3.606	11.45
CLAHE	0.6011	**2.687**	**1.856**	11.55	3.376	11.88
Circular Gabor Filter	**0.4692**	3.116	1.999	9.895	**3.335**	11.58
HFE Filter	0.7399	3.804	5.785	10.75	3.807	**4.184**
Denoising	0.9255	2.918	5.662	**6.043**	3.508	4.275

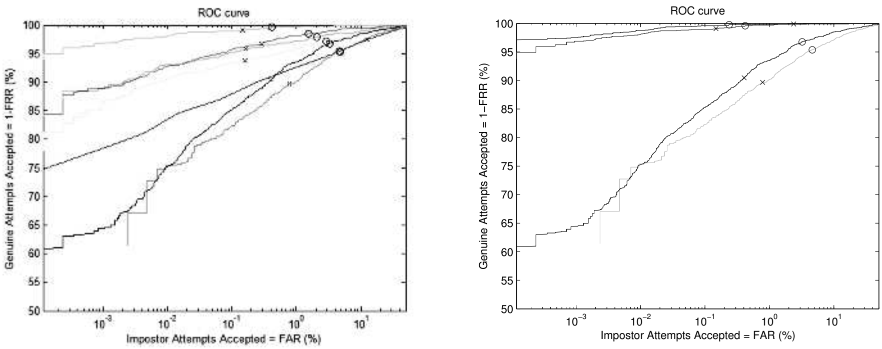
Table 4 compares preprocessing cascades for the other recognition approaches considered. While up to LeeRegion detection all feature extraction techniques are able to take advantage from preprocessing, the optimal EER is achieved at different stages of the cascade, SURF results degenerate until HFE application where the minimal EER value is reached at 4.18%. RLT and WLD reach their optimum at EER of 2.69% and 1.86% when using CLAHE, respectively. The additional circular Gabor filter application delivers the best results for MC and AB at 0.49% and 3.34%, respectively. Finally, for LBP, the best result (EER 6.04%) is achieved when applying a final denoising stage after having applied the circular Gabor filter and CLAHE. Summarising we may state that there is no unique best preprocessing cascade to be identified, but in any case it gets clear that some feature extraction techniques benefit from a cascaded preprocessing application while others suffice with a restricted set only.

Table 5 lists the results of the overall best preprocessing cascades / feature extraction combinations achieved over the entire evaluation dataset. We achieve the top EER at 0.42% for MC and the worst one for LBP at 5.03%. It is interesting to see that the generic SIFT approach outperforms the highly specialised WLD and that even the simple AB technique is not much worse.

Table 5: Comparison of EERs (%) of recognition methods considered.

	MC	WLD	RLT	SIFT	SURF	LBP	AB
EER	**0.42%**	2.87%	1.64%	1.96%	4.34%	**5.03%**	3.23%

Figure 4.a displays ROCs for all combinations in Table 5 (WLD: yellow, LBP: magenta, MC: cyan, RLT: red, SIFT: green, SURF: blue, and AB: black). While in terms of EER, AB is superior to SURF, for lower FAR values SURF gets superior quickly. The remaining schemes maintain the EER ranking almost consistently. Figure 4.b shows the ROC of the best fusion technique which is obtained using weighted sum fusion (weights: 1.2, 3.2, 1.1) of LBP, MC, and AB at an EER of 0.25%. Also the ROCs of the three single techniques are included for comparison (Fusion: red, LBP: green, MC: blue, AB: black).



(a) Comparison

(b) Best Fusion

Figure 4: ROC results

It is interesting to observe that the best fusion result is not obtained by combining the three best performing techniques but in contrast, by combining the best performing single technique with the two worst ones. Obviously, it is not the single performance that determines the final fusion result. While MC captures the structure of the veins most efficiently, AB contributes finger shape and LBP adds general texture information also between veins (mainly local information around the pixels and not lines are extracted). Thus, those three technique can really be said to contribute quite orthogonal properties of the finger region. Another good result was achieved using a combination of MC, LBP and

SIFT with weighted product fusion (weights 0.7, 2.8, 1.4) resulting in an EER of 0.27%. Here similar considerations concerning opposing properties do apply.

4 Conclusion

Preprocessing has a major impact on the recognition performance of different feature extraction techniques. There is neither a single preprocessing approach nor a single preprocessing cascade that is the optimal technique for all subsequent feature extraction techniques. In contrary, preprocessing needs to be carefully optimised for each single feature extraction method. We have found that cascading different preprocessing techniques significantly improves recognition performance as compared to applying a single approach in many cases.

Score level fusion has turned out to improve recognition performance considerably, especially when combining features which represent complementary properties of the images' finger regions. In these combinations, not only vessel structure but also finger shape and non-vessel texture is represented, leading to EERs superior to the best EERs of single techniques by almost 100% .

With respect to fusion, we aim to further investigate the inclusion of Gabor and Matched Filter feature extraction as well as vein-minutiae informations to further improve results.

References

[BETVG08] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-Up Robust Features (SURF). *Comput. Vis. Image Underst.*, 110:346–359, June 2008.

[HDL+10] Beining Huang, Yanggang Dai, Rongfeng Li, Darun Tang, and Wenxin Li. Finger-vein authentication based on wide line detector and pattern normalization. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 1269–1272. IEEE, 2010.

[KZ12] A. Kumar and Y. Zhou. Human identification using finger images. *IEEE Transactions on Image Processing*, 21(4):2228–2244, 2012.

[LLP09] Eui Chul Lee, Hyeon Chang Lee, and Kang Ryoung Park. Finger vein recognition using minutia-based alignment and local binary pattern-based feature extraction. *International Journal of Imaging Systems and Technology*, 19(3):179–186, 2009.

[Low99] D. G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision and Pattern Recognition*, volume 2, pages 1150 – 1157. IEEE, 1999.

[LXW+13] Y. Lu, J. Xie, Z. Wang, S. Yoon, and D.S. Park. An Available Database for the Research of Finger Vein Recognition. In *Proceeding of the 2013 6th International Congress on Image and Signal Processing (CISP'13)*, 2013.

[LYYX13] F. Liu, G. Yang, Y. Yin, and X. Xi. Finger-vein recognition based on fusion of pixel level feature and super-pixel level feature. In *Proceedings of the Chinese Conference* *on Biometric Recognition (CCBR'13)*, volume 8232 of *Springer LNCS*, pages 274–281, 2013.

[MNM04] Naoto Miura, Akio Nagasaka, and Takafumi Miyatake. 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, 2004.

[MNM07] Naoto Miura, Akio Nagasaka, and Takafumi Miyatake. Extraction of finger-vein patterns using maximum curvature points in image profiles. *IEICE transactions on information and systems*, 90(8):1185–1194, 2007.

[Ots79] N. Otsu. A Threshold Selection Method from Gray-Level Histogram. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, January 1979.

[PWEL$^+$12] Jialiang Peng, Ning Wang, Ahmed A Abd El-Latif, Qiong Li, and Xiamu Niu. Finger-vein verification using Gabor filter and sift feature matching. In *Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2012 Eighth International Conference on*, pages 45–48. IEEE, 2012.

[QQX$^+$13] H. Qin, L. Qin, L. Xue, X. He, C. Yu, and X. Liang. Finger-Vein Verification Based on Multi-Features Fusion. *Sensors*, 13(11):15048–15067, 2013.

[SPNP14] K.Y. Shin, Y.H. Park, D.T. Nguyen, and K.R. Park. Finger-Vein Image Enhancement Using a Fuzzy-Based Fusion Method with Gabor and Retinex Filtering. *Sensors*, 14(2):3095–3129, 2014.

[TV13] B.T. Ton and R.N.J. Veldhuis. A high quality finger vascular pattern dataset collected using a custom designed capturing device. In *International Conference on Biometrics, ICB 2013*. IEEE, 2013.

[WTZ12] Hua-Bin Wang, Liang Tao, and Jian Zhou. Novel algorithm for hand vein feature extraction and recognition based on vertical integral projection and wavelet decomposition. In *Consumer Electronics, Communications and Networks (CECNet), 2012 2nd International Conference on*, pages 1928–1931. IEEE, 2012.

[YYW12] Y. Yang, G. Yang, and S. Wang. Finger-Vein recognition based on multi-instance. *JDCTA: International Journal of Digital Content Technology and its Applications*, 6(11):86–94, 2012.

[ZK10] Y. Zhou and A. Kumar. Contactless palm vein identification using multiple representations. In *Proceedings of the 4th International Conference on Biometrics: Theory, Applications and Systems (BTAS'10)*, pages 1–6, 2010.

[ZK11] Y. Zhuo and A. Kumar. Human identification using palm-vein images. *IEEE Transactions on Information Forensics and Security*, 6(4):1259–1274, 2011.

[ZTXL09] Jianjun Zhao, Hogliang Tian, Weixing Xu, and Xin Li. A New Approach to Hand Vein Image Enhancement. In *Intelligent Computation Technology and Automation, 2009. ICICTA'09. Second International Conference on*, volume 1, pages 499–501. IEEE, 2009.

[Zui94] K. Zuiderveld. Contrast Limited Adaptive Histogram Equalization. In Paul S. Heckbert, editor, *Graphics Gems IV*, pages 474–485. Morgan Kaufmann, 1994.

[ZY09] Jing Zhang and Jinfeng Yang. Finger-vein image enhancement based on combination of gray-level grouping and circular Gabor filter. In *Information Engineering and Computer Science, 2009. ICIECS 2009. International Conference on*, pages 1–4. IEEE, 2009.