FINGERPRINT PREPROCESSING ALGORITHM WITH FOCUSE ON FUTURE LIVENESS DETECTION

Jakub Kašpar

Doctoral Degree Program (1), FEEC BUT

E-mail: xkaspa36@stud.feec.vutbr.cz

Supervised by: Martin Vítek

E-mail: vitek@feec.vutbr.cz

Abstract: The aim of this paper is to introduce the problem of fingerprint liveness detection and propose a method for preprocessing of fingerprints, which will be used for liveness detection. In the first part of this paper the basic problematic of false fingerprints is described. Further in the paper the principle of fingerprint preprocessing is introduced. The last part of this paper presents problematic fingerprint types and how well is the algorithm able deal to with it. The algorithm was tested on the LivDet database.

Keywords: preprocessing, fingerprints, liveness

1 INTRODUCTION

Fingerprints of each person on the planet are unique, which makes it a perfect object for identification process. Unfortunately, the process of fingerprints acquiring through scanning can be easily deceived with false fingerprint models. The failure of the identification system can lead to scenarios from minor security problems (e.g. breaking in company storage room), to some major security breaches (e.g. at international airports), which could lead to catastrophic consequences that could cost many innocent human lives.

This is the main reason that liveness detection should be implemented into fingerprint identification software. Apart from individual liveness sign detection, the really important step of this process is the right fingerprint preprocessing. The best version of image for some liveness signs is binary version of fingerprint, but fingerprint images usually contain variety of noise and artifacts, which makes transformation of grey scale image to binary image rather difficult. [2]

2 PREPROCESSING

Fingerprint images come in a large variety of forms and different formats and often contain a lot of redundant information. Ideal preprocessing algorithm should be able to separate all noise and artifacts and give only correctly segmented binary fingerprint. The individual steps of proposed algorithm are shown in Figure 1.

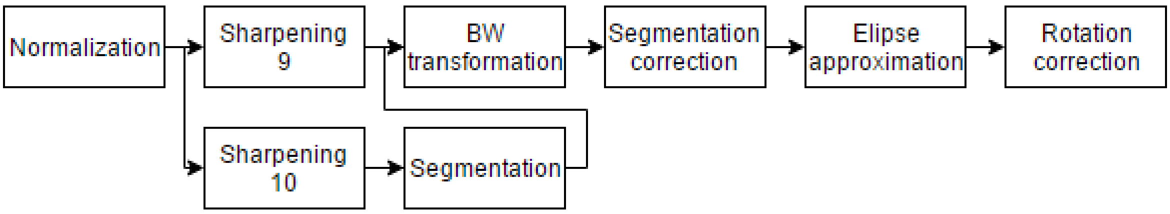


Figure 1: The complete process of fingerprint preprocessing

2.1 IMAGE NORMALIZATION

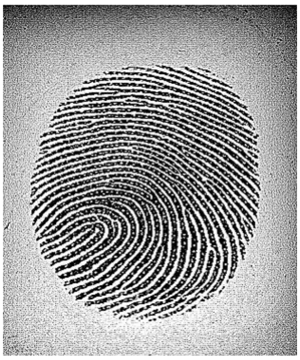
First step of fingerprint preprocessing is normalization. Usually the fingerprint images are quite dark and it is difficult to separate fingerprint ridges from image background. From that reason, the contrast is increased via finding the lowest and the highest intensity value and assigning them values 0 and 1. The rest of intensity values are evenly distributed in this range. The example of the original fingerprint and normalized fingerprint is shown in Figure 2.

Figure 2: The example of original and normalized fingerprint

2.2 SHARPENING

After fingerprint is normalized, two sharpening masks 3x3 px are applied. Central pixel value of the first sharpening mask is 9 and of the second mask is 10 [1]. These masks are applied separately on the normalized fingerprint with two different outputs as it was shown in block diagram in Figure 1. Fingerprints after applying sharpening mask are shown in Figure 3.

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$



$$\frac{1}{2}\begin{bmatrix} -1 & -1 & -1 \\ -1 & 10 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Figure 3: Sharpening Mask and fingerprint after its application with central pixel value of 9 (left) and 10 (right)

These masks are used for two reasons. Firstly it differentiates the ridges of the fingerprint, so when image is transformed to binary version ridges does not pool together. This is achieved by applying mask with central value 9 which is standard sharpening mask. The second reason is further change of background. This is achieved by applying mask with central value 10. This mask eliminates the noisy background and still does not erase any part of fingerprint, which the mask with greater central value does. Fingerprint on the left is used for binary transformation. Fingerprint on the right is used for gray threshold calculation and further segmentation of fingerprint.

2.3 SEGMENTATION AND TRANSFORMATION TO BINARY IMAGE

From the right picture in the Figure 3 threshold for transformation is calculated using Otsu's method [3]. Then the image is transformed to binary version. Then the morphological operation "close" is used [4]. Output of this operation is image of regions, from which the largest corresponds with the area where fingerprint is located. Then transformation to binary image is used once again on the left picture from the Figure 3 with the same threshold acquired in right picture, but this time only in segmented area acquired in the previous step. This is better for cases, when segmented area is not perfectly around the fingerprint. Darker background helps to close unwanted area around the segmented fingerprint, but in some cases segmentation still has to be corrected. The example of binary transformation using calculated threshold and its closed version is in Figure 4.

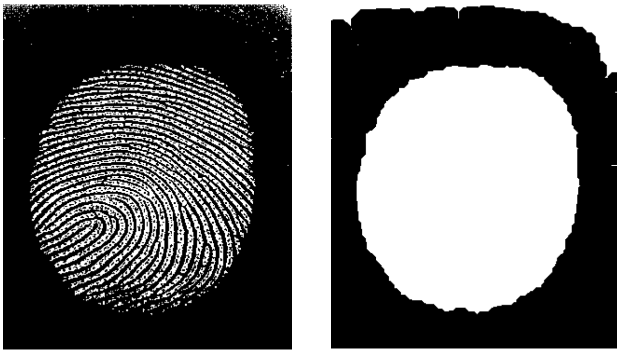


Figure 4: The example of binary transformation of fingerprint (left) and its closed version (right).

2.4 SEGMENTATION CORRECTION

After segmentation and transformation, whole fingerprint image is morphologically closed once again. If the segmentation in previous step is correct, this operation changes nothing, but if the segmentation mask was larger than fingerprint, this step helps to divide the excess area from the fingerprint area. Then as in previous segmentation step, the largest area is selected as fingerprint and the rest is eliminated. The example of the incorrectly segmented fingerprint before and after correction is shown in Figure 5.

Figure 5: Segmentation before (left) and after (right) correction

2.5 ELLIPSE APPROXIMATION AND ROTATION

In this step, major and minor axis length of the fingerprint is measured. Also from the position of the axis the orientation of fingerprint is calculated. From these values, ellipse is created and centered on the fingerprint. This is the final step of segmentation, because everything that exceeds the ellipse is cut off. Also the ellipse and fingerprint are rotated such a way, that major axis is in the angle of 90°. Rotation step is just for position normalization of all fingerprint types. The example of fingerprint ellipse approximation and further rotation is shown in Figure 6.

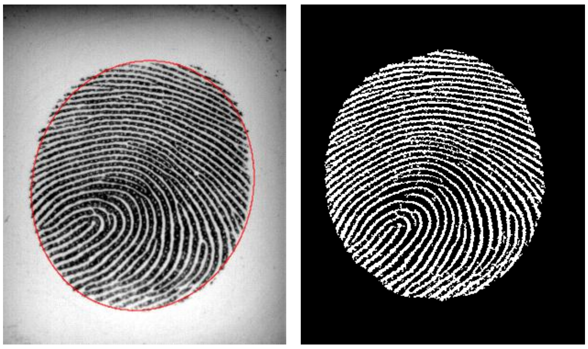


Figure 6: The example of normalized fingerprint approximated by ellipse (left) and its rotated binary version (right)

3 EVALUATION

The proposed algorithm was tested on the fingerprint database LivDet 2013. It is database made for competition in fingerprint liveness detection. It contains large variety of fingerprint types which are problematic for preprocessing and liveness detection. Database does not contain reference preprocessed data (only raw images), so evaluation of proposed algorithm is only subjective. From the 1000 tested fingerprints, only 17 were not classified as correctly preprocessed. Main criteria for correct preprocessing was, that there cannot be missing more than 5 - 7 % of fingerprint in segmented area. There was also checked pooling of ridges and correct segmentation. Quality of the proposed preprocessing software is therefore 98.3 %In the Figure 7 are shown examples of problematic fingerprint types. In the future, the quality of proposed preprocessing will be evaluated objectively based on accuracy of liveness detection.

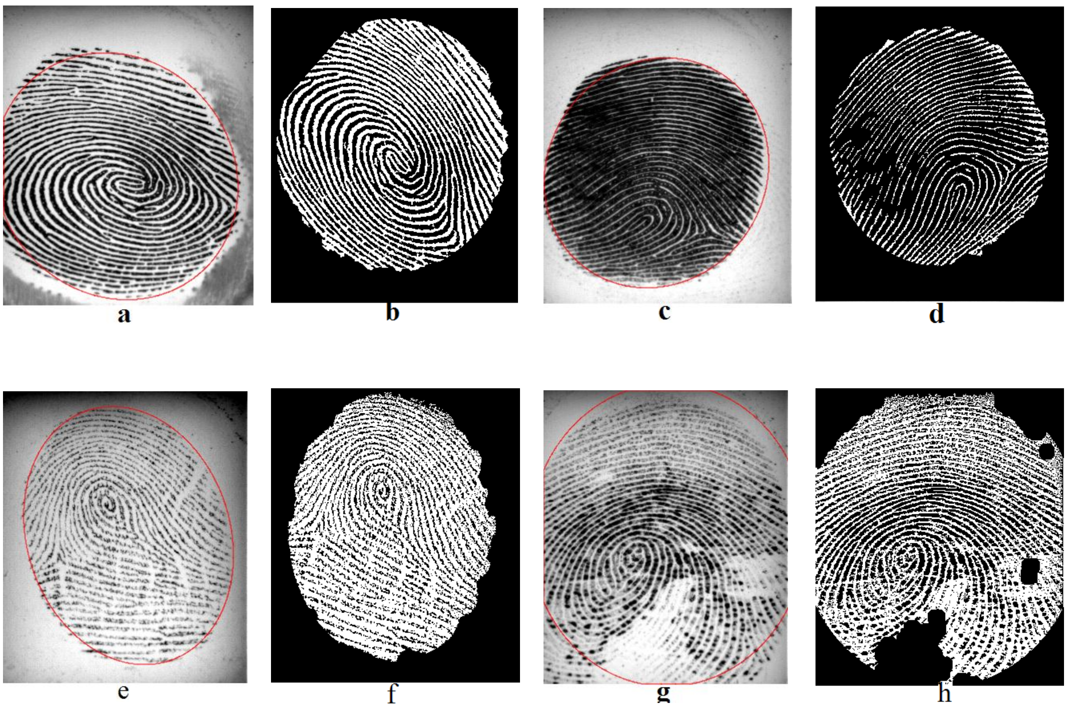


Figure 7: Examples of problematic fingerprints in original and binary version. a – b Dark fingerprint surroundings, c – d Fingerprint with very thick ridges, e – f Fingerprint with not enough ink, g – h Fingerprint with uneven distribution of ink.

3.1 DARK FINGERPRINT SURROUNDINGS

This artifact, made by mess on a fingerprint scanner, would usually cause incorrect segmentation. Dark areas around the fingerprint would be classified as a part of the fingerprint because of the similar shade. This problem is in proposed algorithm solved, because the fingerprint is segmented from the image sharped with the mask with central value 10. This step almost completely eliminates dark areas and makes segmentation possible.

3.2 FINGERPRINTS WITH VERY THICK RIDGES

This is a problem caused by applying too much ink on finger or by applying too much pressure while making a fingerprint. The final fingerprint has almost no space between ridges. It may happen, that in binary version of the fingerprint the ridges would pool together into one big black blob. This problem is partially solved by using sharpening mask with central value 9. Sharpening is able to more differentiate the gaps between the ridges, to the limit where the gaps are barely visible. Beyond that, sharpening is no longer effective.

3.3 NOT ENOUGH INK ON FINGERPRINT

In this case, the quality of scanned fingerprints is not good enough. Fingerprint seems grainy and they are very bright. If fingerprints are bright rather evenly, it is no problem for proposed algorithm to transform them correctly into binary version. Problem is that output image has the same quality as the input image. For quality increase, it would be necessary to implement some fingerprint image enhancer, but for the detection of liveness it is not necessary.

3.4 UNEVEN INK DISTRIBUTION ON FINGERPRINT

This is the most problematic case for proposed algorithm. Majority of fingerprint (around 90 %) is very dark, but the rest are very bright areas. It is basically combination of two previous problems. This causes problems with segmentation. These bright areas are classified as “not part of the fingerprint” because they are below the threshold for binary transformation. This is the only major problem of proposed algorithm and working on the solution in still in progress.

4 CONCLUSION

The result of this paper is novel and verified approach for preprocessing of fingerprints, with focus on liveness detection. The software was tested on LivDet database with large variety of problematic fingerprints. Evaluation of testing data was done subjectively, because the lack of reference data. Testing was done on 1000 fingerprint images and the quality was estimated on 98.3 %. The proposed algorithm is able to correctly preprocess all kinds of problematic fingerprints with the exception of extremely unevenly ink distributed ones. The proposed preprocessing will be used in the future for detection of specific liveness signs, which are best to detect on binary image. The segmentation and rotation will be used for all liveness signs.

REFERENCES

[1] WALEK, Petr a Jiří JAN. *Analýza medicínských obrazů*. Brno, 2013. ISBN 978-80-214-4792-9

[2] DRAHANSKY, Martin. Liveness Detection in Biometrics. *Advanced Biometric Technologies* [online]. InTech, 2011. DOI: 10.5772/17205. ISBN 978-953-307-487-0.

[3] OTSU, Nobuyuki: A Threshold Selection Method from Gray-Level Histograms. IEEE *Transactions on Systems, Man and Cybernetics*, 1979, roč. 9, č. 1: s. 62-66.

[4] JAN, Jiri: Medical image processing, reconstruction and restoration: concepts and methods. Boca Raton: Taylor & Francis, 2006, ISBN 0-8247-5849-8, 730 s.