Fingerphoto Recognition with Smartphone Cameras

Chris Stein, Claudia Nickel, Christoph Busch

Hochschule Darmstadt - CASED
Mornewegstraße 32
D-64295 Darmstadt

chris.stein@stud.h-da.de, claudia.nickel@.h-da.de, christoph.busch@h-da.de

Abstract: This paper is concerned with the authentication of people on smartphones using fingerphoto recognition. In this work, fingerphotos are captured with the built-in camera of the smartphone. The proposed authentication method is analyzed for feasibility and implemented in a prototype as application for the Android operating system. Algorithms for the capture process are developed to ensure a minimum of quality of the captured photos to enable a reliable fingerphoto recognition. Several methods for preprocessing of the captured samples are analyzed and performant solutions to evaluate the photos are developed to enhance the recognition rates. This is achieved by evaluating a wide range of different parameters and configurations of the algorithms as well as various combinations of preprocessing chains for the captured samples. The operations for preprocessing are selected with respect to their computational effort to guarantee that they can be executed on a smartphone with limited computation and memory capacity. The developed prototype is evaluated in user tests with two different smartphones. Additionally, a biometric database containing photos of the two test devices from 41 test subjects is created. These fingerphotos are used to evaluate and optimize the procedures.

1 Motivation

Smartphones are nowadays very popular and the smartphone-market is growing rapidly. These devices have many advanced functions that go beyond telephone communication. Due to the versatile functions, many personal data like e-mails, photos or even passwords are stored on these personal assistants. Such data must be protected. The common methods for authentication on smartphones are PIN, password or recognition pattern, which are all based on knowledge. But a survey from Breitinger and Nickel [BN-2010] shows that 86% of the participants use as protection mechanism only a keylock or even no lock-mechanism on their phone. The main reasons are that the access to the phone is faster and that no thoughts on security are made. According to the mentioned survey, 54% of the participants would use biometric authentication methods and would prefer fingerprint recognition as modality.

This work proposes a biometric authentication with fingerphoto recognition. For this purposes, the built-in camera of the smartphones is used. The latest smartphones have at least one integrated high resolution camera to capture the finger in sufficient quality and enough computational capacities to process the photos and execute algorithms for the fingerphoto recognition. Hence, there are no extra devices needed to perform the solution proposed in this work. The capture process with fingerphotos is performed touchless. Hence, no latent fingerprint is left, which is an advantage over many classical fingerprint sensors. Additionally, the biometric authentication method has clear advantages with reference to security compared to the knowledge-based method. Biometric characteristics cannot be delegated, forgotten or copied like e.g. passwords.

The rest of the paper is organized as follows. In Section 2, the related work to our work is briefly presented. The challenges of the fingerphoto recognition are discussed in Section 3 and the objectives of this work as well as the approach to achieve the objectives are addressed in Section 4. In the following Sections 5 to 9 we introduce the solutions of the capture process, the finger recognition, the preprocessing of the images, the minutia extractor and the template comparator. Finally, the evaluation and results of the developed prototype is shown in Section 10 as well as the conclusions and future work in Section 11.

2 Related Work

The development of an authentication system for smartphones using fingerphotos and the research of fingerphoto recognition under daily circumstances has very recently raised a lot of attention. Previous work includes [SN-2011], where we have tested five smartphones for their suitability for fingerphoto recognition. Due to the inability of most cameras to focus on the finger, only one of the five evaluated smartphones was able to capture suitable fingerphotos. The work by Derawi et al. [DYB-2011] is concerned with fingerphoto recognition with two different smartphones. But the results are not comparable with this work because the photos were taken only in one session and under different conditions and processing was done offline on a PC. Moreover one smartphone was placed on a fixed hanger to capture the fingerphotos. The achieved EER was 4.66%. The other one was held by a (third) human operator. The result was an EER with 14.65%. Both conditions do not correspond to a realistic scenario.

Fingerphoto recognition with a low resolution camera in a fixed position under laboratory conditions was tested in [HTY-2010]. A continuous shooting mode for the camera was used to capture multiple photos at once from test subjects in one session. A low EER up to 1.23% with preprocessing of the captured photos was achieved under the mentioned circumstances. The work of Müller and Sanchez-Reillo [MS-2009] shows that fingerphoto recognition is even possible with web cams. Web cams without autofocus have the ability to focus on very close objects. A low resolution of 640x480 Pixel of the fingerphotos is sufficient. A False Acceptance Rate (FAR) of 0.18% and a False Rejection Rate (FRR) of 10.29% were achieved.

3 Challenges

From the previous Section it can be concluded that the main problem is that smartphone cameras are not designed for biometric use. Not all cameras are able to focus on the necessary close distance to capture the pattern ridges of the finger and the depth of field is very limited. Without a proper focus on the finger, it is impossible to detect the pattern ridges of the finger. If the finger is too far away from the camera, the effective usable resolution of the fingerphoto is reduced and the risk that the finger cannot be detected increases. Additionally, the low amount of configuration possibilities of the smartphone cameras tightens the conditions for the fingerphoto recognition. Another problem is that the sensors of the cameras are usually small due the compact design of the smartphones. Thus, these cameras tend to produce higher noise having a high impact on the photo quality. It must be also kept in mind that smartphones have limited computing resources. Hence, the computational costs of the processing algorithms must be within the scope.
Besides the challenges of the capture device, the capture process offers further ones. Various potential poses of the finger must be considered: The orientation angle, pitch angle and position of the finger are variable as well as the distance of the finger to the camera and the background (see Figure 1). The fingerphotos are also affected of different light conditions that have impact of the fingerphoto recognition. In addition, the structure and the consistency of the finger, like bulge, peculiarity of the finger ridges, wear and dirt, have also influence of the quality of the fingerphoto recognition.

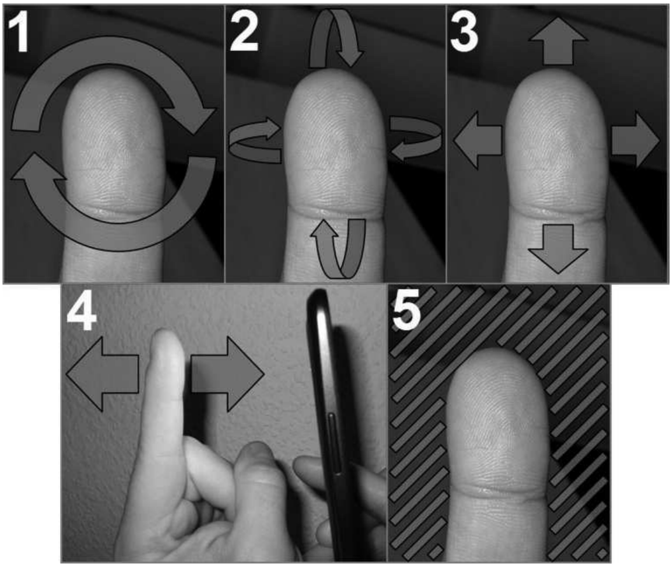

Figure 1: Different orientation angles (1), pitch angles (2) and positions (3) of the finger as well variable distances of the finger to the camera (4) and different backgrounds (5) are challenges of the capture process.

4 Objectives and Approach

Primary objective of this work is to develop a reliable authentication system for smartphones using fingerphotos. The proposed solution is realized as Android application. Algorithms for quality assurance (see Section 6) as well algorithms for finger recognition and finger segmentation (see Section 7) are developed for this purpose. Several preprocessing steps are evaluated to enhance the fingerphotos (see Section 8) before the minutiae are extracted and the template is generated. A template comparator is implemented to determine the number of matching minutiae (see Section 9). The application is integrated as a module for the Modular Biometric Authentication Service System (MBASSy) [WN-2010]. MBASSy is a framework which allows the user to utilize various biometric authentication methods. Finally, the developed application is evaluated regarding recognition rates, performance and usability (see Section 10).

5 Capture Process

The intention of the fingerphoto recognition is that for authentication the user simply positions his finger close in front of the camera in order to capture a biometric sample (see Figure 2). The focus is set to "macro mode", such that the camera uses the closest possible focus. The LED is switched on during the capture process and a two-flash system mode (similar to the "red-eye-reduction mode" on digital cameras) is used. The LED spotlights the finger such that it appears brighter than the background. This simplifies the segmentation of the finger against the background. Another advantage is the reduced camera noise and risk of blurring caused from hand-motion due the high brightness from the LED. The LED also stabilizes the lighting conditions and creates more homogeneous illumination.

The algorithms for finger detection and quality assurance check continuously the preview images of the camera after the capture process has been initiated by the user. The results of the algorithms are calculated in real-time and are displayed on the graphical user interface of the developed prototype. A photo is automatically taken when all criteria (see Section 7) for the fingerphoto recognition are fulfilled.

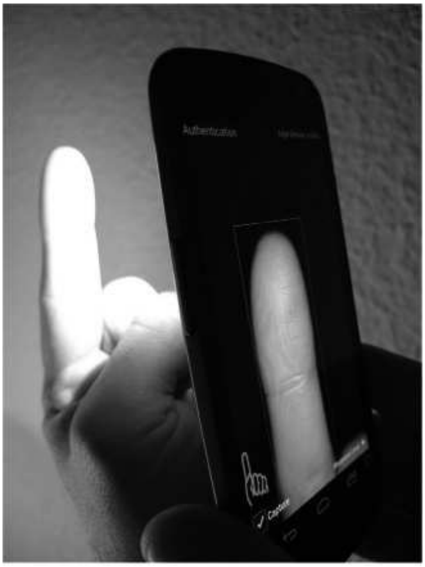

Figure 2: Active capture process with sight on the graphical user interface of the application.

6 Quality Assurance

The most important criterion of the quality of a fingerphoto is the sharpness level, which must be high in order to detect the ridges of the finger. We propose an edge-based approach to determine the sharpness of an image. The Sobel filter is used for this purpose, which calculates the edges (gradient magnitudes) in the image. High-frequent transitions (strong edges) can only appear in sharp images and are clearly visible as bright lines in the edge image (see Figure 3). Blurred images do not contain sharp edges and therefore there are almost no visible lines on the edge image (see Figure 4). An own defined metric, the “edge density”, is used to measure the sharpness in the images. Equation (6.1) provides a suitable metric which is resulting in higher values for sharp images.

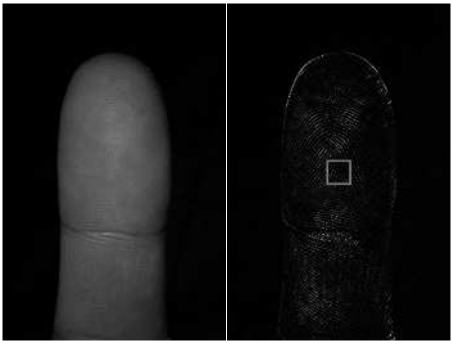

Figure 3: Sharp fingerphoto (left) and the belonging edge image (right); only the few pixel in the center (within the marked rectangle) are analyzed.

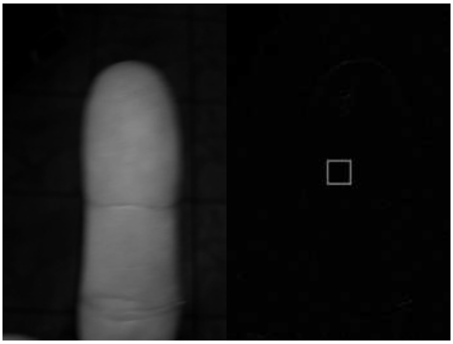

Figure 4: Blurred fingerphoto (left) and the belonging edge image (right); only the few pixel in the center (within the marked rectangle) are analyzed.

E(x,y): gradient-magnitude of the pixel at position (x,y) in the edge image
M: number of columns of the edge image
N: number of rows of the edge image

$$Edge\ density_{Image} = \frac{1}{\mathrm{M} * \mathrm{N}} \sum_{x=1}^{\mathrm{M}=x} \sum_{y=1}^{\mathrm{N}=y} \mathrm{E}(x,y) \qquad (6.1)$$

It is sufficient to calculate the edge density based on a small area of e.g. 50x50 pixels in the center of the photo to reduce the computing time. The luminance of the pixels in the red color channel is compared to a threshold to guarantee that all analyzed pixels belong to the finger. A photo is only taken when this color check is passed and the edge density exceeds an empirically defined threshold. The edge density is calculated on the final fingerphoto again due to the possibility of motion or shakes during the shutter delay of the camera.

7 Finger Recognition

This Section covers the required processing steps to localize the finger in the captured image.

7.1 Finger Detection on the Preview Image

First, the finger must be detected in the preview image. Our algorithm uses the color information of the pixel to determine the finger area. Only the red channel is evaluated to reduce the computational effort. The red channel is most relevant for the finger recognition because the skin-color of the inner finger side is mainly red. The color of the pixel is analyzed starting at the four image borders (left, up, right, down) and continuing towards the center of the image until a certain threshold for the red value is exceeded. When the red value of a pixel exceeds the threshold, then the border for that direction is set. The four set borders determine the Region of Interest (ROI). The steps of this algorithm are illustrated in Figure 5.

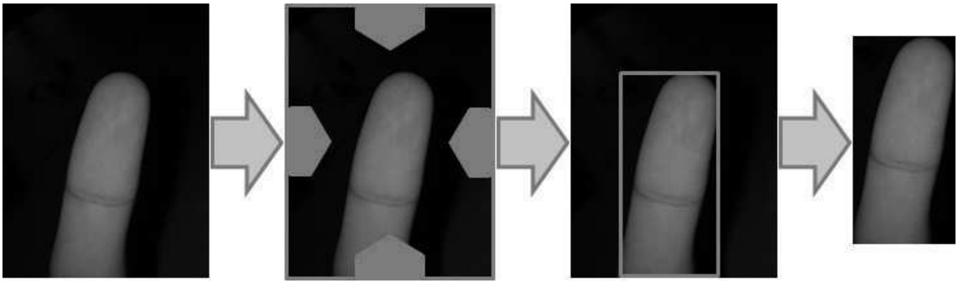

Figure 5: Procedural steps of the algorithm for finger detection on the preview image.

7.2 Rotation of the Fingerphoto

The user is allowed to hold his finger at any desired orientation in the camera. But to compare the features of two fingers, it is necessary that the fingerphotos have a uniform orientation. Therefore, a correction angle must be calculated and applied to the fingerphoto. This is realized with two points P1 and P2 which are set near the center of the image (see Figure 6). The red values of the pixels are analyzed starting from these points and continuing towards the borders until the red value threshold is exceeded. The distances P1x and P2x between the respective start point and the point where the red value exceeds the red threshold are calculated. The correction angle α can be calculated using the distance Py between P1 and P2 and the distances P1x and P2x according to the following formula:

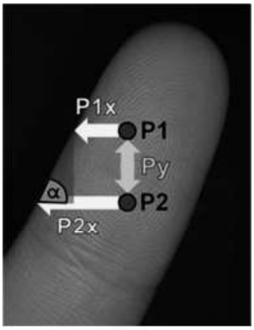

Figure 6: The angle α is calculated by using the distances P1x, P2x and Py.

$$\alpha = atan\left(\frac{Py}{P2x - P1x}\right)$$

7.3 Finger Detection and Segmentation on the Final Fingerphoto

The ROI must be recalculated on the final fingerphoto because the calculated values can differ from the preview image due the shutter lag. The finger recognition works similar to the algorithm on the preview image. The difference is that the startpoint for the checks for a red value of the pixel starts in the center of the image and continues to the borders. This allows detecting the border of the upper finger segment. Hence, the ROI is reduced by the pixels beyond the upper finger segment which do not possess any essential features for the fingerphoto recognition.

The finger is segmented from the background after the ROI is determined. This can be achieved, when all values of the ROI below a defined red value are set to black. All other red values remain unchanged. This results in the segmented finger foreground area.

7.4 Plausibility Checks

The calculated finger foreground area of the fingerphoto undergoes additional plausibility checks. The width and height as well their ratio is checked. The area must exceed a minimum width and a minimum height. Otherwise it can be assumed that an error occurred during finger recognition and the calculated area is too small and does not provide enough information for fingerphoto recognition. Additionally, the height of the area must be greater than the width because a finger is always taller than wide. If this criterion is not met, the calculated correction angle is likely to be wrong and a uniform orientation is not given. The photo is discarded when at least one criterion is not fulfilled and the user is notified about the reason.

8 Preprocessing of the Finger Foreground Area

The finger foreground area is preprocessed after the algorithms of the quality assurance are passed. The preprocessing is necessary for a reliable detection of the minutiae.

A very long finger foreground area indicates that the first finger segment was not properly detected. In this case the lower part of the finger foreground area is removed, such that it does not exceed a defined maximal height. Then the finger foreground area is scaled to a fixed width. The ratio between width and height is kept (by scaling the height with the same scale-factor). This is an important step to compensate the different distances of a finger from the camera resulting in different dimensions of the fingerforeground area. Next, the median filter is applied to reduce the camera noise. At last the finger foreground is binarized with a local binarization filter. The calculation is done by analyzing the red values of the neighborhood pixels of a certain block size to determine the average value. A pixel is set to "white" if this average is above the threshold; otherwise it is set to "black". The output of an optimal binarization is that the ridges of the fingerlines are set to "white" and the valleys are set to "black".

9 Minutia extractor and Template comparator

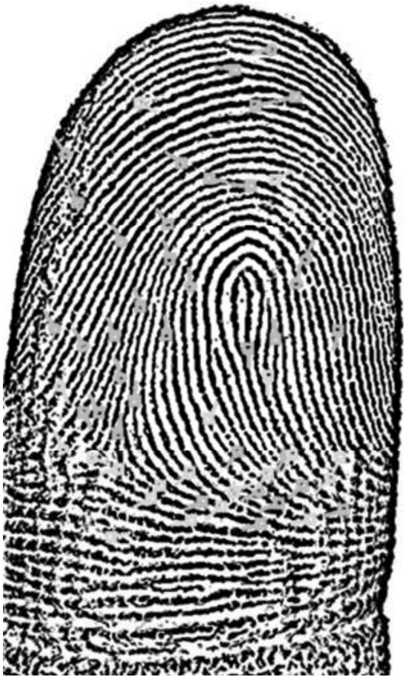

Figure 7: Minutiae extracted from the preprocessed sROI.

The minutiae are extracted from the preprocessed finger foreground area by applying the open source minutia extractor FingerJetFX from DigitalPersona [DP-2012]. The generated templates are stored according to the standardized format ISO/IEC 19794-2. Each extracted minutia that is stored in the template contains the information about position, orientation angle, minutia type and quality score.

These properties can be exploited, when a template is submitted to the comparison subsystem (comparator). But only the two basic minutia types ridge-ending and ridge-bifurcation are detected from the extractor. An examination of the extracted minutiae has shown that the accuracy is not very high. Many false minutiae are detected and often a high error of the calculated orientation angle could be observed after manual inspection (see Figure 7). This deficiency of the extractor is most likely due to the fact that the DigitalPersona algorithm was designed to operate on images from optical or capacitive sensors. Moreover, all extracted minutiae obtain a high to very high quality score making this information useless. The strength of the extractor is the low computational effort allowing a performant execution on smartphones.

A simple comparator optimized for the generated templates from the minutia extractor has been developed in our work. The comparator is based on the common method to find the corresponding minutiae pairs in the reference and probe template. For this purpose, local comparisons are executed for each minutia in the probe template. The more minutiae pairs are found the higher is the similarity score of the templates. A minutiae pair can be found when the following criteria are met: The minutiae type is the same and the Euclidian distance as well as the orientation angle of the two minutiae does not exceed the defined tolerances of the comparator. If several minutiae are found that fulfill these criteria, the one with the lowest Euclidian distance is paired. The tolerance for the Euclidian distance and for the difference of the angle as well the amount of minutiae pairs to be found for positive authentication are parameters of the comparator which have to be configured.

10 Evaluation and results

The described algorithms are implemented in an application for the Android operating system. The application was evaluated in a user test with 41 subjects using the smartphones Samsung "Nexus S" and "Galaxy Nexus". The photos are taken with the highest supported resolution of the camera with five megapixels but were reduced to half of their dimension such that they could be processed despite the limited heap[1]. The user tests were scheduled on two sessions for each subject. The data capture took place on two different days. The enrolment was performed in the first session. Three templates were created from the left and right index finger of each subject. Two photos are captured for one template. The template is generated from the photo with the highest edge density.

The authentication was performed in the second session. Six suitable photos (passing the quality assurance) for fingerphoto recognition were taken from the left and right index finger per person. All captured photos during the tests are stored to create a biometric database for advanced tests with several preprocessing steps on the photos and different parameters for the template comparator to investigate the impact on the recognition rates and to optimize the parameters. In addition, a survey about the usability was performed to get the participant's opinions on the application and the fingerphoto recognition.

10.1 Recognition

On this database different configuration settings are benchmarked according to their Equal Error Rate (EER). The database is divided into training and test set for each device. The data from the two sessions is used separately. Figure 8 shows the Detection trade-off (DET) curves from the "Galaxy Nexus" with different preprocessing steps with a distance tolerance of 50 pixels and an angle tolerance of 45 degree for the comparator. The best EER of 19.1% is achieved with a threshold of 20 identified minutiae pairs for positive authentication. The optimal preprocessing chain on the finger foreground area consists of cutting the height, scaling to a fixed width, applying a median and binarization filter. It should be noted that a reliable extraction of the minutiae on unprocessed fingerphotos was not possible with the extractor from DigitalPersona.

The DET curves in Figure 9 show the impact of different distance tolerances and comparators on the EER. The curves are determined with the photos from the "Galaxy Nexus" and the optimal preprocessing chain. It can be clearly seen that a lower distance tolerance increases the EER. Hence, it is necessary to choose a comparative high tolerance due to the free capture method. An alternative comparator using the Modified Hausdorff-distance (MHD) [DJ-1994] was also evaluated. The achieved EER of 33.7% shows that this technique is not suitable for minutiae comparisons. Adding the criteria of the minutia type to the comparison process, as it is done in our proposed comparator, does not decrease the EER significantly (33.7% to 31.3%). The EERs from the slightly older "Nexus S" are similar to the "Galaxy Nexus". The lowest EER is 22.3% and was achieved under the same conditions.

[1] Amount of access memory that Android assigns an application.

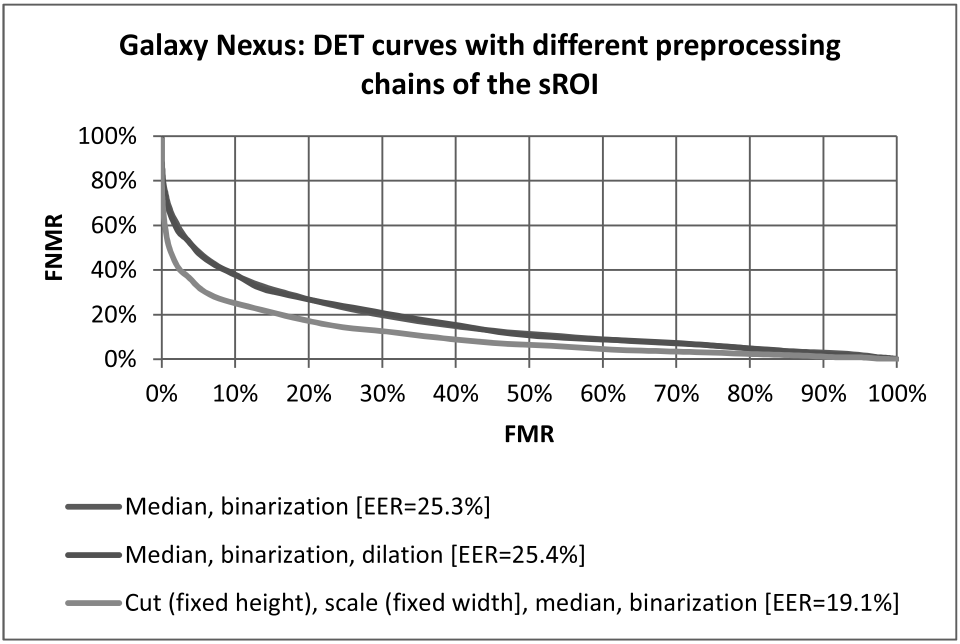

Figure 8: DET curves with different preprocessing chains of the finger foreground area.

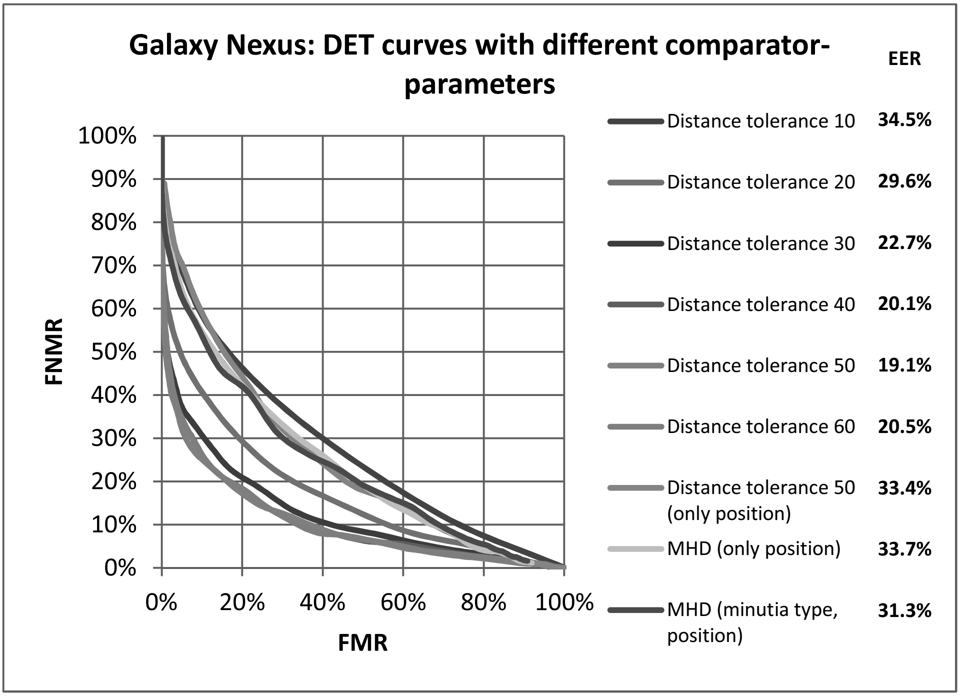

Figure 9: DET curves with different comparator-parameters. In addition the results for the Modified Hausdorff-Distance (MHD) are given.

10.2 Finger detection and Segmentation Errors

In total, 1494 fingerphotos in the user tests on both devices passed the algorithms for quality assurance and were processed with the finger detection and segmentation algorithms. Manual inspection showed that 51 finger foreground areas were not properly calculated. This results in a low error rate of 3%. The errors were either false detected borders of the area, such that it does not contain the whole finger, or a falsely calculated orientation angle that results in an incorrect rotation of the area.

10.3 Duration of the Capture Process and Processing Time

The duration of the capture process and the processing time was measured in the user test. The average duration for both devices of the enrolment process is approximately 50 seconds and for the authentication process 22 seconds (see Figure 10). The reason for the longer duration of the enrolment is that two photos are taken for the template generation. It must be mentioned that all participants were new to the system. They did not have any experience with fingerphoto recognition before and need to learn the handling of such a system. Hence, it can be assumed that the time to complete the capture process gets significantly lower the more experience with the system is made.

The measured average processing time of the photo after it was taken is shown in Figure 11. The total processing time with 1.9 seconds for the "Nexus S" and with 1.5 seconds for the "Galaxy Nexus" is low. The "Galaxy Nexus" is faster due to the faster CPU. The most processing time is used for the preprocessing of the photos.

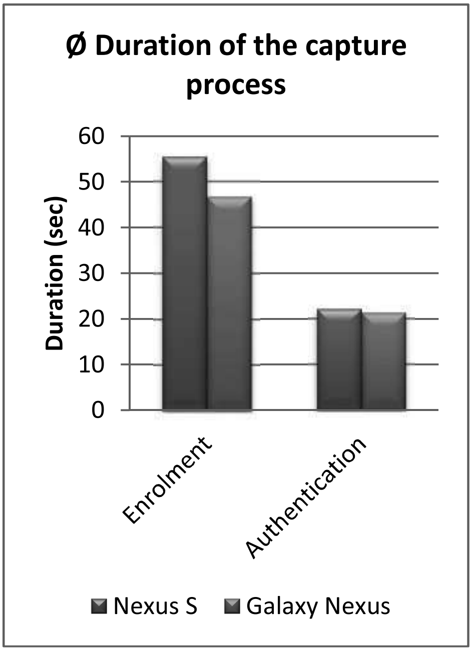

Figure 10: Average duration of the capture process for both test devices

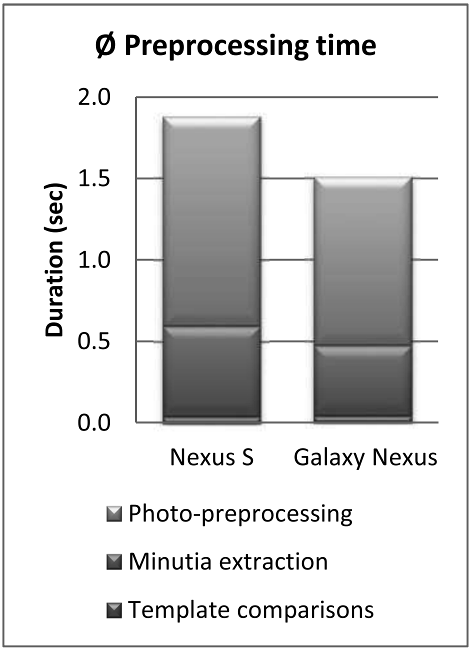

Figure 11: Average processing time for both test devices

11 Conclusions and Future Work

A complete authentication system for a mobile operating system based on fingerphotos was developed and evaluated. It has been shown that fingerphoto recognition on smartphones is possible. A capture process for fingerphotos in combination with efficient postprocessing algorithms was developed. The implemented algorithms provide a very good detection and segmentation of the finger and ensure the suitability of the captured photos for fingerphoto recognition. The capture process allows the user to present the finger to the camera at any desired orientation angle. The EER measured under ordinary circumstances with optimized parameters and preprocessing of the photos is less than 20%.

Because many device dependent variables exist, thresholds for the parameters of the algorithms must be determined empirically for each device. This can be implemented as training-mode for automatic calibration and optimization of the parameters on the device. There is potential to improve the recognition rates by using a different minutia extractor which is optimized for fingerphotos. In the future, more powerful smartphones will be available that can process even more complex algorithms and will allow further techniques for fingerphoto recognition. Also the cameras of the smartphones will be improved in the next generations of smartphones and lead to better recognition rates. Even capturing the fingerprint pattern by using a video could be possible, because more and more smartphone cameras are able to capture high resolution videos. This technique could achieve better usability and spoof-protection because much more captures (with different orientations) can be conducted in a short time frame of a single transaction.

References

[BN-2010] Breitinger, Frank; Nickel, Claudia: *User Survey on Phone Security and Usage.* In: BIOSIG 2010 - Proceedings, BIOSIG, Germany (2010), p. 139-144

[DJ-1994] Dubuisson, Marie-Pierre; Jain, Anil K.: *A Modified Hausdorff Distance for Object Matching*. In: Pattern Recognition - Volume 1 - Conference A: Computer Vision and Image Processing, Proceedings of 12th IAPR, Israel (1994), p. 566-568

[DP-2012] DigitalPersona: *FingerJetFX OSE.* http://www.digitalpersona.com/fingerjetfx/, last visited: 2012-07-10

[DYB-2011] Derawi, Mohammad Omar; Yang, Bian; Busch, Christoph: *Fingerprint Recognition with Embedded Cameras on Mobile Phones.* In: MobiSec 2011, Denmark (2011)

[HTY-2010] Hiew, Bee Yan; Teoh, Andrew Beng Jin; Yin, Ooi Shih: *A Secure Digital Camera Based Fingerprint Verification System.* In: Journal of Visual Communication Image Representation (2010), volume 21, issue 3, p. 219-231

[MS-2009] Mueller, Robert; Sanchez-Reillo, Raul: *An Approach to Biometric Identity Management using Low-Cost Equipment.* In: IIH-MSP 2009, Japan (2009), p. 1096-1100

[SN-2011] Stein, Chris; Nickel, Claudia: *Eignung von Smartphone-Kameras zur Fingerabdruckerkennung und Methoden zur Verbesserung der Qualität der Fingerbilder.* In: BIOSIG 2011 - Proceedings, BIOSIG, Germany (2011), p. 297-304

[WN-2010] Witte, Heiko; Nickel, Claudia: *Modular Biometric Authentication Service System (MBASSy).* In: BIOSIG 2010 - Proceedings, BIOSIG, Germany (2010), p. 115-120