Towards a Standardised Testsuite to Assess Fingerprint Matching Robustness: The StirMark Toolkit – Cross-Feature Type Comparisons

Jutta Hämmerle-Uhl, Michael Pober, Andreas Uhl

Towards a Standardised Testsuite to Assess Fingerprint Matching Robustness: The StirMark Toolkit – Cross-Feature Type Comparisons

Jutta Hämmerle-Uhl, Michael Pober, and Andreas Uhl

Multimedia Signal Processing and Security Lab (WaveLab)
Department of Computer Sciences, University of Salzburg
andreas.uhl@sbg.ac.at

Abstract. We propose to establish a standardised tool in fingerprint recognition robustness assessment, which is able to simulate a wide class of acquisition conditions, applicable to any given dataset and also of potential interest in forensic analysis. As an example, StirMark image manipulations (as being developed in the context of watermarking robustness assessment) are applied to fingerprint data to generate test data for robustness evaluations, thereby interpreting certain image manipulations as being highly related to realistic fingerprint acquisition conditions. Experimental results involving three types of fingerprint features and matching schemes (i.e. correlation-based, ridge feature-based, and minutiae-based) applied to FVC2004 data underline the need for standardised testing and a corresponding simulation toolset.

1 Introduction

One of the big issues in fingerprint recognition is robustness of recognition accuracy against sample image quality degradation [1, 2]. The performance of a fingerprint recognition system is usually heavily affected by fingerprint image quality. A wide variety of factors influence the quality of a fingerprint image: Skin conditions (*e.g.* , dryness, moisture, dirt, cuts and bruises), sensor conditions (*e.g.* , dirt, noise, size), and other acquisition conditions like user cooperation or crime scene preservation in forensic settings, etc. Some of these factors are inevitable and some of them change over time. Poor quality images often result in spurious and missed features, therefore decreasing the recognition accuracy of the overall system.

However, the different levels at which fingerprint features are extracted [2] and the different feature types extracted at these levels influence the impact of quality degradations on recognition performance in various ways. Moreover, there is interplay among different types of feature extraction and acquisition technology / conditions such that it is not clear a priori which type of feature extraction is favourable under which conditions. Therefore, it is essential to provide reliable methodology to comparatively assess fingerprint recognition robustness under varying conditions.

This issue is classically tackled from two sides: First, benchmarking frameworks have been established, which facilitate a common evaluation basis with standardised

protocols for various fingerprint recognition algorithms, see *e.g.* the fingerprint verification contests (FVC [2]), independent suggestions like [3], and the BioSecure evaluation framework [1]. Second, usually these frameworks rely on the establishment of test data which are used to compare the different algorithms on a common basis. A very good example, specifically focusing onto the robustness issue, are the FVC data sets. FVC2002 (only i) & iv)) and FVC2004 data have been acquired in a way to introduce higher intraclass variation by i) putting the finger at slightly different vertical position, ii) applying low or high pressure against the sensor, iii) exaggerating skin distortion and rotation, and iv) drying or moistening fingers. For FVC2006, the population was chosen to be more heterogeneous, including manual workers and elderly people.

While the availability of these and similar datasets is a significant achievement, the data collection and database establishment is tedious work. Moreover, if additional acquisition conditions should be considered which have not been included into the original dataset, re-enrolment is required, involving complicated procedures for getting the original people back to enrolment. Also, it is hard to compare the different quality degradations from dataset to dataset (*e.g.* FVC, MCYT, BIOMET, MSU), since usually, there is no standardised manner to generate the acquisition conditions applied. Therefore, the experimental results of recognition algorithms in case applied to different datasets are hardly comparable and the results shown in many papers are difficult to interpret.

A strategy to cope with the various problems of generating natural datasets is to generate synthetic fingerprints, the SFinGe [4] being the most well known tool for doing this. The generated fingerprints have proven to be highly realistic and serve as a sensible tool to generate large datasets for benchmarking. While SFinGe also allows to apply some manipulations to the images, *e.g.* noise insertion, translations, rotations and uses a skin deformation model, a simulation of specific sensor types is not foreseen.

In the area of robust watermarking, a similar situation could be observed – while of course the notion of robustness is different in watermarking (means basically the ability of embedded data to withstand common image manipulations or unspecific attacks), the general problem was of comparable nature: Each watermarking scheme presented was evaluated on a specific dataset, where especially the types of introduced image manipulations and their respective extent to prove robustness varied from paper to paper, thus making a comparison of techniques impossible. To cope with the situation, standardised benchmark toolsets consisting of a collection of parameterisable image manipulations have been created, including StirMark [5] and CheckMark [6]. This enabled developers and authors to apply these manipulations to publicly available datasets thus making their results comparable.

In recent work [7], we have proposed to establish a standardised tool in fingerprint recognition robustness assessment, which is able to simulate a wide class of acquisition conditions, applicable to any given dataset. As an example, StirMark image manipulations have been applied to fingerprint data to generate test data for robustness evaluations. Since these manipulations can be applied to any dataset, the effect of manipulations on data originating from different sensors and acquisition conditions can be studied with respect to recognition accuracies of the algorithms used. Contrasting to previous work [7], where experiments have been restricted to fingerprint matchers of minutiae type, here we focus on fingerprint matchers relying on very different fea-

ture types and compare the obtained results. Additionally, different distortion types are investigated as compared to [7].

In Section 2, we explain the StirMark image manipulations and discuss the interpretation of those procedures in the context of fingerprint acquisition and quality, respectively. Section 3 briefly reviews the fundamental ideas behind three very different types of fingerprint feature extraction and matching techniques which are subsequently used in experiments. Experimental results are presented in Section 4 where we shortly describe the employed FVC2004 dataset and experimental conditions with respect to evaluation protocols. Finally, we present fingerprint verification results generated on the FVC2004 dataset processed with a set of StirMark image manipulations with increasing strength. Section 5 concludes the paper.

2 The StirMark Toolkit

The StirMark Benchmark is a generic benchmark test for evaluating the robustness of digital image watermarking methods, developed by Fabien A. P. Petitcolas *et al.* [5, 8]. The basic idea behind the robustness tests in the StirMark benchmark is, that a digital watermark within an image can be attacked and possibly rendered useless, by introducing small, ideally imperceptible perturbations into the marked image. To be suitable for application in a common generic benchmark, the specific types of perturbations are pre-defined and the respective intensity is adjustable via a given set of parameters. The corresponding software is currently available "StirMark Benchmark 4.0" at `http://www.petitcolas.net/fabien/watermarking/stirmark/`. A related, also watermarking-robustness focused toolset is CheckMark [6] which could be used by analogy, however, it is less well supported.

In the following, we describe the set of StirMark image manipulations that has been selected for this study. We explain the way each manipulation is defined, how it is parameterised to achieve varying strength of the manipulation, and we discuss which realistic fingerprint acquisition condition could be modelled by applying the manipulation to fingerprint sample images. Thus, only a subset of the complete range of StirMark tests is used, which simulate "natural" perturbations – in other words, tests, whose influence on fingerprint images creates perturbed versions thereof, that resemble cases appearing in real-life fingerprint application scenarios. It has to be noted that we do not consider all manipulations even if they would be suitable candidates – *e.g.* , JPEG compression, although contained in the StirMark suite, is not applied here since there have been quite some studies focusing on the effects of JPEG compression in fingerprint recognition [9, 10]. Example images shown have been generated by applying StirMark tests with increasing intensity to a sample image taken from the FVC2004 database DB1 (see Section 4.1).

Additive Noise is introduced to the input image. The amount of noise is adjustable and can range from "none" to "completely random image", controlled by a single parameter, ranging from 0 to 100. Fig. 1 shows examples for increasing noise content.

This test is intended to simulate noise, that might "naturally" appear in fingerprint sample images. Possible causes for this kind of noise could be actual dust on the contact area during acquisition of the imprint, graining caused by the acquisition equipment

(a) Noise level 3 (b) Noise level 9 (c) Noise level 15

Fig. 1. Examples for the *Additive Noise* test, applied to an image from DB1 (ID 91_2).

itself (sensor noise) or any other kind of systematic error introduced during processing, transmission and/or storage of the collected images (e.g. a grainy surface the latent fingerprint has been taken off can cause noise in forensics).

Median Cut Filtering This test applies a median cut filter to the input image. The size of the filter mask can be set (height and width of the filter take the same value and only odd-valued dimensions are accepted), the upper limit is a size of 15, thus resulting in a 15×15 filter. Fig. 2 shows examples for medium filter sizes.

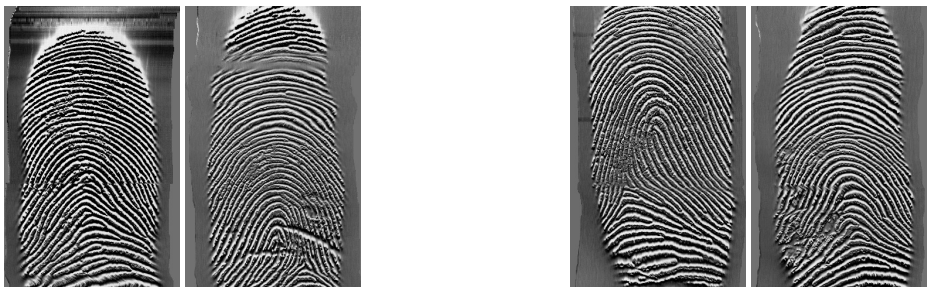
(a) Filter size 3 (b) Filter size 7 (c) Filter size 9

Fig. 2. Examples for the *Median Cut Filtering* test, applied to an image from DB1 (ID 91_2).

The *Median Cut Filtering* test is used to simulate smudgy fingerprints, as they are common in real-life applications, for example when the fingertip is too moist during the acquisition by the scanner. The result is a certain amount of blur to the image, but additionally it also corrupts the clarity of the ridge-and-furrow structure of the imprint.

Remove Lines and Columns This tests removes rows and columns from a given image at the specified frequency k – *"remove 1 line in every k lines."* It has to be noted that the line removal operation naturally also reduces the size of the output image. Fig. 4 illustrates the effect of this test when applied to fingerprint images.

This test aims to simulate errors in fingerprint images, that occasionally occur during fingerprint acquisitions, in case the scanner is not able to read the fingerprint in its entirety, but misses/skips certain lines. Especially sweep sensors are prone to this kind of complications. Two corresponding examples can be found in Fig. 3.a.



(a) Missing lines (b) Warping effects

Fig. 3. Examples for distortions from actual acquisition problems.

(a) Remove 1/100 lines (b) Remove 1/50 lines (c) Remove 1/10 lines

Fig. 4. Examples for the *Remove Lines* test, applied to an image from DB1 (ID 91_2).

Rotation rotates the image by a given angle, the set of angular values that will be inspected in the experiments is $\{-20°, -15°, -10°, -5.5°, -5°, 7°, 7.5°, 13°, 18°, 20°\}$. Examples for rotations of $-15°$, $-5.5°$, and $20°$ can be seen in Fig. 5.

(a) Rot. of $-15°$ (b) Rot. of $-5.5°$ (c) Rot. of $20°$

Fig. 5. Examples for the *Rotations Lines* test, applied to an image from DB1 (ID 91_2).

Rotation is a very typical, not to say – omnipresent – challenge for fingerprint matching, as in very few cases a finger will be presented twice in exactly the same

orientation to the contact area during image acquisition. Thus, this test provides the means for comparison of the rotational alignment capabilities of the various fingerprint matchers.

Affine Transformation is a generic manipulation for arbitrary affine image transformations. The user specifies the parameters $a, \ldots, f$ of the inverse transformation matrix of the form:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a\ b \\ c\ d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix}$$

The application of affine transformations to fingerprint images is intended to simulate distortions of the entire finger imprint, that can appear in real-life situations during the fingerprint acquisition, depending on the way, the finger is pressed on the contact area. As special cases, we consider *shearing* and *stretching*.

Stretching in X-direction is parameterised by setting $b = c = e = f = 0$ and $d = 1$, while configurations 1 - 8 set a to the values $\{1.035, 1.070, 1.105, 1.140, 1.175, 1.210, 1.280, 1.350\}$. Configurations 1, 5, and 8 are shown in Fig. 6.

(a) $a = 1.035$ (b) $a = 1.175$ (c) $a = 1.350$

Fig. 6. Examples for the *Affine Transformations – Stretching in X-Direction* test, applied to an image from DB1 (ID 91_2).

A certain *stretching* might appear in the finger imprint, when the amount of force applied while pressing the finger on the contact area is large or larger than usual. Considering the forensic scenario, stretching of a fingerprint appears if the finger was imprinted on a soft or flexible surface.

Shearing in Y-direction is parameterised by setting $b = e = f = 0$ and $a = d = 1$, while configurations 1 - 6 set c to the values $\{0.05, 0.10, 0.15, 0.20, 0.25, 0.30\}$. Configurations 1, 4, and 6 are shown in Fig. 7. A *shearing* effect can occur, when the force that is exercised while pressing the finger on the contact area is not exerted perpendicular to this area. For example, when the finger is presented, with the user pushing rather in direction to the upper-right corner of the sensor, than straight downwards.

Small Random Distortions *The* StirMark test. Being a combination of several basic manipulations (i.e. random minor geometric distortion followed by resampling and interpolation, a transfer function to emulate analog/digital converter imperfections, global "bending", high frequency displacement, and JPEG compression), this test originally aims to simulate a resampling process, i.e. the errors introduced when printing an image

(a) $c = 0.05$ (b) $c = 0.20$ (c) $c = 0.30$

Fig. 7. Examples for the *Affine Transformations – Shearing in Y-Direction* test, applied to an image from DB1 (ID 91_2).

and then scanning it again. This test is executed with parameter values $\{0.6, 1.0, 1.4, 1.8, 2.2, 2.6, 3.0, 3.4, 3.8, 4.2\}$, three out of which are illustrated in Fig. 8. The involved image warping is performed both on a global, as well as on a very local level, adding even more to the "natural" and "coincidental" character of the output fingerprint images.

In its character of being a combination of several different image distortions, by applying *the* StirMark test on fingerprint images, we aim to simulate an interaction of various naturally occurring image perturbations: Foremost a random warping of the ridge lines, that in real life would be caused by *e.g.* unevenly distributed pressure exercised on the contact area during acquisition, or if this contact area were to be uneven by itself. Also inaccuracies or errors introduced by the fingerprint scanner can be a source for this type of deformation (two corresponding examples are shown in Fig. 3.b).

(a) rnddist 0.6 (b) rnddist 2.6 (c) rnddist 4.2

Fig. 8. Examples for the *Small Random Distortions* test, applied to an image from DB1 (ID 91_2).

3 Fingerprint Feature Types and Matching

An interesting question is to ask if a certain type of fingerprint feature extraction method has particular strong points or weaknesses when dealing with a specific type of acquisition condition. In order to get a sensible answer we will consider significantly different

types of fingerprint feature extraction schemes, based on the discriminative characteristics fingerprint do contain [2]. On a global level, the overall ridge flow structure with the embedded singular points (i.e. loops, whorls, deltas) can be perceived. Going into more detail, looking at the ridge and furrow structure in the fingerprint from a more localised point of view, then characteristics like orientation and frequency of the local ridge flow can be observed. On a local level, details of the ridge shapes themselves can be determined. The most important ones are *ridge endings* and *ridge bifurcations* which are subsumed in the term *minutiae*. Looking even closer at a fingerprint, then also diminutive intra-ridge features are detectable – the finger's sweat pores. However, their respective pattern can only be determined in sufficiently good quality in high-resolution fingerprint images of 1000dpi and above.

Now depending on which type of features is used to determine similarity, fingerprint matching approaches can be classically categorised into one of the following classes:

Correlation-Based Matcher These approaches use the fingerprint images in their entirety, the global ridge and furrow structure of a fingerprint is decisive. Images are correlated at different rotational and translational alignments, image transform techniques may be utilised for that purpose.

Ridge Feature-Based Matcher Matching algorithms in this category likewise deal with the overall ridge and furrow structure in the fingerprint, yet in a localised manner. Characteristics like local ridge orientation or local ridge frequency are used to generate a set of appropriate features representing the individual fingerprint.

Minutiae-Based Matcher The set of minutiae within each fingerprint is determined and stored as list, each minutia being represented (at least) by its location and direction. The matching process then basically tries to establish an optimal alignment between the minutiae sets of two fingerprints to be matched, resulting in a maximum number of pairings between minutiae from one set with compatible ones from the other set.

The minutiae-based approach is the most common and most widely used method for fingerprint matching. Of course, there also exist fingerprint matching algorithms, that combine some or all of the above listed techniques (termed *hybrid*), aiming to combine the particular strong points of each individual approach into a single, more precise matcher (*e.g.* by using minutiae alignment to compensate for rotation and translation but use ridge-based features for actual matching [11]).

As a representative of the correlation-based matcher type we use a custom implementation of the phase only correlation (POC) matcher [12]. The POC of two images is computed by calculating the normalised cross spectrum (or cross-phase spectrum) from the DFT of the two images. The POC is then obtained by taking the inverse DFT of the normalised cross spectrum. Properties like shift invariance, brightness invariance, and high immunity to noise make POC an interesting candidate for biometric matching since template alignment is eased.

The algorithm first conducts rotational alignment by computing POC for rotated fingerprints in a range of $\pm 20^\circ$ with a step-width of 1°, employing bi-cubic interpolation. The rotated version with the highest POC response is used in subsequent matching. Displacement alignment is performed according to the position of the POC peak as

computed before, subsequently probe and gallery images are cropped to the common area of intersection (containing fingerprint data), as the non-overlapping regions lead to uncorrelated noise in the POC function. Finally, a band-limited version of the POC is computed. The idea is to limit the frequency spectrum involved in matching to only those areas, that are strongly related with the actual fingerprint information – especially the inherent elliptical frequency band originating from the ridge pattern – thereby excluding the interfering components in the high frequency areas. The final matching score is then established by summing up the P highest peaks (Ito *et al.* suggest $P = 2$, while we found $P = 1$ to perform better for two out of three databases) of the band-limited POC function. It has to be noted that fingerprint enhancement as used in the subsequent algorithm [13] also improves matching results for two out of three databases and is used correspondingly.

As a representative of the ridge feature-based matcher type we use a custom implementation of the fingercode approach (FC) [14], in particular we follow improvements as suggested by Ross *et al.* [15, 11] and de Sa *et al.* [16] which avoid the usage of a circular tessellation around a core point, and partially apply the fingerprint enhancement strategy as suggested by Hong *et al.* [13].

As a first stage, normalisation is applied by pixel-wise adjusting the gray-levels to obtain an image with pre-specified mean and variance. Based on the output of a Marr-Hildreth operator, a least square estimate of the local ridge orientation in blocks of 16×16 pixels is established subsequently. A low-pass filter is used to smoothen the result which is called orientation image. The normalised fingerprint image and the orientation image are used to create the frequency image, representing the local ridge frequency. In an oriented window the *x-signature* is calculated per block by projecting the respective gray-level values of the normalised image onto the length of the window which is placed in a direction orthogonal to the ridge orientation of the block. The frequency can be determined by taking the reciprocal of the average distance between peaks in the x-signature. Interpolation of invalid blocks (i.e. those where the x-signature did not form a discrete sinusoidal-shape wave) is done with a discrete Gaussian kernel. Contrasting to the original papers, we use high frequency together with a check of an admissible dynamic range as criteria to determine blocks which actually represent useful fingerprint texture, thus declared foreground blocks to be further used in matching.

For actual feature extraction, a Gabor filter bank consisting of eight separate Gabor filters, each oriented at a different constant angle is convolved with the image, examining the varying responses of the ridges and furrow structure to the differently oriented filters. This results in eight distinct filtered images for each of which a standard deviation in a 16×16 neighbourhood is computed per pixel, the union over all eight images is called *Standard Deviation Map*. If the fingerprint image is intended for database registration (i.e. enrolment), this map is subsampled by a factor of 16 to generate ridge feature images, the union of which is called *Ridge Feature Map*. Translational alignment is achieved by computing correlations among differently displaced ridge feature images in the Fourier domain and compensating the shift identified by maximal correlation, rotational alignment is done by storing ridge feature maps of rotated fingerprint versions in an angular range of $[-20°, +20°]$ with a step-width of $1°$, and again taking

the version with the maximal correlation. The matching score is obtained by computing Euclidian distance among aligned ridge feature map entries.

As a representative of the minutiae-based matcher type we use *mindtct* and *bozorth3* from the "NIST Biometric Image Software" (NBIS) package (available at `http://fingerprint.nist.gov/NBIS/`) for minutiae detection and matching, respectively. *mindtct* generates several image quality maps and binarises the fingerprint images as a first step. Subsequently, minutiae are detected in admissible areas by detecting specified pixel patterns, followed by false minutiae removal and minutiae quality assessment. *bozorth3* is designed to be rotation and translation invariant and provides a matching score based on traversing certain inter-fingerprint compatibility tables.

A comparison of the three fingerprint recognition schemes with respect to recognition performance on the three "natural" FVC2004 databases (without StirMark manipulations being applied) is provided in the subsequent subsection.

4 Experiments

We first provide details about the employed FVC2004 data set. Subsequently, experimental results are presented and discussed, covering questions of robustness of recognition accuracy against various StirMark manipulations and in particular a comparison of the behaviour of the different feature types in that respect.

4.1 Experimental Settings

We employ three out of four databases provided for the FVC2004 [17] as shown in Table 1, each with 500dpi resolution (DB3 with 512 dpi).

Table 1. Details on the fingerprint images in the three employed FVC2004 databases and EERs for the considered fingerprint recognition schemes when applied to the original, "undistorted" sample image databases.

	Sensor Type	Model	Image Size	NBIS (%)	FC (%)	POC (%)
DB1	Optical	CrossMatch *V300*	640×480	14.81	12.54	22.60
DB2	Optical	Digital Persona *U.are.U 400*	328×364	11.12	9.60	9.69
DB3	Thermal Sweep	Atmel *FingerChip*	300×480	6.68	8.98	15.07

It does not make too much sense to include the fourth dataset of FVC2004 (although it would be possible in principle of course) since it consists of synthetically generated fingerprint images (SFinGe [4] was used). In order to model real-life distortions for these data, a much more sensible way would be to apply respective distortion "operations" already during the generation process of the synthetic fingerprints instead of applying them ex post to the final data.

The procedure for performance evaluation is basically the same in all FVCs, from 2000 to 2006. We follow this specification by conducting all genuine tests and the required impostor tests for DB1, DB2, and DB3, thereby obtaining FNMR and FMR as

required. Finally, equal error rate (EER) is determined and used as measure for recognition accuracy to compare different settings. Table 1 also shows the result when applying the three considered feature types to the FVC2004 test data *without* having applied any StirMark manipulations, but already within the StirMark framework.

It can be clearly seen that the ranking of the three feature types is heavily dependent on the used dataset. Each type of feature extraction is ranked first for a single dataset, while only FC is never ranked third. Only when considering DB2, the performance is really close.

4.2 Experimental Results

Fig. 9.a shows the influence of additive noise on recognition performance considering DB2. Especially for a higher degree of noise content FC clearly shows the best robustness, POC is also better compared to NBIS for this setting, but clearly inferior to FC.

Noise Level	NBIS (%)	FC (%)	POC (%)
unperturbed	11.12	9.60	9.69
03	10.86	11.85	10.65
07	15.03	14.22	14.36
11	20.54	17.74	20.22
15	30.78	21.80	26.94

(a) DB2

Noise Level	NBIS (%)	FC (%)	POC (%)
unperturbed	6.68	8.98	15.07
03	7.05	9.25	15.28
07	7.19	10.50	15.16
11	7.08	14.79	15.71
15	7.91	24.99	17.46

(b) DB3

Fig. 9. EERs for *Additive Noise* test

In Fig. 9.b we see that the situation changes when considering a different dataset, DB3 in this example. NBIS recognition results are hardly affected even by a significant amount of noise, entirely contrasting to the results obtained on DB2. Also, the ranking between FC and POC is swapped, POC results are also quite stable while FC recognition accuracy severely suffers from high noise content.

It is also interesting to note that on unperturbed data, FC is superior to POC, while POC is getting clearly superior under the influence of more noise being present. This effect underlines the need for systematic robustness testing in a feature-type comparative manner.

This example overall shows that even feature-type ranking results with respect to robustness achieved on a specific database cannot be generalised but need to be verified for each single dataset. This nicely illustrates the general need for systematic testing and evaluation tools.

Fig. 10.a shows robustness results with respect to median cut filtering on DB1, which is another example that ROC performance on unperturbed data cannot predict robustness properties. While NBIS is clearly superior to POC on the original data, it gets inferior when introducing significant mean cut filtering.

Filter Size	NBIS (%)	FC (%)	POC (%)
unperturbed	14.81	12.54	22.60
03	15.50	12.90	23.63
05	17.69	13.52	24.92
07	32.17	16.55	30.71
09	46.88	28.26	38.11

(a) *Median Cut Filtering*

k	NBIS (%)	FC (%)	POC (%)
unperturbed	11.12	9.60	9.69
90	11.04	9.71	10.00
70	11.60	9.73	10.24
40	11.99	9.47	11.18
20	12.92	9.97	14.75

(b) *Remove Lines*

Fig. 10. EERs for robustness tests conducted on sample image databases DB1 and DB2, respectively.

In Fig. 10.b very high stability of NBIS and FC against line removal is shown, while POC suffers considerably in case the amount of missing lines is increasing.

One of the most important robustness issues is fingerprint rotation, since this effect is omnipresent in sample acquisition. Fig. 11 compares the results for DB1 and DB2.

Rotation	NBIS (%)	FC (%)	POC (%)
unperturbed	14.81	12.54	22.60
-15	13.00	14.74	24.34
-5.5	12.94	12.90	22.67
13	13.05	13.63	23.79
20	13.41	15.44	26.18

(a) DB1

Rotation	NBIS (%)	FC (%)	POC (%)
unperturbed	11.12	9.60	9.69
-15	11.00	14.13	14.22
-5.5	11.28	12.04	10.89
13	10.59	13.57	13.01
20	10.94	16.27	18.08

(b) DB2

Fig. 11. EERs for *Rotation* test conducted on sample image databases.

The first thing to note is the excellent robustness of NBIS against rotation for both datasets. FC and POC are both affected, however, the extent of result degradation is much larger for DB2 as compared to DB1. For example, based on experimental results obtained on DB1, one would have predicted EER$\approx$12.5 for POC on DB2 under 20° rotation, in fact we observe EER to be 18.08 which almost doubles EER as compared to not manipulated data. Again, the need for dedicated testing for each sensor type is confirmed.

Affine transformations also model a class of very important acquisition conditions. Table 2 shows the catastrophic effect of stretching in a single dimension only. No feature extraction type can handle this type of distortion in a sufficient degree. Obviously, there is need to introduce stretching robustness into feature sets.

Shearing robustness as illustrated in Fig. 12.a is shown to be much better as compared to stretching. Apart from very strong distortions, NBIS and FC can handle shear-

Table 2. EERs for *Affine Transformations – Stretching in X-Direction* test conducted on sample image database DB3.

Configuration	NBIS (%)	FC (%)	POC (%)
unperturbed	6.68	8.98	15.07
2	7.70	10.99	23.89
4	11.13	13.90	31.72
6	14.14	17.98	37.20
8	23.38	25.79	42.69

ing quite well, while POC exhibits steadily decreasing EERs for an increasing amount of shearing.

Configuration	NBIS (%)	FC (%)	POC (%)
unperturbed	14.81	12.54	22.60
1	13.85	12.57	22.64
2	13.88	12.76	24.79
3	14.88	13.30	27.43
4	16.26	14.15	29.90
5	17.96	14.71	37.73
6	21.46	15.82	40.22

(a) *Shearing in Y-Direction*

Factor	NBIS (%)	FC (%)	POC (%)
unperturbed	11.12	9.60	9.69
0.6	10.78	12.42	10.89
1.0	11.35	12.34	11.49
1.8	11.75	12.40	13.23
2.6	12.57	12.61	16.34
3.4	13.23	13.57	19.00
4.2	14.82	14.05	21.96

(b) *Small Random Distortions*

Fig. 12. EERs for robustness tests conducted on sample image databases DB1 and DB2, respectively.

Finally, robustness results against a combination of manipulations are shown in Fig. 12.b. These rather localised distortions can be handled quite well by NBIS and FC, at least up to some medium extent. POC, similar to its sensitivity against affine transformations, exhibits steadily increasing EERs for increasing strength of the distortions.

5 Conclusion

We have employed the StirMark benchmark testsuite to generate large scale test data to assess robustness of fingerprint recognition schemes in various acquisition conditions. Experimental results confirm a significant variability of robustness properties across different types of fingerprint feature extraction schemes **and** across different datasets considered. For example, we have observed significant impact of Median Cut Filtering and Affine transforms like Stretching and Shearing for almost all feature types and

datasets, while Noise Insertion is tolerated well by some feature types (FC and POC on DB1 and NBIS and POC on DB3) but leads to considerable impact for all techniques on DB2. As compared to our previous work [7] where only minutiae-based matching schemes have been compared, we see even larger variability in this present work when comparing matching results relying on entirely different feature extraction schemes.

These results underline the need for a standardised tool in fingerprint recognition robustness assessment, which is able to simulate a wide class of acquisition conditions, applicable to any given dataset.

While we have motivated the interpretation of several image manipulations contained in the StirMark benchmark as being closely related to a wide class of fingerprint acquisition conditions (including some forensic settings), these experiments only represent a first step. In fact, the aim is to establish a benchmark explicitly designed for systematic fingerprint recognition robustness evaluations, where these current StirMark based results can serve as first guidelines to model actual fingerprint acquisition conditions more accurately.

References

[1] Alonso-Fernandes, F., Bigun, J., Fierrez, J., Fronthaler, H., Kollreider, K., Ortega-Garcia, J.: Fingerprint recognition. In Petrovska-Delacretaz, D., Chollet, G., Dorizzi, B., eds.: Guide to Biometric Reference Systems and Performance Evaluation. Springer-Verlag (2009) 51–88

[2] Maltoni, D., Maio, D., Jain, A., Prabhakar, S.: Handbook of Fingerprint Recognition (2nd Edition). Springer-Verlag (2009)

[3] Noviyanto, A., Pulungan, R.: A comparison framework for fingerprint recognition methods. In: Proceedings of the 6th SEAMS-UGM Conference (Computer, Graph and Combinatorics). (2011) 601–614

[4] Cappelli, R.: Synthetic fingerprint generation. In Maltoni, D., Maio, D., Jain, A., Prabhakar, S., eds.: Handbook of Fingerprint Recognition (2nd Edition). Springer-Verlag (2009) 271–302

[5] Kutter, M., Petitcolas, F.A.P.: Fair evaluation methods for image watermarking systems. Journal of Electronic Imaging **9**(4) (October 2000) 445–455

[6] Meerwald, P., Pereira, S.: Attacks, applications and evaluation of known watermarking algorithms with Checkmark. In Wong, P.W., Delp, E.J., eds.: Proceedings of SPIE, Electronic Imaging, Security and Watermarking of Multimedia Contents IV. Volume 4675., San Jose, CA, USA, SPIE (January 2002) 293–304

[7] Hämmerle-Uhl, J., Pober, M., Uhl, A.: Towards standardised fingerprint matching robustness assessment: The stirmark toolkit – cross-database comparisons with minutiae-based matching. In: Proceedings of the 1st ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec'13), Montpellier, France (June 2013) To appear.

[8] Petitcolas, F.A.P., Anderson, R.J., Kuhn, M.G.: Attacks on copyright marking systems. In Aucsmith, D., ed.: Information Hiding: Second International Workshop. Volume 1525 of Lecture Notes in Computer Science., Portland, OR, USA, Springer Verlag, Berlin, Germany (April 1998) 218–238

[9] Funk, W., Arnold, M., Busch, C., Munde, A.: Evaluation of image compression algorithms for fingerprint and face recognition systems. In Cole, J., Wolthusen, S., eds.: Proceedings from the Sixth Annual IEEE Systems, Man and Cybernetics (SMC) Information Assurance Workshop, IEEE Computer Society (June 2006) 72–78

[10] Mascher-Kampfer, A., Stögner, H., Uhl, A.: Comparison of compression algorithms' impact on fingerprint and face recognition accuracy. In Chen, C., Schonfeld, D., Luo, J., eds.: Visual Communications and Image Processing 2007 (VCIP'07). Number 6508 in Proceedings of SPIE, San Jose, CA, USA, SPIE (January 2007) 650810–1 – 65050N–10

[11] Ross, A., Jain, A.K., Reisman, J.: A hybrid fingerprint matcher. Pattern Recognition **36**(7) (2003) 1661–1673

[12] Koichi, I., Hiroshi, N., Koji, K., Takafumi, A., Tatsuo, H.: A fingerprint matching algorithm using phase-only correlation. IEICE Transactions on Fundamentals **E87-A**(3) (March 2004) 682–691

[13] Hong, L., Wan, Y., Jain, A.: Fingerprint image enhancement: Algorithm and performance evalution. IEEE Transactions on Pattern Analysis and Machine Intelligence **20**(8) (August 1998) 777–789

[14] Jain, A.K., Prabhakar, S., Hong, L., Pankanti, S.: Filterbank-based Fingerprint matching. IEEE Transactions on Image Processing **9**(5) (2000) 846–859

[15] Ross, A., Reisman, J., Jain, A.K.: Fingerprint matching using feature space correlation. In: Biometric Authentication. Volume 2359 of LNCS., Springer Verlag (2002) 48–57

[16] de Sá, G., de Alencar Lotufo, R.: Improved fingercode matching function. In: SIBGRAPI. (2006) 263–272

[17] Maio, D., Maltoni, D., Cappelli, R., Wayman, J.L., Jain, A.K.: FVC2004: Third Fingerprint Verification Competition. In: ICBA. Volume 3072 of LNCS., Springer Verlag (2004) 1–7