Quality Assessment for Fingerprints Collected by Smartphone Cameras

Guoqiang Li, Bian Yang, Martin Aastrup Olsen, and Christoph Busch

Norwegian Information Security Laboratory
Gjøvik University College
Gjøvik, Norway
Email: {guoqiang.li;bian.yang;martin.olsen;christoph.busch}@hig.no

Abstract

We propose an approach to assess the quality of fingerprint samples captured by smartphone cameras under real-life scenarios. Our approach extracts a set of quality features for image blocks. Without needing segmentation, the approach determines a sample's quality by checking all image blocks divided from the sample and for each block a trained support vector machine gives a binary indication - "high-quality" or "non-high-quality" (including the low quality case and the background block case). A quality score is then generated for the whole sample. Experiments show this approach performs well in identifying the high quality blocks - the Spearman correlation coefficient between the proposed quality scores and samples' normalized comparison scores (ground truth) reaches 0.53 while the rate of false detection (background blocks judged as high-quality ones) is still low as 4.63 percent over a challenging dataset collected under various real-life scenarios.

1. Introduction

Fingerprint recognition has been widely used in industry and forensic area. It is quite common to select the dedicated sensors to acquire biometric samples in an controlled environment compliant to existing standards [5, 6]. However, smartphones are being found in almost everyone's pocket nowadays and normally embedded with a 5-mega-pixels (or above) camera, it becomes feasible to use these general-purposed cameras for capturing fingerprint samples. Previous research [3, 11, 14, 12] has shown that it is feasible to implement the fingerprint recognition functionality using
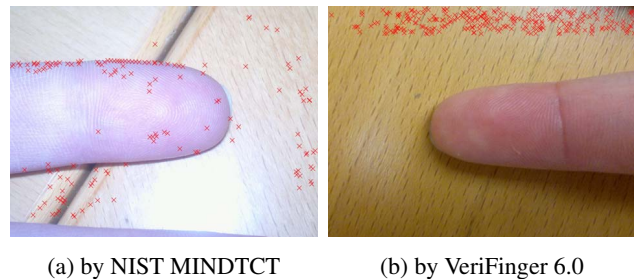
(a) by NIST MINDTCT   (b) by VeriFinger 6.0

Figure 1: High quality samples detected by NFIQ (score 1) with red cross indicating the detected minutae.

smartphones' cameras as an alternative to dedicated fingerprint sensors. Compared to the quality of the fingerprint samples captured under the ideal laboratory environment, the sample quality is quite unstable while data acquisition takes place under a real-life scenario [9] due to camera motion, de-focusing, poor illumination and complicated backgrounds. Thus it is essential to assess the sample quality before implementing practically useful biometrics-enabled applications on these smartphone cameras.

Several quality assessment methods and mechanisms have been proposed in the literature, such as [13, 4, 7], but they are designed for samples generated by dedicated fingerprint sensors. There are two fingerprint samples shown in Figure 1 which are considered as high quality ('score 1' by NIST Fingerprint Image Quality (NFIQ) [13]) but with a lot of spurious minutiae detected on the background. These methods are not designed to cope with fingerprint samples captured by smartphone cameras [14] with so complicated environments requiring accurate segmentation and noise (variance in lighting and color) suppression of the foreground (finger area). Consequently, their simple pre-processing mechanisms (*e.g.*, the quality map used in NFIQ to identify foreground blocks) which were accustomed to contact-based fingerprint patterns are not capable towards

such contactless-based samples any more.

We propose a segmentation-free approach in this paper to assess the quality of fingerprint samples captured by smartphones' cameras under real-life scenarios. A critical challenge during taking photo under real-life scenarios is the unpredictable background which may cause false detection of the finger area. Instead of using pixel-level foreground (finger area) segmentation, which could be both inaccurate and high in computational complexity for a mobile device, the approach checks each image block's quality status - high quality or non-high quality (*i.e.* the low quality and the background cases) - and combines all blocks' quality decisions to produce the final quality score for the sample. This quality score can be adopted to predict the sample's utility in terms of recognition performance, and then the camera can decide either to store the samples (if the quality score is large enough) or to automatically adjust the camera settings (such as the focusing distance or flash) for the next sample capture.

The remaining sections are organized as follows: Section 2 presents the proposed approach; Section 3 shows experimental results; and Section 4 concludes this paper.

2. The Proposed approach

2.1. Processes of the proposed approach

A conventional quality assessment usually include two steps: fingerprint area segmentation and quality prediction of the fingerprint area. Instead, we propose a one-step quality assessment approach which will not differentiate the foreground (fingerprint area) from the background in view of the computational efficiency and low-memory consumption requirement to mobile phones.
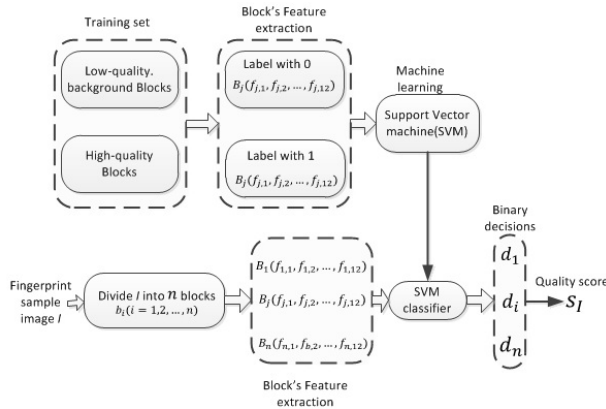

Figure 2: Processes of the proposed approach.

Figure 2 illustrates the processes of the proposed approach which uses support vector machine (SVM) to gener-

ate a quality binary decision d_i (1 = high quality; 0 = non-high quality) for each block $b_i(i = 1, 2, ..., n)$ divided from a sample image I. The SVM classifier is trained from a ground truth data set composed of high-quality block features with label one and non-high-quality block features with label zero to obtain a binary classifier. During quality assessment, we use the trained classifier to predict if each input block b_i divided from sample image I should be classified as high-quality or not. A global quality score S_I is generated to indicate the whole fingerprint sample's quality by counting the number of blocks labelled as high quality. To make this quality indicator more accurate, sample images can be resized to offset the variance in the finger-to-camera distance, as we did in Section 3.1.2.

2.2. Proposed quality features

2.2.1 Image block alignment in ridge orientation

A sample image is divided into non-overlapping blocks $\mathbf{b}_i^0(i = 1, 2, ..., N)$ sized $R \times C$ in pixel (R and $C = 2^k, k = 1, 2, 3, ...$) on which the quality features are computed. Before computing the features, the image blocks are aligned according to their ridge orientation using the PCA based gradient orientation estimation method [2]. That is, inside each block $\mathbf{b}_i^0$ neighbouring pixels' differences d_v and d_h (in both vertical and horizontal directions respectively) are obtained to form a gradient vector with orientation $tan^{-1}(d_v/d_h)$. Then the principal component analysis θ_i is calculated by exploiting PCA to find the principal one among all orientations of the $(R - 1) \times (C - 1)$ calculated gradient vectors. Now by clock-wise rotating the $\sqrt{2}(R-1) \times \sqrt{2}(C-1)$ size area concentric to $\mathbf{b}_i^0$ by angle θ_i we can crop a block $\mathbf{b}_i$ sized $R \times C$ concentric to $\mathbf{b}_i^0$. In this way we assume $\mathbf{b}_i$ has the maximum gradient value in the horizontal direction.

2.2.2 Quality features for block quality assessment

We propose 12 quality features $f_i(i = 1, 2, ..., 12)$ in three categories to assess an image block's quality: (1) pixel based features; (2) autocorrelation based features; and (3) frequency features from autocorrelation result. And the details are as follows.

1. Pixel based features

(1) f_1: Exposure (a block's gray level). Denote the average pixel value of $\mathbf{b}_i$, *i.e.*

$$f_1 = \frac{1}{R \times C} \sum_{r=1}^{R} \sum_{c=1}^{C} b_i(r, c) \qquad (1)$$

where $b_i(r, c)$ is the pixel value at the r-th row and c-th column inside the block $\mathbf{b}_i$.

(2) f_2: Significance of the principal component analysis. We represent it using the first eigenvalue λ_1 of the covariance matrix of all gradient vectors in the PCA calculation.

(3) f_3: Certainty of the block principal gradient orientation. We use a modified definition of *ocl* (orientation certainty level) in [8] as follows:

$$f_3 = \begin{cases} 1 - \frac{\lambda_2}{\lambda_1} & \text{if } \lambda_1 \neq 0 \\ 0 & \text{if } \lambda_1 = 0 \end{cases} \quad (2)$$

where λ_2 is the second eigenvalue of the covariance matrix of all gradient vectors in the PCA calculation.

2. Autocorrelation based features

Considering the fact that $\mathbf{b}_i$ has the principal gradient orientation aligned to the horizontal direction, autocorrelation calculation along the horizontal direction of $\mathbf{b}_i$ could be useful to enhance the dominant spatial frequencies and thus the autocorrelation result can be used for quality feature extraction. Instead of calculating autocorrelation directly, we do the autocorrelation on the horizontally-differential vectors $\mathbf{d}_i(r), 1 \leq r \leq R-1$. The details are as follows:

$$\mathbf{acr}_i = \sum_{r=1}^{R-1} autocorr(\mathbf{d}_i(r)) \quad (3)$$

where $\mathbf{d}_i(r) = (b_i(r,2) - b_i(r,1), b_i(r,3) - b_i(r,2), ..., b_i(r,C) - b_i(r,C-1))$.

The resultant $\mathbf{acr}_i$ is the $(C-1)$-dimensional sum-up vector with each row's autocorrelation calculated as follows:

$$autocorr(\mathbf{d}_i(r))(j) = \sum_{c=1}^{C-1} d_i(r,c)d_i(r,c+j) \quad (4)$$

where $(0 \leq j \leq C-2)$, with all $(C-1)$ amplitudes divided by the highest amplitude of $autocorr(\mathbf{d}_i(r))$. Before the follow-up feature extraction, low-pass filtering by setting zero the higher half of DCT-transform frequencies is used to smoothen the autocorrelation resultant vector. We denote the final $(C-1)$ dimensional vector as $\mathbf{ACR}_i$.

(4) f_4: $\mathbf{ACR}_i$'s peak active rate. From the observations in the experiments, we find the peaks of the $\mathbf{ACR}_i$ curve have a stable increasing rate if the sample quality is good enough. We use the 1-order polynomial (*i.e.* a straight line) to fit the M detected peak points with the x-coordinates $P_1(x), P_2(x), ..., P_M(x)(M) \ll C-1$ in the $\mathbf{ACR}_i$ curve (shown in Figure 3) and obtain a straight line with slope S on which M amplitudes $A(P_n(x))(n = 1, 2, ..., M)$ can be found. Then the $\mathbf{ACR}_i$'s peak active rate is defined as

$$f_4 = \frac{1}{M} \sum_{n=1}^{M} A(P_n(x))(A(P_n(x)) > 0). \quad (5)$$

(5) f_5: $\mathbf{ACR}_i$'s peak pick-up rate. We denote it as the slope S directly:

$$f_5 = S \quad (6)$$

(6) f_6: $\mathbf{ACR}_i$'s peak variance rate. We use this rate to represent the degree the M amplitudes $A(x)$ on the
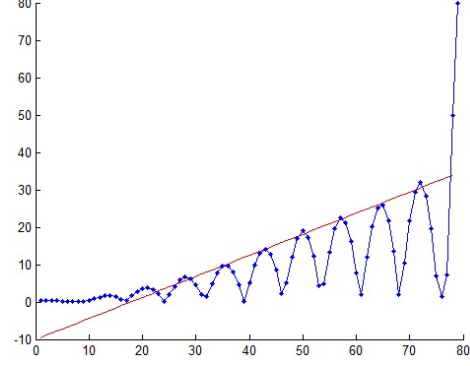

Figure 3: $\mathbf{ACR}_i$ curve, C=80 (the straight line is the linear best fit of the M peak points).

fitted line diverge from the actual M peak amplitudes $P_1(y), P_2(y), ..., P_M(y)$

$$f_6 = UP/DOWN. \quad (7)$$

where, $UP = (\frac{1}{M} \sum_{n=1}^{M} |P_n(y) - A(P_n(x))|)$
$DOWN = (max(A(P_n(x))) - min(A(P_n(x))))$.

(7) f_7: $\mathbf{ACR}_i$'s peak drop rate. We use this rate to represent the degree of the amplitude drop $AD_j = P_{j+1}(y) - P_j(y)(j = 1, 2, ..., M-1)$ of one peak compared to its previous counterpart. From the observation in the experiments, large drops in amplitude seldom happen to high quality blocks.

$$f_7 = UP/DOWN \quad (8)$$

where, $UP = 1 - (\sum_{j=1}^{M-1} |AD_j(AD_j < 0)|/(M-1))$
$DOWN = (max(A(P_n(x))) - min(A(P_n(x))))$.

3. Frequency features from the autocorrelation result

This category of features is derived from the frequency characteristics of the FFT coefficients of $\mathbf{ACR}_i$, which we denote as $f\mathbf{ACR}_i$ Frequency features are useful to represent the ridge spatial frequency characteristics.

(8) f_8: Principal frequency's amplitude:

$$f_8 = max(abs(f\mathbf{ACR}_i)) \quad (9)$$

(9) f_9: Principal frequency's index in vector $f\mathbf{ACR}_i$.
(10) f_{10}: Principal frequency's dominance rate.

$$f_{10} = \frac{4 \times \sum_{n=2}^{C/4} Q_i(n)}{(C-4) \times Q_i(1)} \quad (10)$$

where we denote $Q_i(1), Q_i(2), ..., Q_i(C/4)$ as the first quarter of $f\mathbf{ACR}_i$'s components sorted by descending amplitude.

(11) f_{11}: Principal frequency's prominence rate – close neighbours.

$$f_{11} = \frac{(\sum_{n=-H}^{H} f\mathbf{ACR}_i(L+n)) - f\mathbf{ACR}_i(L)}{2H \times f\mathbf{ACR}_i(L)} \quad (11)$$

where L is denoted as the feature f_9 that is the principal frequency's index in the vector $f\mathbf{ACR}_i$. We consider $2H$ neighbours around the principal frequency.

(12) f_{12}: Principal frequency's prominence rate – second close neighbours:

$$f_{12} = \frac{\sum\limits_{n=-X}^{X} f\mathbf{ACR}_i(L+n) - \sum\limits_{n=-H}^{H} f\mathbf{ACR}_i(L+n)}{2(X-H) \times f\mathbf{ACR}_i(L)} \quad (12)$$

where L is the principal frequency's index in the vector $f\mathbf{ACR}_i$, $0 < H < X < C$. And $L - X > 0$, otherwise $f_{12} = 0$.

2.2.3 Feature dynamic range normalization

All the 12 features $f_i(i = 1, 2, ..., 12)$ are z-score normalized prior to being used by the SVM as:

$$f_i' = \frac{f_i - E(f_i)}{\sigma(f_i)} \quad (13)$$

where $E(f_i)$ and $\sigma(f_i)$ are expectation and standard deviation values of the feature f_i.

3. Experimental design and results

Good sample quality can be represented by its high normalized comparison scores [13]. We evaluate in this section how well the normalized comparison score, as the ground truth, and the quality score generated from the proposed approach correlate. Spearman's rank correlation coefficient [10] is computed between the two for each sample.

3.1. Experimental setup

3.1.1 Data collection and experimental settings

Three smart phones - iPhone 4, Samsung Galaxy S, and Nokia N8 - were selected to capture fingerprint samples from 100 different fingers of 25 groups (corresponding to 25 subjects) of right index finger, right middle finger, left index finger and left middle finger. Table 1 lists the specification of selected mobile phone cameras. Three scenarios are tested: indoor scenario with good illumination but challenging background with similar color and texture as fingers (shown in Figure 1); dark scenario with illumination only from the smartphone automatic flash; outdoor scenario with complex background such as buildings, lawns, lakes and trees. Figure 4 shows the finger examples generated in the three scenarios respectively. We use each phone to capture three samples for each finger in the first and third scenarios, but only Nokia N8 in the second scenario (the other two failed to take photos in darkness). In total, there are 2100 fingerprint samples captured. For quality assessment, the parameters used in our experiments were set as: $R = C = 80$, $H = 2$, $X = 4$ (refer to Section 2.2). 100

Mobile phone	Nokia N8	iPhone 4	Samsung Galaxy S
Mega pixel	12.0	5.0	5.0
Resolution	1536×1936	2592×1936	1600×960
Auto-focus	Yes	Yes	Yes
Image format	JPEG	JPEG	JPEG
ISO control	automatic	automatic	automatic
Flash source	Xenon	LED	no flash
Flash setting	automatic	automatic	no flash
Aperture	f/2.8	f/2.8	f/2.6
Sensor size	1/1.83"	1/3.2"	1/3.6"

Table 1: Specification of the 3 smartphones' cameras.

high-quality blocks and 200 non-high-quality ones (visually judged as ground truth) with size RC were randomly cropped from samples for SVM training.
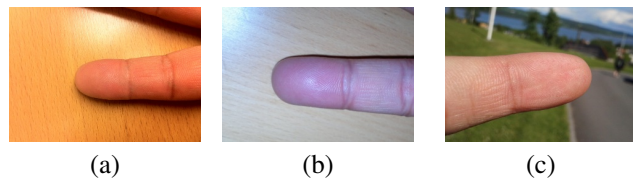

(a)   (b)   (c)

Figure 4: Fingerprint samples under 3 different scenarios: (a) In-door, (b) Dark(auto flash) and (c) Out-door.

3.1.2 Sample pre-processing

Although our proposed approach does not need to segment the foreground (finger area) for quality assessment, in practice for recognition purpose pre-processing are usually needed over the samples directly output from the cameras. Such pre-processing steps could include (1) segmentation of the fingerprint area; (2) sample resizing (to offset the distance variance of fingers from the camera); and (3) fingerprint area enhancement. Instead of performing such pre-processing steps for quality assessment, we need to do them in this paper to calculate the normalized comparison score [13] of each sample to obtain the ground truth of the sample's quality. We tested two types of pre-processed samples in our experiments as follows.

Pre-processing Type 1. Segmentation only, in which only manually segmentation is performed to crop the foreground, without applying sample resizing and enhancement. This type provides the baseline condition for normalized comparison score calculation.

Pre-processing Type 2. Manual segmentation of the foreground, foreground resizing, and foreground enhancement. Resizing is realized by the following steps: (1) fitting the

finger-tip shape as a half-circle, detect this finger-tip circle using Hough transform over the boundary of the foreground; (2) align the radius of the detected finger-tip half-circle to a fixed value (20 pixels in our experiments); and (3) resize the whole cropped sample according to the new aligned radius value. In this way, all the resized samples contain finger-tips with almost the same radius value. After the resizing, the fingerprint enhancement implementation from [1] is applied to generate ridge orientation and frequency enhanced images.

Note that for both types, segmentation is done in a manual way which is necessary because the segmented foreground is deemed as ground truth for normalized comparison score calculation. At the recognition phase, segmentation algorithm such as the pre-processing in [12] can be applied to realize segmentation in real time. How to improve the pre-processing steps is out of the scope of this paper.

Also note that all the pre-processing steps mentioned above are only for recognition performance and normalized comparison scores calculation and they are not at all used by our proposed quality assessment approach. In this paper, all the quality scores are generated from the full size original samples with full backgrounds. We assume such a pre-processing-free quality estimation step is desirable for smartphones in terms of efficiency and power saving, considering the accurate segmentation and enhancement of foreground shall involve high computational complexity.

3.1.3 Accuracy performance evaluation

To evaluate the recognition accuracy performance we generate two datasets called 'original cropped samples' and 'enhanced cropped samples' corresponding to the Type 1 and Type 2 processed data in Section 3.1.2.

We used two software - NIST MINDTCT and the NeuroTechnology VeriFinger 6.0 to generate the templates from original cropped samples and enhanced cropped samples. By NIST MINDTCT there are 2100 templates generated as expected. By VeriFinger 6.0 there are only 424 templates belonging to 73 fingers generated from the original cropped samples and 906 templates belonging to 97 fingers generated from the enhanced cropped samples due to the sample quality checking functionality inherent in the software. We see by sample enhancement the number of samples that can generate templates by VeriFinger 6.0 doubles (from 424 to 906). We give in Table 2 and 3 an example of accuracy performance using VeriFinger 6.0 over the 424 templates from the original cropped samples. Note that in both tables the sums of references and probes are less than 424 - this is because some fingers have only one sample for a specific camera or scenario and thus not selected for performance calculation.

Computing scenario	Nokia	iPhone	Samsung
Number of reference images	63	31	23
Number of probe images	193	44	44
Number of imposter scores	11929	1239	968
EER	4.3%	2.3%	5.3%

Table 2: EER value of intra-cameras using VeriFinger 6.0 based on 424 templates from original cropped samples.

Computing scenario	Indoor	Darkness	Outdoor
Number of reference images	50	53	26
Number of probe images	117	97	55
Number of imposter scores	5268	4378	1008
EER	19.6%	0.01%	1.8%

Table 3: EER value of intra-scenario using VeriFinger 6.0 based on 424 templates from original cropped samples.

3.2. Distribution of the quality scores

Figure 5(a) gives the distribution of the quality scores of all 2100 samples, with the minimum score 0 and the maximum score 47. As a reference to the performance examples in Table 2 and 3, the distribution of the quality scores of the 424 samples from which templates can be generated by VeriFinger 6.0 is depicted in Figure 5(b). We can observe some correlation between the proposed quality scores and the binary quality decision made by VeriFinger 6.0 (*i.e.*, most of the samples that generate templates have the quality score larger than 4).

3.3. Evaluation of the proposed quality assessment approach

In this section we analyse the correlation between the normalized comparison score c_i and the quality score q_i generated by our approach for each sample x_i. The normalized comparison score is defined as follows according to the NIST definition [13].

$$c(x_i) = \frac{s_m(x_i) - E[s_n(x_{ji})]}{\sigma(s_n(x_{ji}))} \tag{14}$$

where $E[\cdot]$ is mathematical expectation, and $\sigma(\cdot)$ is standard deviation, $s_m(x_i)$ is the genuine comparison score gener-
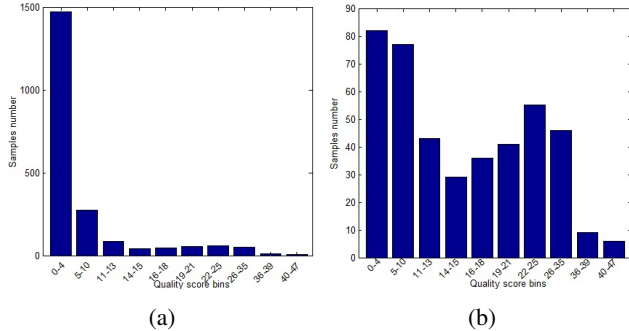
(a)   (b)

Figure 5: Quality scores distribution: (a) 2100 samples and (b) 424 samples that can successfully generate minutiae templates by VeriFinger 6.0 extractor.
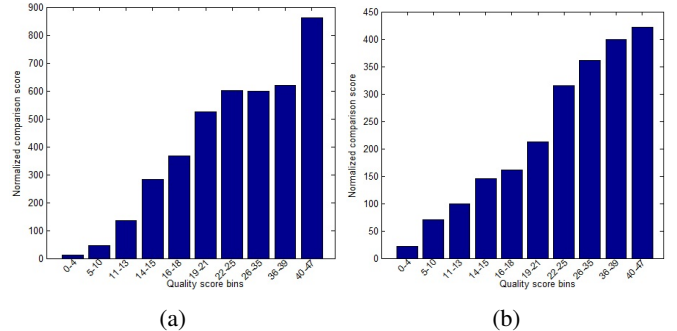

(a)   (b)

Figure 6: Normalized comparison scores v.s. proposed quality scores under 10 quality score bins. (a): Normalized comparison scores generated from original cropped samples (424 samples with templates + 1676 zero comparison score samples). (b): Normalized comparison scores generated from enhanced cropped samples ((906 samples with templates + 1194 zero comparison score samples)).

ated by comparing the samples from the same finger and $s_n(x_{ji})$ are the imposter scores of sample x_i generated by comparing the samples from different fingers, $\forall j, i \neq j$.

As we mentioned in the section 3.1.3, there are two types of datasets (original cropped samples and enhanced cropped samples) that are used to generate the normalized comparison scores. The comparison scores $s_m(x_i)$ and $s_n(x_{ji})$ are produced by both NIST BOZORTH3 and NeuroTechnology VeriFinger 6.0 comparator. In the VeriFinger 6.0 case, we assign the comparison score 0 to those samples that cannot successfully generate templates. We use the samples with maximum sum of intra-finger sample comparison scores as references for enrolment in recognition performance testing. In order to include these reference samples into the correlation calculation, we assign the largest comparison score found in the testing to all these reference samples for normalized comparison score calculation. At last, we obtain a group of score pairs $(c_i, q_i), i = 1, 2, ..., 2100$. To illustrate the correlation of the two types of scores, we can quantize the quality scores q_i into 10 bins and calculate the average value of the normalized comparison scores c_i in each quality score bin. An example of such correlation is shown in Figure 6 where the comparison scores are generated by VeriFinger 6.0 from the two datasets (original and enhanced cropped samples). The graphs indicate very high correlation between the two types of scores.

We compute the Spearman's rank correlation coefficient ρ as a quantitative method to analyze how well two variables c_i and q_i correlate. The results are given in Table 4 with different experimental settings for generating the normalized comparison scores (note that for generating the proposed quality scores we use the same 2100 full size original samples with full background for all settings). The results show that the proposed quality metrics are accurate to assess the samples' quality in all settings assuming the normalized comparison score for each sample as the ground

Experimental settings using normalized comparison scores as ground truth	ρ
424 Original cropped samples using NeuroTechnology VeriFinger 6.0 comparator	0.59
906 Enhanced cropped samples using NeuroTechnology VeriFinger 6.0 comparator	0.42
2100 Original cropped samples (424 template-generated samples + 1676 samples with manually set normalized comparison score zero) using VeriFinger 6.0 comparator	0.47
2100 Enhanced cropped samples (906 template-generated samples + 1194 samples with manually set normalized comparison score zero) using VeriFinger 6.0 comparator	0.53
2100 Original cropped samples using BOZORTH3	0.20
2100 Enhanced cropped samples using BOZORTH3	0.49
2100 NFIQ on original full samples	-0.06
2100 NFIQ on original cropped samples	0.07

Table 4: Spearman's rank correlation coefficients ρ under different experimental settings using the normalized comparison scores as ground truth for quality (for quality score generating full size original samples with full background are used for all settings).

truth of sample quality. The NFIQ related results in Table 4 provide a reference to demonstrating the effectiveness and advantage of the proposed quality assessment approach. Generally speaking, the quality of samples that can suc-

Experimental settings using VeriFinger 6.0's template generation decision as ground truth	ρ
Original cropped samples	0.45
Enhanced cropped samples	0.57

Table 5: Spearman's rank correlation coefficients ρ under two experimental settings using VeriFinger 6.0 sample quality checking binary decision as ground truth for quality (for quality score generating full size original samples with full background are used for all settings).

	False detected blocks	Total detect high quality blocks	Rate
424 samples	112	6312	1.77%
906 samples	258	8369	3.08%
2100 samples	471	10155	4.63%

Table 6: Rate of false detection (background blocks identified as high-quality ones).

Features	1st	2nd	3rd	4th
ρ	0.05	0.04	0.08	0.30
Features	5th	6th	7th	8th
ρ	0.30	0.26	0.27	0.27
Feature	9th	10th	11th	12th
ρ	0.14	0.27	0.26	0.35

Table 7: Spearman's rank correlation coefficient ρ between individual block features and the block quality decision.

Type	Group 1	Group 2	Group 3
Quality score	0-3	4-11	12-47
Samples number	1401	376	323
EER from original images	48.6%	46.6%	45.2%
EER from enhanced images	49.0%	35.0%	24.1%

Table 8: EER under different levels of quality score from the 2100 samples using NIST BOZORTH3.

cessfully produce template via NeuroTechnology VeriFinger 6.0 extractor should be better than that of those samples that fail to generate the templates. If we assign a score 1 to those samples with template generated and a score 0 to those without, a pair of (t_i, q_i) can be constructed where t_i equals 0 or 1. The Spearman's rank correlation coefficients for these 2100 pairs (t_i, q_i) are shown in Table 5, which also indicates a high correlation.

3.4. The false detection case

In our experiments, there are a few samples with false detection (background blocks labelled as high-quality ones) mostly in the in-door scenario of the challenging background - the office desk surface (seen in Figure 1 and Figure 4(a)) - has the texture and the color looking similar to finger areas. The total number of false detected high-quality blocks is 471 blocks accounting for 4.63% of all 10155 high quality blocks on these 2100 samples, listed in Table 6.

3.5. Correlation between individual features and the block quality decision

We also evaluated the correlation between each of 12 block features and the binary quality decision for each block by computing Spearman's rank correlation coefficient. Table 7 shows the correlation coefficient for each feature.

3.6. Purpose verification of quality assessment: EER under different levels of quality scores

Recall that the purpose of sample quality assessment is to select high quality samples for recognition use. To verify if this purpose is achieved by the proposed approach or not, we calculate EERs under three levels of quality scores using NIST BOZORTH3 and NeuroTechnology VeriFinger 6.0 on different datasets. The sample with maximum quality score is always selected as the reference sample for each finger in all experiments. There are four types of combinations to compute EERs as follows:

(1). We divide the 2100 original cropped samples into 3 groups in terms of quality score: Group 1 with quality score 0-3 (more than 50% samples are with low quality), Group 2 with quality score 4-11, and Group 3 with quality score larger than 11. NIST MINDTCT and BOZORTH3 are used to extract and compare the templates. The experimental results are shown at the row "EER from original samples" in Table 8.

(2). Using the same settings as (1) but on 2100 cropped enhanced samples. The experimental results are shown at the row "EER from enhanced samples" in Table 8.

(3). We only used the 424 original cropped samples with templates generated by NeuroTechnology VeriFinger 6.0. And the three groups are $[0-9], [10-19], [20, 47]$. NeuroTechnology VeriFinger 6.0 is used to generate the comparison scores. The results are shown in Table 9.

(4). We only use 906 enhanced cropped samples with templates generated by NeuroTechnology VeriFinger 6.0.

Type	Group 1	Group 2	Group 3
Quality score	0-9	10-19	20-47
Samples number	147	136	141
Number of genuine scores	90	74	90
Number of imposter scores	3757	2746	3266
EER	22.2%	12.8%	3.9%

Table 9: EER under different levels of quality score from 424 original samples using VeriFinger 6.0.

Type	Group 1	Group 2	Group 3
Quality score	0-3	4-14	15-47
Samples number	362	324	220
Number of genuine scores	275	242	147
Number of imposter scores	19187	16027	8968
EER	35.3%	22.5%	2.7%

Table 10: EER under different levels of quality score from 906 enhanced samples using VeriFinger 6.0.

And the three groups are $[0 - 3], [4 - 14], [15, 47]$. NeuroTechnology VeriFinger 6.0 is used to generate the comparison scores. The results are shown in Table 10.

Note that in Table 8-10 we try to group the samples in even distribution of sample amount. We observe that EERs are significantly reduced along the increase of sample quality except the case NIST BOZORTH3 operating on original samples which is however not very likely to be adopted for practical use. The experimental results demonstrate the effectiveness of our proposed quality assessment approach in predicting the quality of fingerprint samples generated by smartphone cameras. Note that for Group 1 and 2, the EERs in Table 10 is higher than those in Table 9, which could be due to the fact that sample enhancement increases the number of samples that can generate templates but decreases the average sample quality in the meanwhile.

4. Conclusion and future work

This paper proposes an effective fingerprint sample quality assessment approach for the samples captured by smartphone cameras using a set of block based quality features. Our approach is pre-processing-free (without needing segmentation and enhancement) and block-based (memory saving and parallelizable in computation) thus potentially efficient in computation on mobile devices. The correlation between the quality score generated by the proposed approach and the normalized comparison score (as ground truth of quality) of each sample has been evaluated by computing Spearman's rank correlation coefficient of the two scores. Experimental results demonstrate that the proposed quality assessment approach is capable of identifying the high-quality fingerprint area from both those low-quality ones and those complicated background ones and thus capable of predicting the sample quality. Our future work will focus on reducing the false detection rate and improve the block size normalization across different cameras.

References

[1] Fingerprint enhancement implementation. `http://www.csse.uwa.edu.au/~pk/research/matlabfns/`. Accessed: 2013-01-30.

[2] A. Bazen and S. Gerez. Directional field computation for fingerprints based on the principal component analysis of local gradients. In *Proceedings of ProRISC2000, 11th Annual Workshop on Circuits, Systems and Signal Processing*, pages 215–222. Citeseer, 2000.

[3] M. Derawi, B. Yang, and C. Busch. Fingerprint recognition with embedded cameras on mobile phones. *Security and Privacy in Mobile Information and Communication Systems*, pages 136–147, 2012.

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

[5] ISO/IEC 19794-1:2011. Information technology – Biometric data interchange formats – Part 1: Framework.

[6] ISO/IEC 19794-4:2011. Information technology – Biometric data interchange formats – Part 4: Finger image data.

[7] ISO/IEC 29794-1:2009. Information technology – Biometric sample quality – Part 1: Framework.

[8] ISO/IEC TR 29794-4:2010. Information technology – Biometric sample quality – Part 4: Finger image data.

[9] G. Li, B. Yang, R. Raghavendra, and C. Busch. Testing mobile phone camera based fingerprint recognition under real-life scenarios. *NISK*, 2012, 2012.

[10] J. L. Myers and A. Well. *Research design and statistical analysis*. Psychology Press, 2003.

[11] V. Piuri and F. Scotti. Fingerprint biometrics via low-cost sensors and webcams. In *2nd IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS), 2008*, pages 1–6. IEEE, 2008.

[12] C. Stein, C. Nickel, and C. Busch. Fingerphoto recognition with smartphone cameras. In *Proceedings of the International Conference of the Biometrics Special Interest Group (BIOSIG), 2012*, pages 1–12. IEEE, 2012.

[13] E. Tabassi, C. Wilson, and C. Watson. NIST-IR 7151 - fingerprint image quality. In *Technical report, NIST*, 2004.

[14] B. Yang, X. Li, and C. Busch. Collecting fingerprints for recognition using mobile phone cameras. *Electronic Imaging*, 8304:83040L, 2012.