A New Approach to Fake Finger Detection Based on Skin Distortion*,**

A. Antonelli, R. Cappelli, Dario Maio, and Davide Maltoni

Biometric System Laboratory - DEIS,
University of Bologna, via Sacchi 3, 47023 Cesena - Italy
{athos, cappelli, maio, maltoni}@csr.unibo.it

Abstract. This work introduces a new approach for discriminating real fingers from fakes, based on the analysis of human skin elasticity. The user is required to move the finger once it touches the scanner surface, thus deliberately producing skin distortion. A multi-stage feature- extraction technique captures and processes the significant information from a sequence of frames acquired during the finger movement; this information is encoded as a sequence of *DistortionCodes* and further analyzed to determine the nature of the finger. The experimentation carried out on a database of real and fake fingers shows that the performance of the new approach is very promising.

1 Introduction

Thanks to the largely-accepted uniqueness of fingerprints and the availability of low-cost acquisition devices, fingerprint-based authentication systems are becoming more and more popular and are being deployed in several applications: from logon to PC, electronic commerce, ATMs, to physical access control for airports and border control [7]. On the other hand, as any other security system, fingerprint recognition is not totally spoof-proof; the main potential attacks can be classified as follows [1][4]: 1) attacking the communication channels, including replay attacks on the channel between the sensor and the rest of the system and other types of attacks; 2) attacking specific software modules (e.g. replacing the feature extractor or the matcher with a Trojan horse); 3) attacking the database of enrolled templates; 4) presenting fake fingers to the sensor. The feasibility of the last type of attack has been recently proved by some researchers [2][3]: current fingerprint recognition systems can be fooled with well-made fake fingers, created with the collaboration of the fingerprint owner or from latent fingerprints (in that case the procedure is more difficult but still possible). Some approaches recently proposed in the literature to address this problem can be found in [5] [6]. This work introduces a novel method for discriminating fake fingers from real ones based on the analysis of a peculiar characteristic of the human skin: the elasticity. Some preliminary studies showed that when a real finger moves on a scanner surface, it produces a significant amount of distortion, which is quite different from that produced by fake fingers. Usually fake fingers are more rigid than skin and

* This work was partially supported by European Commission (BioSec - FP6 IST-2002-001766).
** Patent pending (IT #BO2005A000399).

D. Zhang and A.K. Jain (Eds.): ICB 2006, LNCS 3832, pp. 221 – 228, 2005.

the deformation is lower and, even if made of highly elastic materials, it seems very difficult to precisely emulate the specific way a real finger is distorted, because is related to how the external skin is anchored to the underlying derma and influenced by the position and shape of the finger bone. The rest of this work is organized as follows: section 2 describes the proposed approach, section 3 reports the experimentation carried out to validate the new technique and section 4 draws some conclusions.

2 The Fake Finger Detection Approach

The user is required to place a finger onto the scanner surface and, once in touch with it, to apply some pressure while rotating the finger in a counter-clockwise direction (this particular movement has been chosen after some initial tests, as it seems comfortable for user and it produces the right amount of deformation). A sequence of frames is acquired at high frame rate (at least 20 fps) during the movement and analyzed to extract relevant features related to skin distortion. At the beginning of the sequence, the finger is assumed relaxed (i.e. non-distorted), without any superficial tension. A pre-processing stage is performed to simplify the subsequent steps; in particular:

- any frame such that the amount of rotation with respect to the previous one (inter-frame rotation) is less than θ_{min} ($\theta_{min} = 0.25°$ in our experimentation) is discarded (the inter-frame rotation angle is calculated as described in section 2.2);
- only frames acquired when the (accumulated) finger rotation is less than ϕ_{max} ($\phi_{max} = 15°$ in our experimentation) are retained: when angle ϕ_{max} is reached, the sequence is truncated (the rotation angle of the finger is calculated as described in section 2.5).

Let $\{F_1, F_2, ..., F_n\}$ be a sequence of *n* images that satisfies the above constraints; the following steps are performed on each frame F_i (figure 1):

- isolation of the fingerprint area from the background;
- computation of the optical flow between the current frame and the next one;
- computation of the distortion map;
- temporal integration of the distortion map;
- computation of the DistortionCode from the integrated distortion map.

For each image F_i, the isolation of the fingerprint area from the background is performed by computing the gradient of the image block-wise: let $\mathbf{p} = [x, y]^T$ be a generic pixel in the image and $F_i(\mathbf{p})$ a square image block (with side 12 in our tests) centred in $\mathbf{p}$: each $F_i(\mathbf{p})$ whose gradient module exceeds a given threshold is associated to the foreground. Only the foreground blocks are considered in the rest of the algorithm.

2.1 Computation of the Optical Flow

Block-wise correlation is computed to detect the new position $\mathbf{p}'$ of each block $F_i(\mathbf{p})$ in frame F_{i+1}. The vector $\Delta\mathbf{p}_i = \mathbf{p}' - \mathbf{p}$ denotes, for each block $F_i(\mathbf{p})$, the

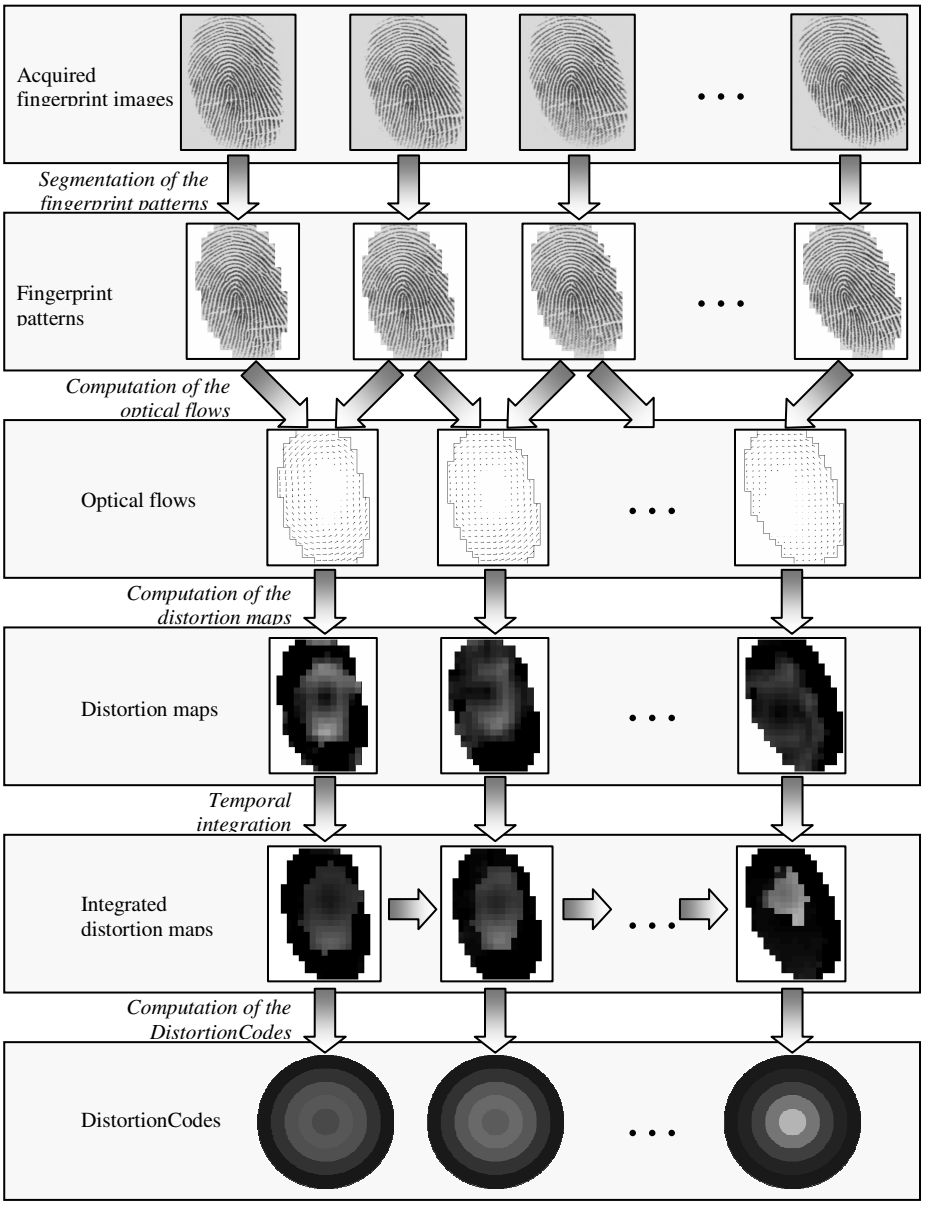

Fig. 1. The main steps of the feature extraction approach: a sequence of acquired fingerprint images is processed to obtain a sequence of DistortionCodes

estimated horizontal and vertical movements ($\Delta\mathbf{p}_i = [\Delta x, \Delta y]^T$); these movement vectors are known in the literature as *optical flow*. This method is in theory only translation-invariant but, since the images are taken at a fast frame rate, for small blocks it is possible to assume a certain rotation- and deformation-invariance.

In order to filter out outliers produced by noise, by false correlation matches or by other anomalies, the block movement vectors $\Delta\mathbf{p}_i$ are then processed as follows.

1. Each $\Delta\mathbf{p}_i$ such that $\|\Delta\mathbf{p}_i\| \geq \max_{\Delta\mathbf{p}_{i-1}} \|\Delta\mathbf{p}_{i-1}\| + \alpha$ is discarded. This step allows to remove outliers, under the assumption that the movement of each block cannot deviate too much from the largest movement of the previous frame blocks; α is a parameter that should correspond to the maximum expected acceleration between two consecutive frames ($\alpha = 3$ in our tests).
2. For each $\Delta\mathbf{p}_i$, the value $\overline{\Delta\mathbf{p}_i}$ is calculated as the weighted average of the 3x3 neighbours of $\Delta\mathbf{p}_i$, using a 3x3 Gaussian mask; elements discarded by the previous step are not included in the average: if no valid elements are present, $\overline{\Delta\mathbf{p}_i}$ is marked as "invalid".
3. Each $\Delta\mathbf{p}_i$ such that $\|\Delta\mathbf{p}_i - \overline{\Delta\mathbf{p}_i}\| \geq \beta$ is discarded. This step allows to remove elements that are not consistent with their neighbours; β is a parameter that controls the strength of this procedure ($\beta = \sqrt{3/2}$ in our experimentation).
4. The values $\overline{\Delta\mathbf{p}_i}$ are recalculated (as in step 2) by considering only the $\Delta\mathbf{p}_i$ retained at step 3.

2.2 Computation of the Distortion Map

The centre of rotation $\mathbf{c}_i = [cx_i, cy_i]^T$ is estimated as a weighted average of the positions $\mathbf{p}$ of all the foreground blocks $F_i(\mathbf{p})$ such that the corresponding movement vector $\overline{\Delta\mathbf{p}_i}$ is valid:

$$\mathbf{c}_i = \mathrm{E}\left[\left\{\mathbf{p}\,\frac{1}{1+\|\overline{\Delta\mathbf{p}_i}\|} \mid \overline{\Delta\mathbf{p}_i} \text{ is valid}\right\}\right], \tag{1}$$

where $\mathrm{E}[\mathrm{A}]$ is the average of the elements in set A.

An inter-frame rotation angle θ_i (according to $\mathbf{c}_i$) and a translation vector $\mathbf{t}_i = [tx_i, ty_i]^T$ are then computed in the least square sense, starting from all the average movement vectors $\overline{\Delta\mathbf{p}_i}$. If the finger were moving solidly, then each movement vector would be coherent with θ_i and $\mathbf{t}_i$. Even if the movement is not solid, θ_i and $\mathbf{t}_i$ still encode the dominant movement and, for each block $\mathbf{p}$, the distortion can be computed as the incoherence of each average movement vector $\overline{\Delta\mathbf{p}_i}$ with respect to θ_i and $\mathbf{t}_i$. In particular, if a movement vector were computed according to a solid movement, then its value would be:

$$\widetilde{\Delta\mathbf{p}_i} = \begin{bmatrix} \cos\theta_i & \sin\theta_i \\ -\sin\theta_i & \cos\theta_i \end{bmatrix}(\mathbf{p}_i - \mathbf{c}_i) + \mathbf{c}_i + \mathbf{t}_i - \mathbf{p}_i \tag{2}$$

and therefore the distortion can be defined as the residual:

$$D_i(\mathbf{p}) = \begin{cases} \|\widetilde{\Delta\mathbf{p}_i} - \overline{\Delta\mathbf{p}_i}\| & \text{if } \overline{\Delta\mathbf{p}_i} \text{ is valid} \\ \mathit{undefined} & \text{otherwise} \end{cases} \tag{3}$$

A *distortion map* is defined as a block-wise image whose blocks encode the distortion values $D_i(\mathbf{p})$.

2.3 Temporal Integration of the Distortion Map

The computation of the distortion map, made on just two consecutive frames, is affected by the following problems:

- the movement vectors are discrete (because of the discrete nature of the images) and in case of small movement the loss of accuracy might be significant;
- errors in seeking the new position of blocks could lead to a wrong distortion estimation;
- the measured distortion is proportional to the amount of movement between the two frames (and therefore depend on the finger speed), without considering previous tension accumulated/released. This makes difficult to compare a distortion map against the distortion map in another sequence.

An effective solution to the above problems is to perform a temporal-integration of the distortion map, resulting into an *integrated distortion map*. The temporal integration is simply obtained by block-wise summing the current distortion map to the distortion map "accumulated" in the previous frames. Each integrated distortion element is defined as:

$$TID_i(\mathbf{p}) = \begin{cases} TID_{i-1}(\mathbf{p}) + D_i(\mathbf{p}) & \text{if } \left\|\widetilde{\Delta\mathbf{p}_i}\right\| > \left\|\overline{\Delta\mathbf{p}_i}\right\| \text{ and } \overline{\Delta\mathbf{p}_i} \text{ is valid} \\ TID_{i-1}(\mathbf{p}) & \text{if } \overline{\Delta\mathbf{p}_i} \text{ is invalid} \\ 0 & \text{if } \left\|\widetilde{\Delta\mathbf{p}_i}\right\| \leq \left\|\overline{\Delta\mathbf{p}_i}\right\| \end{cases} \tag{4}$$

with $TID_0(\mathbf{p}) = 0$.

The rationale behind the above definition is that if the norm of the average movement vector $\overline{\Delta\mathbf{p}_i}$ is smaller than the norm of the estimated solid movement $\widetilde{\Delta\mathbf{p}_i}$, then the block is moving slower than expected and this means it is accumulating tension. Otherwise, if the norm of $\overline{\Delta\mathbf{p}_i}$ is larger than the norm of $\widetilde{\Delta\mathbf{p}_i}$, the block is moving faster than expected, thus it is slipping on the sensor surface, releasing the tension accumulated.

The integrated distortion map solves most of the previously listed problems: i) discretization and local estimation errors are no longer serious problems because the integration tends to produce smoothed values; ii) for a given movement trajectory, the integrated distortion map is quite invariant with respect to the finger speed.

2.4 The Distortion Code

Comparing two sequences of integrated distortion maps, both acquired under the same movement trajectory, is the basis of our fake finger detection approach. On the other hand, directly comparing two sequences of integrated distortion maps would be computationally very demanding and it would be quite difficult to deal with the unavoidable local changes between the sequences.

To simplify handling the sequences, a feature vector (called *DistortionCode* for the analogy with the FingerCode introduced in [9]) is extracted from each integrated distortion map: *m* circular annuli of increasing radius ($r \cdot j,\ j = 1..m$, where *r* is the radius of the smaller annulus) are centred in $\mathbf{c}_i$ and superimposed to the map (*r*=20 and *m*=5 in our experimentation). For each annulus, a feature d_{ij} is computed as the average of the integrated distortion elements of the blocks falling inside it:

$$d_{ij} = \mathrm{E}\left[\left\{TID_i(\mathbf{p}) \mid \mathbf{p} \text{ belongs to annulus } j\right\}\right] \tag{5}$$

A DistortionCode $\mathbf{d}_i$ is obtained from each frame F_i, *i*=1..*n*-1:

$\mathbf{d}_i = \left[d_{i1}, d_{i2}, ..., d_{im}\right]^T$

A *DistortionCode sequence V* is then defined as:

$$V = \left\{\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_{n-1}\right\}, \text{ where } \mathbf{v}_k = \mathbf{d}_k \Big/ \sqrt{\sum_{i=1..n-1} \left\|\mathbf{d}_i\right\|^2} \tag{6}$$

The obtained DistortionCode sequence characterizes the deformation of a particular finger under a specific movement. Further sequences from the same finger do not necessarily lead to the same DistortionCode sequence: the overall length might be different, because the user could produce the same trajectory (or a similar trajectory) faster or slower. While a minor rotation accumulates less tension, during a major rotation the finger could slip and the tension be released in the middle of the sequence.

2.5 The Distortion Match Function

In order to discriminate a real from a fake finger, the DistortionCode sequence acquired during the enrolment and associated to a given user is compared with the DistortionCode sequence acquired at verification/identification time. Let $V_T = \left\{\mathbf{v}_{T,1}, \mathbf{v}_{T,2}, ...\mathbf{v}_{T,n_T}\right\}$ and $V_C = \left\{\mathbf{v}_{C,1}, \mathbf{v}_{C,2}, ...\mathbf{v}_{C,n_C}\right\}$ be the sequence acquired during the enrolment (template sequence) and the new one (current sequence), respectively; a *Distortion Match Function* $DMF\left(V_T, V_C\right)$ compares the template and current sequence and returns a score in the range [0..1], indicating how much the current sequence is similar to the template (1 means maximum similarity).

A Distortion Match Function must define how to: 1) calculate the similarity between two DistortionCodes, 2) align the DistortionCodes by establishing a correspondence between the DistortionCodes in the two sequences V_T and V_C, and finally 3) measure the similarity between the two aligned sequences.

A simple Euclidean distance between two DistortionCodes has been adopted as to comparison metric (step 1). As to step 2), DistortionCodes are aligned according to the accumulated rotation angles ϕ_i ($\phi_i = \sum_{k=1..i} \theta_k$, where θ_i is the inter-frame rotation angle between the frames *i* and *i*+1); re-sampling through interpolation is performed to deal with discretization; the result of step 2) is a new DistortionCode sequence $\tilde{V}_T = \left\{\tilde{\mathbf{v}}_{T,1}, \tilde{\mathbf{v}}_{T,2}, ..., \tilde{\mathbf{v}}_{T,n_C}\right\}$, obtained from V_T after the alignment with V_C; $\tilde{V}_T$ has

the same cardinality of V_C. The final similarity can be simply computed (step 3) as the average Euclidean distance of corresponding DistortionCodes in $\tilde{V}_T$ and V_C:

$$DMF(V_T, V_C) = 1 - \sum_{i=1..n_C} \left\| \mathbf{v}_{C,i} - \tilde{\mathbf{v}}_{T,i} \right\| \Big/ \sqrt{m} \cdot n_C \tag{7}$$

The normalization coefficient ($\sqrt{m} \cdot n_C$) ensures that the score is always in the range [0..1].

3 Experimental Results

A fingerprint scanner that embeds a fake-finger detection mechanism has to decide, for each transaction, if the current sample comes from a real finger or from a fake one. This decision will be unavoidably affected by errors: a scanner could reject real fingers and/or accept fake fingers.

Let FAR_{fd} be the proportion of transactions with a fake finger that are incorrectly accepted and let FRR_{fd} be the proportion of transactions with a real finger that are incorrectly rejected. In the following, the EER_{fd} (that is the value such that $FRR_{fd} = FAR_{fd}$) will be reported as a performance indicator. Note that FAR_{fd} and FRR_{fd} do not include verification/identification errors and must be combined with them to characterize the overall system errors.

In order to evaluate the proposed approach, a database of image sequences was collected. The database was acquired in the Biometric System Laboratory of the University of Bologna from 20 volunteers. Two fingers (thumb and forefinger of the right hand) were collected from each volunteer and two additional fingers (thumb and forefinger of the left hand) were collected from six of them; five image sequences were recorded for each finger. 12 fake fingers were manufacted (four made of RTV silicone, four of gelatine and four of latex) starting from fingers of three cooperating volunteers; five image sequences were recorded for each of them. The image sequences were acquired using the optical fingerprint scanner "TouchView II" by Identix, which produces 420×360 fingerprint images at 500 DPI. A Matrox Meteor frame grabber was used to acquire frames at 30 fps). The database was divided into two disjoint sets: a *validation set* (12 real fingers and 6 fake fingers) used for tuning the various parameters of the approach and a *test set* (40 real fingers and 6 fake fingers), used to measure the performance. The following transactions were performed on the test set:

- 400 genuine attempts (each sequence was matched against the remaining sequences of the same finger, excluding the symmetric matches to avoid correlation, thus performing 10 attempts for each of the 40 real fingers);
- 1200 impostor attempts (each of the 30 fake sequences was matched against the first sequence of each real finger). Note that, since only fake-detection performance was evaluated (not combined with identity verification) and considering that the proposed approach is based only on the elastic properties of real/fake fingers, it is

not necessary that a fake finger corresponding to the real finger is used in the impostor attempts: any fake finger can be matched against any real finger without adding any bias to the results.

The EER_{fd} of the proposed approach measured in the above described experimentation was 4.9%.

4 Conclusions and Future Work

We believe the results obtained are very promising: the method achieved a reasonable EER_{fd} (4.9%), proved to be very efficient (on a Pentium IV at 3.2Ghz, the average processing and matching time is less than eight ms) and not too annoying for the user (the whole fake-detection process, including the acquisition of the fingerprint sequence, takes about two seconds). The proposed approach has also the advantage of being software-based (i.e. no additional hardware is required to detect the fake fingers: the only requirement for the scanner is the capability of delivering frames at a proper rate). We are currently acquiring a larger database to perform additional experiments and investigating other alignment techniques for the DistortionCode sequences.

References

[1] N.K. Ratha, J.H. Connell, and R.M. Bolle, "An analysis of minutiae matching strength", Proc. AVBPA 2001, Third International Conference on Audio- and Video-Based Biometric Person Authentication, pp. 223-228, 2001.

[2] T Matsumoto, H. Matsumoto, K. Yamada, S. Hoshino, "Impact of Artificial 'Gummy' Fingers on Fingerprint Systems", Proceedings of SPIE, vol. 4677, January, 2002.

[3] T. Putte and J. Keuning, "Biometrical fingerprint recognition: don't get your fingers burned", Proc. IFIP TC8/WG8.8, pp. 289-303, 2000.

[4] Umut Uludag and Anil K. Jain, "Attacks on biometric systems: a case study in fingerprints", Proceedings of SPIE – v. 5306, Security, Steganography, and Watermarking of Multimedia Contents VI, June 2004, pp. 622-633.

[5] R. Derakhshani, S.A.C. Schuckers, L.A. Hornak, and L.O. Gorman, "Determination of vitality from a non-invasive biomedical measurement for use in fingerprint scanners", Pattern Recognition, vol. 36, pp. 383-396, 2003.

[6] PD Lapsley, JA Less, DF Pare, Jr., N Hoffman, "Anti-Fraud Biometric Sensor that Accurately Detects Blood Flow", SmartTouch, LLC, US Patent #5,737,439.

[7] D. Maltoni, D. Maio, A.K. Jain, and S. Prabhakar, Handbook of Fingerprint Recognition, Springer, 2003.

[8] R. Cappelli, D. Maio and D. Maltoni, "Modelling Plastic Distortion in Fingerprint Images", in proceedings 2nd International Conference on Advances in Pattern Recognition (ICAPR2001), Rio de Janeiro, March 2001, pp.369-376.

[9] A. K. Jain, S. Prabhakar and L. Hong, "A Multichannel Approach to Fingerprint Classification", in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 21 no. 4, April 1999, pp. 348-359.