

# A New Approach for Fingerprint Verification Based on Wide Baseline Matching Using Local Interest Points and Descriptors

Javier Ruiz-del-Solar, Patricio Loncomilla, and Christ Devia

Department of Electrical Engineering, Universidad de Chile  
{jruidz, ploncomi, cdevia}@ing.uchile.cl

**Abstract.** In this article is proposed a new approach to automatic fingerprint verification that is not based on the standard ridge-minutiae-based framework, but in a general-purpose wide baseline matching methodology. Instead of detecting and matching the standard structural features, in the proposed approach local interest points are detected in the fingerprint, then local descriptors are computed in the neighborhood of these points, and afterwards these descriptors are compared using local and global matching procedures. The final verification is carried out by a Bayes classifier. It is important to remark that the local interest points do not correspond to minutiae or singular points, but to local maxima in a scale-space representation of the fingerprint images. The proposed system has 4 variants that are validated using the FVC2004 test protocol. The best variant, which uses an enhanced fingerprint image, SDoG interest points and SIFT descriptors, achieves a FRR of 20.9% and a FAR of 5.7% in the FVC2004-DB1 test database, without using any minutia or singular points' information.

**Keywords:** Fingerprint verification, Wide Baseline Matching, SIFT.

## 1 Introduction

Fingerprint verification is one of the most employed biometric technologies. A fingerprint is the pattern of ridges and furrows on the surface of a fingertip. It is formed by the accumulation of dead, cornified cells [5]. The fingerprint pattern is unique and determined by the local ridge characteristics and the presence of ridge discontinuities, called minutiae. The two most prominent minutiae are ridge termination and ridge bifurcation. Minutiae in fingerprints are generally stable and robust to fingerprint impression conditions. Singular points, called loop and delta, are a sort of control points around which the ridge-lines are “wrapped” [12]. Many approaches to automatic fingerprint verification have been proposed in the literature and the research on this topic is still very active. In most of the cases the automatic verification process is based on the same procedure employed by human experts:

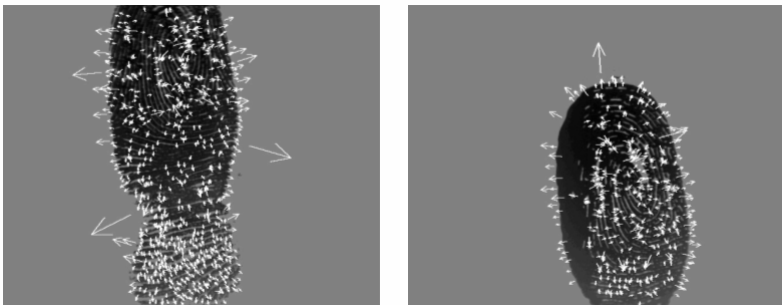
- (i) Detection of structural features (ridges, minutiae, and/or singular points), and in some cases derived features as the orientation field, which allow characterizing the fingerprints, and
- (ii) Comparison between the features in the input and reference fingerprints. This comparison is usually implemented using minutiae-based matching, ridge pattern

comparison and/or correlation between the fingerprints. The mentioned comparison methodologies can be described as [12]: *Minutiae-based matching*: It consists of finding the alignment between the input and the reference minutiae sets that results in the maximum number of minutiae pairings; *Ridge feature-based matching*: The approaches belonging to this family compare fingerprints in term of features extracted from the ridge pattern (e.g. local orientation and frequency, ridge shape, texture information); and *Correlation-based matching*: Two fingerprint images are superimposed and the correlation (at the intensity level) between corresponding pixels is computed for different alignments (e.g., various displacements and rotations).

In state of the art fingerprint verification systems several structural features and comparison methodologies are jointly employed. For instance, in the 2004 Fingerprint Verification Competition (FVC2004) the 29 participants (from 43) that provided algorithm's information employed the following methodologies [2]:

- Features: minutiae (27), orientation field (19), singular points (12), ridges (10), local ridge frequency (8), ridge counts (6), raw or enhanced image parts (4), and texture measures (3).
- Comparison methodology: minutiae global (20), minutiae local (15), correlation (7), ridge pattern geometry (5), and ridge pattern texture (2).

In this general context the main objective of this article is to propose a new approach to automatic fingerprint verification that it is not based on the standard ridge-minutiae-based framework, but in a general-purpose wide baseline matching methodology. Instead of detecting and matching the standard structural features, in the proposed approach local interest points are detected in the fingerprint, then local descriptors are computed in the neighborhood of these points, and finally these descriptors are compared using local and global matching procedures. The local interest points do not correspond to minutiae or singular points, but to local maxima in a scale-space representation of the fingerprint image (see examples in Figure 1).



**Fig. 1.** Example of detected local interest points in the test and template fingerprints. Interest points are displayed as arrows, whose origin, orientation and size corresponds to the position  $(x,y)$ , orientation  $q$  and scale  $s$  of the corresponding interest points.

The main intention in proposing this new approach is to show an alternative procedure for solving the fingerprint verification procedure. We believe that this new approach can complement and enrich the standard procedures, and it can be used in

addition to them. In this sense, we are not proposing a methodology for replacing the standard one, but a complementary solution.

This article is structured as follows. In section 2 the employed methodology for solving the general wide baseline matching problem is described. In section 3 is presented the adaptation of this methodology to fingerprint verification. In section 4 are presented preliminary results of this new approach for fingerprint verification. Finally, in section 5 some conclusions and projections of this work are given.

## 2 Wide Baseline Matching Using Local Interest Points and Descriptors

Wide baseline matching refers to a matching process where the images to be compared are allowed to be taken from widely separated viewpoints, so that a point in one image may have moved anywhere in the other image. Object recognition using a reference image (model) can be modeled as a wide baseline matching problem.

In the context, wide baseline matching (object recognition) approaches based on local interest points (invariant features) have become increasingly popular and have experienced an impressive development in the last years [3][7][8][13][17]. Typically, local interest points are extracted independently from both a test and a reference image, then characterized by invariant descriptors, and finally the descriptors are matched until a given transformation between the two images is obtained. Most employed local detectors are the Harris detector [4] and the Lowe's sDoG+Hessian detector [7], being the Lowe's detector multiscale and the Harris detector single scale. Best performing affine invariant detectors are the Harris-Affine and the Hessian-Affine [15], but they are too slow to be applied in general-purpose applications. The most popular and best performing invariant descriptor [14] is the SIFT (Scale Invariant Feature Transform) [7].

For selecting the local detector and invariant descriptor to be used in a given application it should be taken into account the algorithm's accuracy, robustness and processing speed. Lowe's system [7] using the SDoG+Hessian detector, SIFT descriptors and a probabilistic hypothesis rejection stage is a popular choice, given its recognition capabilities, and near real-time operation. However, Lowe's system main drawback is the large number of false positive detections. This is a serious problem when using it in real world applications as for example robot self-localization [19], robot head pose detection [9] or image alignment [20].

One of the main weaknesses of Lowe's algorithm is the use of just a simple probabilistic hypothesis rejection stage, which cannot successfully reduce the number of false positives. Loncomilla and Ruiz-del-Solar (L&R) propose a system that reduces largely the number of false positives by using several hypothesis rejection stages [8][9][10][11]. This includes a fast probabilistic hypothesis rejection stage, a linear correlation verification stage, a geometrical distortion verification stage, a pixel correlation verification stage, a transformation fusion procedure, and the use of the RANSAC algorithm and a semi-local constraints test. Although, RANSAC and the semi-local constraints test have been used by many authors, Lowe's system does not use them. In [10] are compared the Lowe's and the L&R systems using 100 pairs of real-world high-textured images (variations in position, view angle, image covering, partial occlusion, in-plane and out-of-the-plane rotation). The results show that in this dataset the L&R system reduces the

false positive rate from 85.5% to 3.74%, by increasing the detection rate by 5%. For this reason we choose to use this system in this work.

The L&R system considers four main stages: (i) *Generation of local interest points*, (ii) *Computation of the SIFT descriptors*, (iii) *SIFT-matching using nearest descriptors*, and (iv) *Transformation Computation and Hypothesis Rejection Tests*. The first three stages are the standard ones proposed by Lowe, while the fourth stage is employed for reducing the number of false matches, giving robustness to the whole system. This stage is implemented by the following procedure (detailed description in [8][9][10]):

1. Similarity transformations are determined using the Hough transform. After the Hough transform is computed, a set of bins, each one corresponding to a similarity transformation, is determined. Then:
  - a. Invalid bins (those that have less than 4 votes) are eliminated.
  - b.  $Q$  is defined as the set of all valid candidate bins, the ones not eliminated in 1.a.
  - c.  $R$  is defined as the set of all accepted bins. This set is initialized as a void set.
2. For each bin  $B$  in  $Q$  the following tests are applied (the procedure is optimized for obtaining high processing speed by applying less time consuming tests first):
  - a. If the bin  $B$  has a direct neighbor in the Hough space with more votes, then delete bin  $B$  from  $Q$  and go to 2.
  - b. Calculate  $r_{REF}$  and  $r_{TEST}$ , which are the linear correlation coefficients of the interest points corresponding to the matches in  $B$ , that belong to the reference and test image, respectively. If the absolute value of any of these two coefficients is high, means that the corresponding points lie, or nearly lie, in a straight line, and that the affine transform to be obtained can be numerically unstable. If this condition is fulfilled delete bin  $B$  from  $Q$  and go to 2.
  - c. Calculate the probability  $P_{FAST}$  associated to  $B$ . If  $P_{FAST}$  is lower than a threshold  $P_{TH1}$ , delete bin  $B$  from  $Q$  and go to 2. The main advantage of this probability test is that it can be computed before calculating the affine transformation, which speeds up the whole procedure.
  - d. Calculate an initial affine transformation  $T_B$  using the matches in  $B$ .
  - e. Compute the affine distortion degree of  $T_B$  using a geometrical distortion verification test. A certain affine transformation should not deform very much an object when mapping it. Therefore, if  $T_B$  has a strong affine distortion, delete bin  $B$  from  $Q$  and go to 2.
  - f. Top down matching: Matches from all the bins in  $Q$  that are compatible with the affine transformation  $T_B$  are summarized and added to bin  $B$ . Duplication of matches inside  $B$  is avoided.
  - g. Compute the Lowe's probability  $P_{LOWE}$  of bin  $B$ . If  $P_{LOWE}$  is lower than a threshold  $P_{TH2}$ , delete bin  $B$  from  $Q$  and go to 2.
  - h. To find a more precise transformation apply RANSAC inside bin  $B$ . In case that RANSAC success, a new transformation  $T_B$  is calculated and  $B$  is labeled as a RANSAC-approved bin.
  - i. Accept the candidates  $B$  and  $T_B$ , what means delete  $B$  from  $Q$  and include it in  $R$  (the  $T_B$  transformation is accepted).

3. For all pairs  $(B_i, B_j)$  in  $R$ , check if they may be fused into a new bin  $B_k$ . If the bins may be fused and one of them is RANSAC-approved, do not fuse them, and delete the other in order to preserve accuracy. If the two bins are RANSAC-approved, delete the least probable. Repeat this until all possible pairs (including the new created bins) have been checked.
4. For any bin  $B$  in  $R$ , apply the semi-local constraints procedure to all matches in  $B$ . The matches from  $B$  who are incompatible with the constraints are deleted. If some matches are deleted from  $B$ ,  $TB$  is recalculated.
5. For any bin  $B$  in  $R$ , calculate the pixel correlation  $r_{pixel}$  using  $TB$ . Pixel correlation is a measure of how similar are the image regions being mapped by  $TB$ . If  $r_{pixel}$  is below a given threshold, delete  $B$  from  $R$ .
6. Assign a priority to all bins (transformations) in  $R$ . The initial priority value of a given bin will correspond to its associated  $P_{LOWE}$  probability value. In case that the bin is a RANSAC-approved one, the priority is increased in one. Thus, RANSAC-approved bins have a larger priority than non RANSAC-approved ones.

### 3 Proposed System for Fingerprint Verification

The proposed system for fingerprint verification is based on the L&R wide baseline matching system described in the former section. However, for applying this system in real world harsh conditions (state of the art fingerprint testing protocols) two main improvements are included: (i) a fingerprint enhancement pre-processing module, and (ii) a statistical classification post-processing module. In addition an optional module that computes minutiae-based SIFT descriptors is also included in the system, for studying how the minutiae information affects the performance of the system. Figure 2 (d) shows a block diagram of the proposed fingerprint verification system. In the next paragraphs we analyze the fingerprint verification process using the L&R system, and we describe the new processing modules.

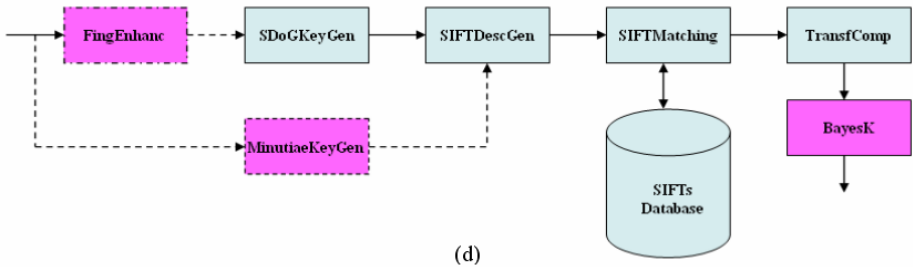
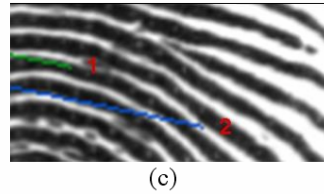
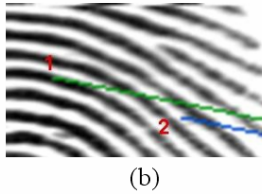
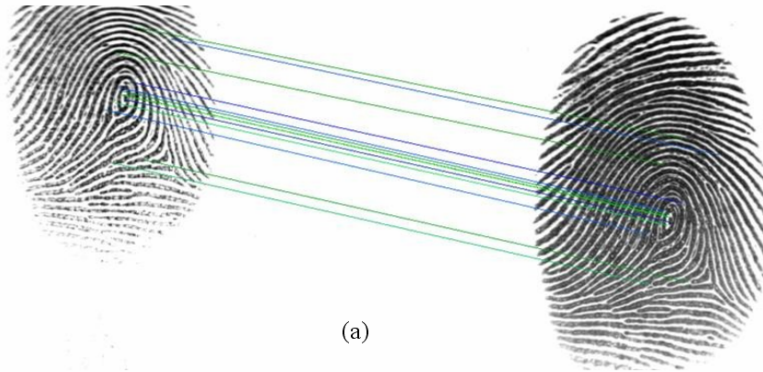
#### 3.1 Fingerprint Verification Analysis

We performed several fingerprint analysis experiments using different fingerprint public databases, and we verified that the L&R wide baseline system allows matching fingerprints. In Figure 2 we show an exemplar experimental result. In Figure 2 (a) can be observed a fingerprints reference-test image pair with the corresponding correct matches. As it can be seen, the wide baseline system achieves matching correctly the two fingerprints. In Figures 2 (b)-(c) are shown some selected matched local interest points. It can be verified that the local interest points do not correspond to minutiae or singular points. As mentioned they correspond to local maxima in the position-scale multi-resolution representation of the fingerprint images.

One of main problems in fingerprint verification is the nonlinear distortion in fingerprint images, which disturbs the matching process [16]. This problem is tackled by limiting the acceptable distortion in the acquisition process, by estimating the

distortion during the matching process [21][1], or by compensating the distortion using fingerprint warping ([16]).

The here-proposed wide baseline matching approach is robust against nonlinear distortions in fingerprint images. It can be proved that any differentiable non-linear transformation that is locally near-orthogonal can be approximated by a bundle of local similarity approximations using the Hough transform methodology, if the density of matches between interest points is enough high (see Appendix).



**Fig. 2.** (a) Fingerprints reference-test image pairs with matches. (b)-(c) Matched local interest points. (d) Block diagram of the proposed L&R system. Dashed lines represent optional modules. Modules in pink are proposed in this work. *Fingenhanc*: Fingerprint Enhancement; *SDoG-KeyGen*: SDoG Keypoints Generation; *MinutiaeKeyGen*: Minutiae Keypoints Generation; *SIFTDescGen*: SIFT Descriptors Generation; *TrasfComp*: Transformation Computation and hypothesis rejection tests; *BayesK*: bayes Classifier.

### 3.2 Fingerprint Enhancement

Due to their low quality, fingerprint images can be optionally enhanced before applying the wide baseline recognition system. The fingerprint is divided into a two-dimensional array of squared regions, and then the local orientation and local frequency are calculated in each region to get a pair of orientation and frequency fields over the complete image. Finally, the pair of fields is used to generate a bank of real-valued Gabor filters, which is applied to the image to enhance it. When this pre-processing module is employed the keypoints are named SDoG-enhanced (SDoG-E) keypoints, otherwise SDoG-non-enhanced (SDoG-NE). To implement this stage the open FVS library is used [24].

### 3.3 Generation of Minutia Keypoints for Each Image

Minutia keypoints are searched over a (Gabor-) enhanced, binarized and thinned version of the input image. The local orientation for each minutia keypoint is obtained from the orientation field calculated, while the local scale for the minutia keypoint is proportional to the inverse of the frequency field. Minutia keypoints are finally characterized by a 4-dimensional vector  $(x, y, \sigma, \theta)$ , which fixes the position, scale and orientation of the minutia keypoint. As both SDoG keypoints and minutia keypoints are described as 4-dimensional vectors, they can be used alone or in an integrated fashion. To obtain the minutiae set, the open FVS library is used [24].

### 3.4 Bayes Classification

After applying the L&R methodology to verification problems in which the fingerprints quality changes largely (changes in finger position, orientation, pressure, skin distortion, etc.), we noted that the number of false positives was very large. We solve this problem by applying a statistical classifier (Naïve Bayes) after the wide baseline matching original system. We defined the following 12 features for the classifier (see details in section 2):

1. *TNMatches*: Total number of matches between the reference and test image.
2. *PTime*: Processing time as a measure of the complexity of the matching process.
3. *NAffinT*: Number of detected affine transformations between reference and test image.
4. *NMatches*: Number of associated matches in the best transformation.
5. *PBT*: Probability of the best transformation ( $P_{LOWE}$ ).
6. *LCorr*: Linear correlation of the best transformation ( $r_{REF}$ ).
7. *PCorr*: Pixel correlation of the best transformation ( $r_{pixel}$ ).
8. *MNDesc*: Maximum number of test image descriptors who are matched to the same reference image descriptor, considering the best transformation.
9. *ScaleAffinMax*: Absolute value of the upper eigenvalue of the affine transformation matrix of the best transformation (i.e., upper scale of the best affine transformation).

10. *ScaleAffinMin*: Absolute value of the lower eigenvalue of the affine transformation matrix of the best transformation (i.e., lower scale of the best affine transformation).
11. *NIncMatches*: Number of matches of the best transformation who are incompatible with the semi-local constraints.
12. *RansacPar*: RANSAC-compatibility with precise affine transform. A small subset of 3 matches from  $B$  is selected to construct a transformation  $TB$  who is tested against all the resting matches in  $B$ . The matches from  $B$  who are correctly mapped using  $TB$  with a very little error are called compatible matches. If more than  $2/3$  of the matches from  $B$  are compatible, the incompatible matches are deleted and a final fine transformation is calculated using the compatible matches. This procedure is tried 10 times using different subsets with 3 matches from  $B$ . If in none of the 10 iterations a fine transformation is obtained, RANSAC fails and the initial transformation is preserved.

We analyzed the relevance of these features using the Weka package [22]. With the Weka's *BestFirst* attributes selection method, which searches the space of attribute subsets by greedy hill-climbing, we selected the final attributes that we use in each of our experiments (see table 1).

## 4 Preliminary Results

We present some preliminary results of the operation of the proposed system for fingerprint verification. We test different flavors of the system. Although this system validation is preliminary we choose to use a state of the art fingerprint database. We select to use the DB1 database from FVC2004. According to the FVC2004 test results, DB1 has proven to be very difficult compared to DB2, DB3 and DB4 [2], mainly because of the presence of a large number of distorted fingerprints (skin distortion was encouraged during some acquisition sessions).

The main characteristics of DB1 are: acquisition using an optical scanner; 120 fingers: 30 persons, 4 fingers per person; 12 impressions per finger: 3 collection sessions, in each session 4 impressions were acquired per finger; and database divided in DB1-train (10 fingers) and DB1-test (100 fingers). The collection sessions encourage variations in the obtained impressions; session 1: changes in finger position and pressure, session 2: skin distortion and variations in rotation, session 3: fingers were dried and moistened.

The FVC 2004 test protocol was followed (see details in Cappelli et al. [2]), and Genuine recognition attempts (GNA) and Impostor recognition attempts (IRA) sets for the DB1-train and DB1-test databases were built. The Naïve Bayes classifier of the proposed fingerprint verification system was trained using the Weka package [22] and 10-fold cross-validation.

The training of the classifier was performed using DB1-train. Several tests were executed over the DB1-test database using different flavors of the proposed system. These flavors were obtained using different keypoint-generators (SDoG-E: SDoG-enhanced; SDoG-NE: SDoG-non-enhanced; or minutia), and SIFT descriptors of different sizes (small: 4x4 region's size; medium: 5x5 region's size; or large: 30x30 region's size). To give a short name for a given descriptor-generator, the notation



G@XxY will be used; G represents the keypoint generator, and XxY represents the region’s size of the associated SIFT descriptor. If several descriptor-generators are used simultaneously, the combined generator notation will be expressed as the sum of the individual generators notation. The features selected for training the classifiers depends on the kind of keypoints and descriptors been employed (see table 1).

The results obtained from the tests (see table 2) show that the SDoG-E@30x30 (large size descriptors, enhanced image, no minutiae) is the flavor that produces the best TP v/s FP pair. As several small, local regions of fingerprints can look similar, a local symmetry problem exists in any fingerprint image. The use of large regions in the SIFT descriptor calculation helps to break the local symmetry observed in the fingerprint, and then helps to produce more distinctive descriptors. The image enhancement process before the SDoG keypoints calculation, which is a novelty in general-purpose wide-baseline methods, helps to remove acquisition noise, giving a very repeatable fingerprint image, which produces more repeatable keypoints and descriptors, improving the verification results.

The Bayesian classifier helps to discard a great amount of false detections, which are produced by the fingerprint local symmetry problem mentioned above. This can be illustrated showing that, when using the classifier, the verification results obtained on the DB1 database are TP=79.1% and FP=5.7%, while when not using it are TP=98.5% and FP=73.29%. Thus, the main effect of the classifier is to reduce largely the number of false detections.

**Table 1.** Selected Features for classifier training

Method	Selected Features
SDoG-E@30x30	PTime, TNMatches, NMatches, PCorr
SDoG-NE@30x30	PTime, TNMatches, NMatches, PCorr
SDoG-E@5x5	PTime, TNMatches, NAffinT, NMatches, PCorr, MNDesc, ScaleAffinMax, NIncMatches
SDoG-NE@5x5	PTime, TNMatches, NMatches, PCorr
Minutia@30x30 + SDoG-E@30x30	PTime, TNMatches, NAffinT, NMatches, PCorr
Minutia@4x4 + SDoG-E@5x5	PTime, TNMatches, NAffinT, NMatches, MNDesc, ScaleAffinMax, ScaleAffinMin, NIncMatches.
Minutia@4x4	All the 12 characteristics

**Table 2.** Recognition statistics over the DB1-Test database

Method	TP% (100-FRR)	FP% (FAR)
SDoG-E@30x30	79.1	5.7
SDoG-NE@30x30	61.0	16.7
SDoG-E@5x5	77.5	18.4
SDoG-NE@5x5	60.9	31.6
Minutia@30x30 + SDoG-E@30x30	69.6	17.3
Minutia@4x4 + SDoG-E@5x5	83.4	24.5
Minutia@4x4	57.7	10.6

The confrontation of SDoG v/s Minutia + SDoG methods (see Table 2) shows that the direct addition of minutia information to the interest point information before the Hough transform does not help effectively to the matching process. Thus, a future alternative to test is to mix SDoG keypoints information and minutia keypoints information in a more smart way, for example, to use only SDoG keypoints to detect fingerprints using the Hough transform methodology, and then to use minutia information as a posterior verification stage.

When comparing the obtained results with the ones from state of the art systems participating in the FVC2004 competition (systems developed by research institutions working for years in fingerprint verification), we observe that our system could have achieved the top30 position. In the FVC2004 report the results from all the participants are ordered by EER (equal error rate). Our SDoG-E@30x30 test does not get an EER value, but two error values: a FRR=20.9% and a FAR=5.7%, which corresponds to our operational point. We compared our operational point with the ROC curves from the competitors, and we found that the top30 participant has the ROC curve that is nearest to our operational point. We believe that this result is very promising because our approach is the first one that solves a fingerprint verification problem using a general-purpose wide-baseline method, and it can still be improved, extended and tuned for achieving state of the art results.

One of the very interesting results of the application of the proposed algorithm is that it could process the whole DB1 database without any software failure. FVC2004 test developers implement a special treatment of failures during tests [2], because usually the compared verification systems can fail during the enrollment or the verification process. This situation was observed in the FVC2000, FVC2002 and FVC2004 tests, but not in the testing of our system.

## 5 Conclusions

A new approach to automatic fingerprint verification based in a general-purpose wide baseline matching methodology was proposed. Instead of detecting and matching the standard structural features, in the proposed approach local interest points are detected in the fingerprint, then local descriptors are computed in the neighborhood of these points, and finally these descriptors are matched. Image enhancement, several verification stages, and a simple statistical classifier are employed for reducing the number of false positives. The nature of the interest points permits to integrate them with minutia points, but a useful way to integrate both information sources is been investigated.

The proposed fingerprint verification system was validated using the FVC2004 test protocol. Without using any a priori knowledge of the finger minutia and singular points information, the system achieves, in the FVC2004-DB1 database, a FAR of 5.7% and a FRR of 20.9%. We expect to improve these results with a better integration of the minutiae-derived descriptors.

## Acknowledgements

This research was funded by Millenium Nucleus Center for Web Research, Grant P04-067-F, Chile.

## References

1. Bazen, A.M., Gerez, S.H.: Systematic methods for the computation of the directional fields and singular points of fingerprints. *IEEE Trans. Pattern Anal. Machine Intell* 24(7), 905–919 (2002)
2. Cappelli, R., Maio, D., Maltoni, D., Wayman, J., Jain, A.K.: Performance evaluation of fingerprint verification systems. *IEEE Trans. Pattern Anal. Machine Intell* 28(1), 3–18 (2006)
3. Ferrari, V., Tuytelaars, T., Van Gool, L.: Simultaneous Object Recognition and Segmentation by Image Exploration. In: Pajdla, T., Matas, J(G.) (eds.) *ECCV 2004*. LNCS, vol. 3021, pp. 40–54. Springer, Heidelberg (2004)
4. Harris, C., Stephens, M.: A combined corner and edge detector. In: *Proc. 4th Alvey Vision Conf.*, Manchester, UK, pp. 147–151 (1998)
5. Lee, H.C., Lee, H.C.G.: *Advances in Fingerprint Tecnology*. Elsevier, NY (1991)
6. Lowe, D.: Local feature view clustering for 3D object recognition. In: Lowe, D. (ed.) *IEEE Conference on Computer Vision and Pattern Recognition*, Hawaii, pp. 682–688 (2001)
7. Lowe, D.: Distinctive Image Features from Scale-Invariant Keypoints. *Int. Journal of Computer Vision* 60(2), 91–110 (2004)
8. Loncomilla, P., Ruiz-del-Solar, J.: Gaze Direction Determination of Opponents and Teammates in Robot Soccer. In: Bredenfeld, A., Jacoff, A., Noda, I., Takahashi, Y. (eds.) *RoboCup 2005*. LNCS (LNAI), vol. 4020, pp. 230–242. Springer, Heidelberg (2006)
9. Loncomilla, P., Ruiz-del-Solar, J.: Improving SIFT-based Object Recognition for Robot Applications. In: Roli, F., Vitulano, S. (eds.) *ICIAP 2005*. LNCS, vol. 3617, pp. 1084–1092. Springer, Heidelberg (2005)
10. Loncomilla, P., Ruiz-del-Solar, J.: A Fast Probabilistic Model for Hypothesis Rejection in SIFT-Based Object Recognition. In: Martínez-Trinidad, J.F., Carrasco Ochoa, J.A., Kittler, J. (eds.) *CIARP 2006*. LNCS, vol. 4225, pp. 696–705. Springer, Heidelberg (2006)
11. Ruiz-del-Solar, J., Loncomilla, P., Vallejos, P.: An Automated Refereeing and Analysis Tool for the Four-Legged League. LNCS (2006)
12. Maltoni, D., Maio, D., Jain, A.K., Prabhakar, S.: *Handbook of Fingerprint Recognition*. Springer, Heidelberg (2003)
13. Mikolajczyk, K., Schmid, C.: Scale & Affine Invariant Interest Point Detectors. *Int. Journal of Computer Vision* 60(1), 63–96 (2004)
14. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal. Machine Intell* 27(10), 1615–1630 (2005)
15. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Van Gool, L.: A Comparison of Affine Region Detectors. *Int. Journal of Computer Vision* 65(1-2), 43–72 (2005)
16. Ross, A., Dass, S., Jain, A.K.: Fingerprint warping using ridge curve correspondences. *IEEE Trans. Pattern Anal. Machine Intell* 28(1), 19–30 (2006)
17. Schaffalitzky, F., Zisserman, A.: Automated location matching in movies. *Computer Vision and Image Understanding* 92(2-3), 236–264 (2003)
18. Schmid, C., Mohr, R.: Local grayvalue invariants for image retrieval. *IEEE Trans. Pattern Anal. Machine Intell* 19(5), 530–534 (1997)
19. Se, S., Lowe, D., Little, J.: Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *Int. Journal of Robotics Research* 21(8), 735–758 (2002)
20. Vallejos, P.: Detection and tracking of people and objects in motion using mobile cameras. Master Thesis, Universidad de Chile (2007)

21. Kovacs-Vajna, Z.M.: A fingerprint verification system based on triangular matching and dynamic time warping. *IEEE Trans. Pattern Anal. Machine Intel* 22(11), 1266–1276 (2000)
22. Weka: Weka website. Available in November 2006 in:  
<http://www.cs.waikato.ac.nz/ml/weka/>
23. FVS 2003, Fingerprint Verification System, <http://fvs.sourceforge.net>

## Appendix: About the Robustness of Similarity Hough Transform for the Detection of Nonlinear Transformations

Consider two rectangular planar surfaces, named  $S_{REF}$  and  $S_{TEST}$ . In both, let us define a two-dimensional rectangular coordinate system  $(x_R, y_R)$  for  $S_{REF}$  and  $(x_T, y_T)$  for  $S_{TEST}$ . Additionally, we define two functions: one function  $I_{REF}$  from  $S_{REF}$  to  $\mathbb{R}$ , which is named the reference image, and another function  $I_{TEST}$  from  $S_{TEST}$  to  $\mathbb{R}$  named the test image. A similarity transformation that maps the reference surface  $S_{REF}$  into the test surface  $S_{TEST}$  has the following expression:

$$\begin{pmatrix} x_T \\ y_T \end{pmatrix} = e \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} x_R \\ y_R \end{pmatrix} + \begin{pmatrix} t_X \\ t_Y \end{pmatrix} \quad (1)$$

As the similarity transformation has 4 parameters, there must be 4 constraints to fix a transformation, i.e. 4 scalar equations relating the parameters. There is a collection of matches between pairs of scale-invariant interest points. Each pair consists of a point located in the reference image and another point located in the test image. Each point has a position  $(x, y)$ , orientation  $\theta$  and scale  $\sigma$ . Any scale-invariant interest point can be described by an arrow with two relevant points, which are the origin and the head of the arrow. If the interest point is described by the information  $(x, y, \sigma, \theta)$ , the origin of the associated arrow is described by the point  $(x, y)$ , while the head of the arrow is separated from the origin by a distance  $\sigma$  in the direction  $\theta$ . Given that any interest point can be represented as an arrow, any match between two interest points can be thought as a match between two arrows. The match of two arrows can be considered as two point-matches: the match of the origins of the arrow, and the match of the heads of the arrow. These two-point matches correspond to 2 vector equations which relate the parameters of the similarity transformation and determinate the transformation in a unique way:

$$\begin{pmatrix} x_{TEST} \\ y_{TEST} \end{pmatrix} = e \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} x_{REF} \\ y_{REF} \end{pmatrix} + \begin{pmatrix} t_X \\ t_Y \end{pmatrix} \quad (2)$$

$$\begin{pmatrix} x_{TEST} + \sigma_{TEST} \cos(\theta_{TEST}) \\ y_{TEST} - \sigma_{TEST} \sin(\theta_{TEST}) \end{pmatrix} = e \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} x_{REF} + \sigma_{REF} \cos(\theta_{REF}) \\ y_{REF} - \sigma_{REF} \sin(\theta_{REF}) \end{pmatrix} + \begin{pmatrix} t_X \\ t_Y \end{pmatrix} \quad (3)$$

The solution to these equations is the vector of 4 parameters  $(e, \theta, t_X, t_Y)$  of the transformation, which depends of the information of the match between two interest points  $(x_{REF}, y_{REF}, \theta_{REF}, \sigma_{REF})$  and  $(x_{TEST}, y_{TEST}, \theta_{TEST}, \sigma_{TEST})$ . In other words, each match

between two scale-invariant interest points generates a set of parameters  $(e, \theta, t_X, t_Y)$  for a similarity transformation.

$$\begin{aligned}
 \theta &= \theta_{TEST} - \theta_{REF} \\
 e &= \sigma_{TEST} / \sigma_{REF} \\
 t_X &= x_{TEST} - e(x_{REF} \cos \theta + y_{REF} \sin \theta) \\
 t_Y &= y_{TEST} - e(-x_{REF} \sin \theta + y_{REF} \cos \theta)
 \end{aligned}
 \tag{4}$$

A Hough transform is used to count the number of times a set of parameters  $(e, \theta, t_X, t_Y)$  is voted by a match. The parameters space is quantized into bins using 4 indexes  $(i, j, k, z)$ . Each bin has a width of 1/4 of the reference image’s projection size in translation dimensions,  $30^\circ$  in orientation dimension, and a factor of 2 in scale dimension. If  $L_X$  is the width of the reference image and  $L_Y$  is its height, then the following expressions show the parameters’ quantization.

$$\begin{aligned}
 \theta &= 30^\circ k \\
 e &= 2^z \\
 t_X &= -\frac{1}{4} 2^z \cos(30^\circ k) L_X i - \frac{1}{4} 2^z \sin(30^\circ k) L_Y j \\
 t_Y &= \frac{1}{4} 2^z \sin(30^\circ k) L_X i - \frac{1}{4} 2^z \cos(30^\circ k) L_Y j
 \end{aligned}
 \tag{5}$$

Then, each bin has an associated central similarity transformation, which is calculated with the central parameters of the bin:

$$T_{i,j,k,z}(x, y) = 2^z \begin{pmatrix} \cos(30^\circ k) & \sin(30^\circ k) \\ -\sin(30^\circ k) & \cos(30^\circ k) \end{pmatrix} \begin{pmatrix} x - \frac{1}{4} L_X i \\ y - \frac{1}{4} L_Y j \end{pmatrix}
 \tag{6}$$

When a match between points  $(x_{REF}, y_{REF}, \theta_{REF}, \sigma_{REF})$  and  $(x_{TEST}, y_{TEST}, \theta_{TEST}, \sigma_{TEST})$  succeeds, a vote will be accumulated in the 16 more nearest integer values for  $(i, j, k, z)$ . In particular, each match will vote for the nearest  $(i, j, k, z)$  in the Hough transform. If the transform is differentiable, it can be expanded as a second order polynomial in the vicinity of any point.

$$(u, v)^T = T_{(x,y)} = T_{(x_0,y_0)} + J_{(x_0,y_0)}(x - x_0, y - y_0)^T + O(\Delta^2)
 \tag{7}$$

Let us define a transformation as locally near-orthogonal if the perpendicular angles in the image are almost not modified by the application of the transformation, i.e. it preserves locally the perpendicularity of lines in the vicinity of any point in an approximated way. A transformation which is locally near-orthogonal is like a similarity transformation in the vicinity of any point, i.e. it produces a different translation, rotation and scale change in a small vicinity of any point. Let us assume that the transformation  $T$  is locally near-orthogonal, i.e. it is like an approximation of a different similarity transformation in any point. Then, the following expression stands with small epsilons in all the space:

$$J_{(x_0,y_0)} = e_{(x_0,y_0)} \begin{pmatrix} \cos(\theta_{0(x_0,y_0)}) + \varepsilon_{1(x_0,y_0)} & \sin(\theta_{0(x_0,y_0)}) + \varepsilon_{2(x_0,y_0)} \\ -\sin(\theta_{0(x_0,y_0)}) + \varepsilon_{3(x_0,y_0)} & \cos(\theta_{0(x_0,y_0)}) + \varepsilon_{4(x_0,y_0)} \end{pmatrix} \quad (8)$$

In other words, if the Jacobian  $J$  of the transformation includes only a local rotation and a local scale change, the transformation  $T$  produces approximately a local translation  $T(x, y)$ , a local rotation  $\theta_{0(x,y)}$  and a local scale change  $e_{(x,y)}$  in the vicinity of any point  $(x, y)$ . This local translation, rotation and scale change are properties of the transformation  $T$ . Then, if a locally near-orthogonal transformation is applied to the reference image, any interest point  $(x_{REF}, y_{REF}, \theta_{REF}, \sigma_{REF})$  in that image transforms approximately in the following way:

$$\begin{aligned} (x_{REF}, y_{REF}, \theta_{REF}, \sigma_{REF}) &\rightarrow (x_{TEST}, y_{TEST}, \theta_{TEST}, \sigma_{TEST}) \\ (x_{TEST}, y_{TEST}) &= T(x_{REF}, y_{REF}) \\ \theta_{TEST} &= \theta_{REF} + \theta_{0(x_{REF}, y_{REF})} \\ \sigma_{TEST} &= \sigma_{REF} e_{(x_{REF}, y_{REF})} \end{aligned} \quad (9)$$

It can be noted that, when the transformation is locally near-orthogonal, all the points  $(x_{REF}, y_{REF}, \theta_{REF}, \sigma_{REF})$  transforms locally as in the similarity case. Then, the associated indexes  $(i, j, k, z)$  in a vote on the Hough transform originated from a point  $(x_{REF}, y_{REF})$  depend only on the properties of the transformation  $T(x, y)$ , i.e. the local translation, rotation and scale change defines the  $(i, j, k, z)$  for which to vote on each point  $(x_{REF}, y_{REF})$ . Let us define the function  $vote_{(x_{REF}, y_{REF})}$  as the function that returns the nearest integers  $(i, j, k, z)$  to vote for. Let us define too the set  $E_{(x_{REF}, y_{REF})}$  which depends on  $T(x, y)$  and  $(x_{REF}, y_{REF})$  as:

$$E_{(x_{REF}, y_{REF})} = \{ (x, y) \in S_{REF}^O \mid vote_{(x,y)} = vote_{(x_{REF}, y_{REF})} \} \quad (10)$$

$S_{REF}^O = \text{domain of } T(x, y) \text{ over } S_{REF} \text{ without border}$

In other words,  $E_{(x_{REF}, y_{REF})}$  is the set of all points in  $S_{REF}^O$  that votes for the same bin  $(i, j, k, z)$  as  $(x_{REF}, y_{REF})$ . If the transformation is differentiable (i.e. its Jacobian exists and is continuous in all  $S_{REF}^O$ ), then  $E_{(x_{REF}, y_{REF})}$  must include a connected maximal-size vicinity  $V_{(x_{REF}, y_{REF})}$  that includes  $(x_{REF}, y_{REF})$ . As any  $(x_{REF}, y_{REF})$  in  $S_{REF}^O$  must have its corresponding  $V_{(x_{REF}, y_{REF})}$ , it can be concluded that a  $\{V_k, k = 1, \dots, N\}$  partition of  $S_{REF}^O$  can be created. Any of the  $V_k$  with 4 or more matches will produce a detection in Hough space, and then a local similarity approximation will be computed. Then, it is concluded that any differentiable non-linear transformation that is locally near-orthogonal can be approximated by a bundle of local similarity approximations using the Hough transform methodology if several of the existent  $V_k$  have 4 or more votes, i.e., when the density of matches between interest points is enough high.

If the transformation is locally near-orthogonal and the nonlinearities are enough weak, the surface's partition will have only one vicinity  $V_1 = S_{REF}^O$ . Then, only one similarity detection will occur and a simple transformation (as an example, an affine transformation) is enough to approximate the true transformation in all the space  $S_{REF}$ .