Performance Evaluation of Fingerprint Verification Systems

Raffaele Cappelli, Dario Maio, *Member*, *IEEE*, Davide Maltoni, *Member*, *IEEE*, James L. Wayman, and Anil K. Jain, *Fellow*, *IEEE*

Abstract—This paper is concerned with the performance evaluation of fingerprint verification systems. After an initial classification of biometric testing initiatives, we explore both the theoretical and practical issues related to performance evaluation by presenting the outcome of the recent Fingerprint Verification Competition (FVC2004). FVC2004 was organized by the authors of this work for the purpose of assessing the state-of-the-art in this challenging pattern recognition application and making available a new common benchmark for an unambiguous comparison of fingerprint-based biometric systems. FVC2004 is an independent, strongly supervised evaluation performed at the evaluators' site on evaluators' hardware. This allowed the test to be completely controlled and the computation times of different algorithms to be fairly compared. The experience and feedback received from previous, similar competitions (FVC2000 and FVC2002) allowed us to improve the organization and methodology of FVC2004 and to capture the attention of a significantly higher number of academic and commercial organizations (67 algorithms were submitted for FVC2004). A new, "Light" competition category was included to estimate the loss of matching performance caused by imposing computational constraints. This paper discusses data collection and testing protocols, and includes a detailed analysis of the results. We introduce a simple but effective method for comparing algorithms at the score level, allowing us to isolate difficult cases (images) and to study error correlations and algorithm "fusion." The huge amount of information obtained, including a structured classification of the submitted algorithms on the basis of their features, makes it possible to better understand how current fingerprint recognition systems work and to delineate useful research directions for the future.

Index Terms—Biometric systems, fingerprint verification, performance evaluation, technology evaluation, FVC.

1 Introduction

The increasing demand for reliable human identification in large-scale government and civil applications has boosted interest in the controlled, scientific testing and evaluation of biometric systems. Just a few years ago, both the scientific community and commercial organizations were reporting performance results based on self-collected databases and ad hoc testing protocols, thus leading to incomparable and often meaningless results. Current scientific papers on fingerprint recognition now regularly report results using the publicly-available databases collected in our previous competitions [17], [18].

Fortunately, controlled, scientific testing initiatives are not limited within the biometrics community to fingerprint recognition. Other biometric modalities have been the target of excellent evaluation efforts as well. The (US) National Institute of Standards and Technology (NIST) has sponsored scientifically-controlled tests of text-independent speaker recognition algorithms [22], [25] for a number of years and, more recently, of facial recognition technologies as well [10].

- *R. Cappelli, D. Maio, and D. Maltoni are with the Biometric System Laboratory-DEIS, University of Bologna, via Sacchi 3, 47023 Cesena, Italy. E-mail: {cappelli, maio, maltoni}@csr.unibo.it.*
- *J.L. Wayman is with the Biometric Research Center, Office of Graduate Studies and Research, San Jose State University, San Jose, CA 95192-0025. E-mail: jlwayman@aol.com.*
- *A.K. Jain is with the Pattern Recognition and Image Processing Laboratory, Michigan State University, East Lansing, MI 48824. E-mail: jain@cse.msu.edu.*

Manuscript received 10 Jan. 2005; accepted 13 May 2005; published online 11 Nov. 2005.
Recommended for acceptance by H. Wechsler.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0021-0105.

NIST and others have suggested [28], [31] that biometric testing can be classified into "technology," "scenario," and "operational" evaluations. "Technology" evaluations test computer algorithms with archived biometric data collected using a "universal" (algorithm-independent) sensor; "Scenario" evaluations test biometric systems placed in a controlled, volunteer-user environment modeled on a proposed application; "Operational" evaluations attempt to analyze performance of biometric systems placed into real applications. Tests can also be characterized as "online" or "offline," depending upon whether the test computations are conducted in the presence of the human user (online) or after-the-fact on stored data (offline). An offline test requires a precollected database of samples and makes it possible to reproduce the test and to evaluate different algorithms under identical conditions.

We propose a taxonomy of offline tests with the following classifications (Fig. 1):

- *In-house—self-defined test*: The database is internally collected and the testing protocol is self-defined. Generally, the database is not publicly released, perhaps because of human-subject privacy concerns, and the protocols are not completely explained. As a consequence, results may not be comparable across such tests or reproducible by a third party.
- *In-house—existing benchmark*: The test is performed over a publicly available database, according to an existing protocol. Results are comparable with others obtained using the same protocol on the same database. Besides the trustworthiness problem,[1] the

1. Judging one's own work is hard and judging it dispassionately is impossible.

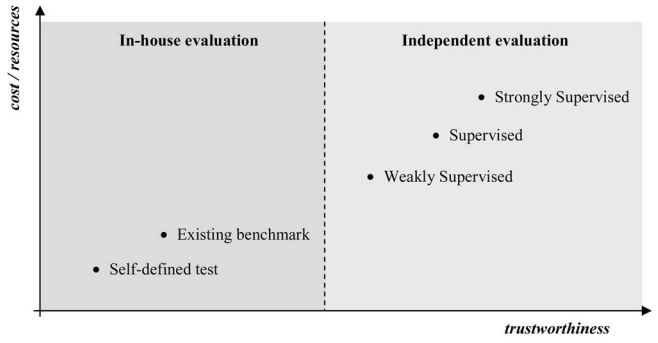


Fig. 1. Classification of offline biometric evaluations.

main drawback is the risk of overfitting the data—that is, tuning the parameters of the algorithms to match only the data specific to this test. In fact, even if the protocol defines disjoint training, validation, and test sets, the entire evaluation (including learning) might be repeated a number of times to improve performance over the final test set. Examples of recent biometric evaluations of this type are [23] and [24].

- *Independent—weakly supervised*: The database is sequestered and is made available just before the beginning of the test. Samples are unlabeled (the filename does not carry information about the sample's owner identity). The test is executed at the testee's site and must be concluded within given time constraints. Results are determined by the evaluator from the comparison scores obtained by the testee during the test. The main criticism against this kind of evaluation is that it cannot prevent human intervention: visual inspection of the samples, result editing, etc., could, in principle, be carried out with sufficient resources. Examples of recent biometric evaluations of this type are: [29], [22], and [9].
- *Independent—supervised*: This approach is very similar to the independent weakly supervised evaluation but, here, the test is executed at the evaluator's site on the testee's hardware. The evaluator can better control the evaluation, but: 1) there is no way to compare computational efficiency (i.e., different hardware systems can be used), 2) some interesting statistics (e.g., template size, memory usage) cannot be obtained, and 3) there is no way to prevent score normalization and template consolidation [20], [16] (i.e., techniques where information from previous comparisons are unfairly exploited to increase the accuracy in successive comparisons). Examples of recent biometric evaluations of this type are [10] and [8].
- *Independent—strongly supervised*: Data are sequestered and not released before the conclusion of the test. Software components compliant to a given input/output protocol are tested at the evaluator's site on the evaluator's hardware. The tested algorithm is executed in a totally-controlled environment, where all input/output operations are strictly monitored. The main drawbacks are the large amount of time and resources necessary for the organization of such events. Examples of recent biometric evaluations of this type are [17], [18], [5], and the FVC2004 evaluation discussed in this paper.

FVC2004 follows FVC2000 [11], [17] and FVC2002 [12], [18], the first two international Fingerprint Verification Competitions organized by the authors in the years 2000 and 2002 with results presented at the 15th International Conference on Pattern Recognition (ICPR) and the 16th ICPR, respectively. The first two contests received significant attention from both academic and commercial organizations. Several research groups have used FVC2000 and FVC2002 data sets for their own experiments and some companies not participating in the original competitions later requested the organizers to measure their performance against the FVC2000 and/or FVC2002 benchmarks. Beginning with FVC2002, to increase the number of companies and, therefore, to provide a more complete overview of the state-of-the-art, anonymous participation was allowed. Table 1 compares the three competitions from a general point of view, highlighting the main differences. Table 2 summarizes the main differences between FVC2004 and the NIST Fingerprint Vendor Technology Evaluation (FpVTE2003), an important test recently carried out by the US National Institute of Standards and Technology [8].

FVC2004 was extensively publicized starting in April 2003 with the creation of the FVC2004 Web site [13]. All companies and research groups in the field known to the authors were invited to participate in the contest. All participants in the past FVC competitions were informed of the new evaluation. FVC2004 was also announced through mailing lists and biometric-related online magazines. Four new databases were collected using three commercially available scanners and the synthetic fingerprint generator SFinGe [2], [4], [1] (see Section 2). A representative subset of each database (sets B: 80 fingerprints from 10 fingers) was made available to the participants prior to the competition for algorithm tuning to accommodate the image size and the variability of the fingerprints in the databases.

Two different subcompetitions (Open category and Light category) were organized using the same databases. Each participating group was allowed to submit one algorithm in each category. The Light category was intended for algorithms characterized by low-computational resources, limited memory usage, and small template size (see Section 3.1).

By the 15 October 2003 registration deadline, we had received 110 registrations. All registered participants received the training subsets and detailed instructions for algorithm submission. By the 30 November 2003 deadline for submission, we had received a total of 69 algorithms from 45 participating groups. Since two algorithms were ultimately not accepted due to their incompatibility problems with the FVC protocol, the final number of evaluated algorithms was 67: 41 competing in the Open category and 26 in the Light category (see Table SM-I in Appendix A.1 which can be found at http://computer.org/tpami/archives.htm). Once all the executables were submitted to the evaluators, feedback was sent to the participants by providing them with the results of their algorithms over sets B (the same data set they had previously been given for algorithm tuning), thus allowing them to verify that run-time problems were not occurring on the evaluator side.

The rest of this paper is organized as follows: Section 2 describes data collection procedure and shows examples of the fingerprints included in the four databases. Section 3 introduces the testing protocol with particular emphasis on the test procedures, the performance indicators used, and the treatment of failures. In Section 4, results are presented and critically discussed by focusing not only on the matching

TABLE 1
The Three Fingerprint Verification Competitions: A Summary

	FVC2000	FVC2002	FVC2004
Call for participation	November, 1999	October, 2001	April, 2003
Registration deadline	March 1st, 2000	January 10th, 2002	October 15th, 2003
Submission deadline	June 1st, 2000	March 1st, 2002	November 30th, 2003
Evaluation period	July–August, 2000	April–July, 2002	January–February 2004
Anonymous participation	Not allowed	Allowed	
Categories	-		*Open* and *Light*
Registered participants	25 (15 withdrew)	48 (19 withdrew)	110 (64 withdrew)
Algorithms evaluated	11	31	*Open Category*: 41 *Light Category*: 26
Presentation of the results	15th ICPR Barcelona, September 2000	16th ICPR Quebec, August 2002 [18]	1st ICBA Hong Kong, July 2004 [19]
Databases	Four new databases, each one containing: set A (100x8) and set B (10x8)		
DB1	Optical (KeyTronic)	Optical (Identix)	Optical (CrossMatch)
DB2	Capacitive (ST Microelectr.)	Optical (Biometrika)	Optical (Digital Persona)
DB3	Optical (Identicator Tech.)	Capacitive (Precise Biometrics)	Thermal-sweeping (Atmel)
DB4	Synthetic (SFinGe v2.0)	Synthetic (SFinGe v2.51)	Synthetic (SFinGe v3.0)
Databases availability	DVD accompanying "Handbook of Fingerprint Recognition" [20]		Not available yet
Website	http://bias.csr.unibo.it/fvc2000	http://bias.csr.unibo.it/fvc2002	http://bias.csr.unibo.it/fvc2004
HW/SW used for running the evaluation	Pentium III (450 MHz) Windows NT FVC Test suite v1.0	Pentium III (933 MHz) Windows 2000 FVC Test suite v1.2	Athlon 1600+ (1,41 GHz) Windows XP FVC Test suite v2.0

TABLE 2
A Comparison of FVC2004 with NIST FpVTE2003

	FVC2004	FpVTE2003
Participants	43	18
Algorithms evaluated	*Open Category*: 41 *Light Category*: 26	*Large Scale Test (LST)*: 13 *Medium Scale Test (MST)*: 18 *Small Scale Test (SST)*: 3 (SST only)
Population	Students (24 years old on the average)	Operational fingerprint data from a variety of U.S. Government sources, including low-quality fingers and low-quality sources
Fingerprint format	Single finger flat impressions acquired through low-cost commercial fingerprint scanners (including small area and sweeping sensors)	Mixed formats (flat, slap and rolled) from different sources: scanned paper cards, from professional and FBI-compliant fingerprint scanners
Perturbations	Deliberately exaggerated perturbations (rotation, distortion, dry/wet fingers, ...)	Difficulties mainly due to intrinsic low-quality fingers of some subjects and sometimes due to non-cooperative users
Database availability	Databases will be made available to the scientific community, as done for the previous two editions	Databases cannot be made available due to data protection and privacy issues
Data collection	All the data were acquired for this event	Data coming from existing U.S. Government sources
Database size	4 databases, each containing 800 fingerprints from 100 fingers	48105 fingerprint sets from 25309 subjects
Evaluation type	Independent - Strongly supervised	Independent – Supervised
Anonymous participation	Allowed	Not allowed
Best EER	Best average EER: 2.07% (in the Open Category)	Best EER on MST: 0.2% (MST is the FpVTE2003 test closest to FVC2004 Open Category)

accuracy but also on efficiency, template size, and computational requirements. Section 5 suggests a simple but effective way to make scores produced by different algorithms directly comparable and applies the method to the analysis of difficult cases at the level of both fingerprint pairs and individual fingers. In Section 6, score correlation is studied and a simple fusion technique (i.e., the sum rule) is shown to be very effective. Finally, Section 7 draws some conclusions and suggests directions for future research.

2 Databases

Four databases created using three different scanners and the SFinGe synthetic generator [2], [4], [1] were used in the

TABLE 3
Scanners/Technologies Used for Collecting the Databases

	Technology	Image	Resolution
DB1	Optical Sensor (CrossMatch V300)	640×480	500 dpi
DB2	Optical Sensor (Digital Persona U.are.U 4000)	328×364	500 dpi
DB3	Thermal Sweeping Sensor (Atmel FingerChip)	300×480	512 dpi
DB4	Synthetic Generator (SFinGe v3.0)	288×384	About 500 dpi

TABLE 4
Failure-to-Acquire Rates for the Three Scanned Databases

Database	Failure to acquire
DB1	0.0%
DB2	4.8%
DB3	37.9%

FVC2004 benchmark (see Table 3). Fig. 2 shows an example image at the same scale factor from each database.

A total of 90 students (24 years old on the average), enrolled in the computer science degree program at the University of Bologna, kindly agreed to act as volunteers for providing fingerprints for DB1, DB2, and DB3:

- Volunteers were randomly partitioned into three groups of 30 persons, each group was associated to a DB and, therefore, to a different fingerprint scanner.
- Each volunteer was invited to report to the collection location in three distinct sessions, with at least two weeks time separating each session, and received brief training on using the scanner before the first session,
- Prints of the forefinger and middle finger of both the hands (four fingers total) of each volunteer were acquired by interleaving the acquisition of the different fingers to maximize differences in finger placement.
- No efforts were made to control image quality and the sensor platens were not systematically cleaned.
- At each session, four impressions were acquired of each of the four fingers of each volunteer.
- During the first session, individuals were asked to put the finger at a slightly different vertical position (in impressions 1 and 2) and to alternately apply low and high pressure against the sensor surface (impressions 3 and 4).
- During the second session, individuals were requested to exaggerate skin distortion [3] (impressions 1 and 2) and rotation (3 and 4) of the finger.
- During the third session, fingers were dried (impressions 1 and 2) and moistened (3 and 4).

In case of failure to acquire, the user was allowed to retry, until all the impressions required for each session were collected. The sweeping sensor used for collection of DB3 exhibited a failure-to-acquire rate that was significantly higher than the other two sensors (Table 4), due to the difficulties volunteers had with its particular acquisition procedure.

At the end of the data collection, we had gathered for each scanned database (DB1, DB2 and DB3) a total of 120 fingers and 12 impressions per finger (1,440 impressions) using 30 volunteers. As in our past competitions, the size of each database actually used in the test was set at 110 fingers, eight impressions per finger (880 impressions). The collection of the additional data gave us a margin in case of collection/labeling errors. To generate the synthetic DB4 to be of comparable difficulty for the algorithms, the SFinGe synthetic generator was tuned to simulate the main perturbations introduced during the acquisition of the three scanned, real databases (translation, rotation, distortion, wet/dry fingers [1]).

Figs. SM-1, SM-2, SM-3, and SM-4 in Appendix A.1 (see http://computer.org/tpami/archives.htm) show sample fingerprints from each database. The main sources of difficulty are evident: small commonality of imaged area between different images of the same finger, skin distortion, artifacts due to noise and wet fingers, poor contrast due to skin dryness or low contact pressure. FVC2004 databases were collected with the aim of creating a benchmark more difficult than FVC2002, in which the top algorithms achieved accuracies close to 100 percent. To this end, more intraclass variation was introduced, with particular emphasis on skin distortion, a well-known difficulty in fingerprint recognition.

3 Test Protocol

3.1 Test Procedure

Participants submitted each algorithm in the form of two executable programs: the first for enrolling a fingerprint image and producing the corresponding template and the second for comparing a fingerprint template to a fingerprint image and producing a comparison score in the range [0, 1]. The executables take the input from command-line arguments and append the output to a text file. The input includes a database-specific configuration file. For each database, participants were allowed to submit a distinct configuration file to adjust the algorithm's internal parameters (e.g., to accommodate the different image sizes). Configuration files are text or binary files and their I/O is the responsibility of the participant's code. These files can also contain precomputed data to save time during enrollment and comparison.

Each algorithm is tested by performing, for each database, the following comparisons:

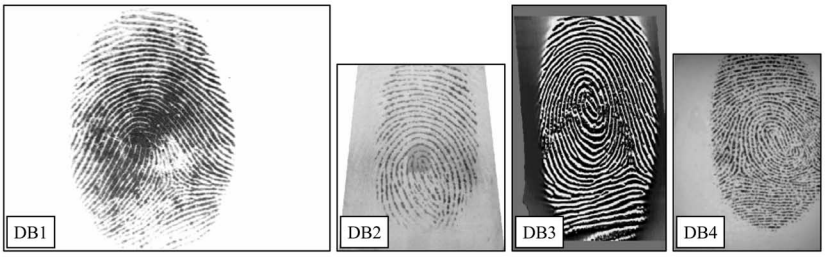


Fig. 2. A fingerprint image from each database, at the same scale factor.

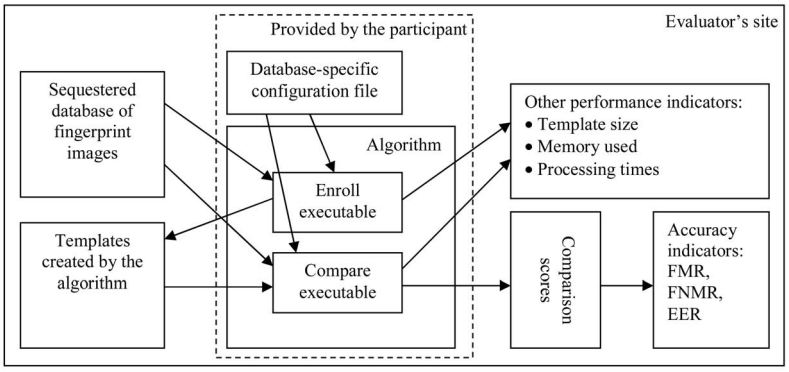


Fig. 3. Testing procedure.

- *Genuine recognition attempts*: The template of each fingerprint image is compared to the remaining images of the same finger, but avoiding symmetric matches (i.e., if the template of image j is matched against image k, template k is not matched against image j);
- *Impostor recognition attempts*: The template of the first image of each finger is compared to the first image of the remaining fingers, but avoiding symmetric matches.

Then, for each database:

- A total of 700 enrollment attempts are performed (the enrollment of the last image of any finger does not need to be performed).
- If all the enrollments are correctly performed (no enrollment failures), the total number of genuine and impostor comparison attempts is 2,800 and 4,950, respectively.

All the algorithms are tested at the evaluators' site on evaluators' hardware: The evaluation is performed in a totally-controlled environment, where all input/output operations are strictly monitored. This enables us to:

- evaluate other useful performance indicators such as processing time, amount of memory used, and template size (see Section 3.2),
- enforce a maximum response time of the algorithms,
- implement measures that guarantee algorithms cannot cheat (for instance matching filenames instead of fingerprints), and
- ensure that, at each comparison, one and only one template is matched against one and only one image and that techniques such as template consolidation [16] and score normalization [31] are not used to improve performance.

The schema in Fig. 3 summarizes the testing procedure of FVC2004.

In the Open category, for practical testing reasons, the maximum response time of the algorithms was limited to 10 seconds for enrollment and 5 seconds for comparison; no other limits were imposed.

In the Light category, in order to create a benchmark for algorithms running on light architectures, the following limits were imposed:

- maximum time for enrollment: 0.5 seconds,
- maximum time for comparison: 0.3 seconds,
- maximum template size: 2 KBytes, and
- maximum amount of memory allocated: 4 MBytes.

The evaluation (for both categories) was executed using Windows XP Professional OS on AMD Athlon 1600+ (1.41 GHz) PCs.

3.2 Performance Evaluation

For each database and for each algorithm, the following performance indicators were measured and reported:

- genuine and impostor score histograms,
- False Match Rate (FMR) and False Non-Match Rate (FNMR) graphs and Decision Error Tradeoff (DET) graph,
- Failure-to-Enroll Rate and Failure-to-Compare Rate,
- Equal Error Rate (EER), FMR100, FMR1000, ZeroFMR, and ZeroFNMR,
- average comparison time and average enrollment time,
- maximum memory allocated for enrollment and for comparison, and
- average and maximum template size.

Formal definitions of FMR (False Match Rate), FNMR (False Non-Match Rate), and Equal Error Rate (EER) are given in [17]. Note that, in single-attempt, positive recognition applications, FMR (False Match Rate) and FNMR (False Non-Match Rate) are often referred to as FAR (False Acceptance Rate) and FRR (False Rejection Rate), respectively. ZeroFMR is given as the lowest FNMR at which no False Matches occur and ZeroFNMR is the lowest FMR at which no False Non-Matches occur.

FMR100 and FMR1000 are the values of FNMR for FMR = 1/100 and 1/1000, respectively. These measures are useful to characterize the accuracy of fingerprint-based systems, which are often operated far from the EER point using thresholds which reduce FMR at the cost of higher FNMR.

FVC2004 introduces indicators measuring the amount of memory required by the algorithms and the template sizes. Table 5 summarizes the performance indicators reported in FVC2004 and compares them with those reported in the previous two competitions.

3.3 Treatment of Failures

An enrollment or comparison attempt can fail, thus resulting in a Failure-to-Enroll (FTE) or Failure-to-Compare

TABLE 5
Performance Indicators Measured in the Three FVC Competitions

Performance indicator	FVC2000	FVC2002	FVC2004
Genuine and impostor score histograms	✓	✓	✓
FMR and FNMR graph	✓	✓	✓
DET graph	✓	✓	✓
Failure To Enroll Rate	✓	✓	✓
Failure To Compare Rate	✓	✓	✓
Equal Error Rate (EER)	✓	✓	✓
FMR100		✓	✓
FMR1000		✓	✓
ZeroFMR	✓	✓	✓
ZeroFNMR	✓	✓	✓
Average match time	✓	✓	✓
Average enroll time	✓	✓	✓
Maximum memory allocated for enrollment			✓
Maximum memory allocated for comparison			✓
Average template size			✓
Maximum template size			✓

(FTC) error, respectively. Failures can be reported by the algorithm (which declares itself to be unable to process a given fingerprint) or imposed by the test procedure in the following cases:

- *timeout*: the algorithm exceeds the maximum processing time allowed,
- *crash*: the program crashes during its execution,
- *memory limit*: the amount of memory allocated by the algorithm exceeds the maximum allowed,
- *template limit* (only for enrollment): the size of the template exceeds the maximum allowed, and
- *missing template* (only for comparison): the required template has not been created due to enrollment failure, such that the comparison cannot be performed.

The last point needs an explanation: in FVC2000 [17], Failure-to-Enroll (FTE) errors were recorded apart from the FMR/FNMR errors. As a consequence, algorithms rejecting poor quality fingerprints at enrollment time could be implicitly favored since many problematic comparisons could be avoided. This could make it difficult to directly compare the accuracy of different algorithms. To avoid this problem, in FVC2004 (as in FVC2002), FTE errors are included into the computation of FMR and FNMR. In particular, each FTE error produces a "ghost template," which cannot be matched with any fingerprint (i.e., any comparison attempt involving a ghost template results in a failure to compare). Although using this technique for including Failure-to-Enroll errors in the computation of FMR and FNMR is both useful and easy for the problem at hand, this practice could appear arbitrary. In Appendix A.2, (see http://computer.org/tpami/archives.htm) it is shown that this operational procedure is equivalent to the formulation adopted in [21], which is consistent with the current best-practices [31].

4 Result Analysis

Reporting results from all the participants on the four databases would require too much space for inclusion into this paper, due to the large number of algorithms evaluated. Detailed results can be found on the competition Web site [13], together with the "medal tables" for the two categories (Open and Light) and the final rankings of the algorithms. This section, after a structured overview of the algorithms (Section 4.1), discusses: the results of the top algorithms (Section 4.2), the main differences between the two categories (Section 4.3), and efficiency, template size, and memory usage (Sections 4.4, 4.5, and 4.6). Note that, in the following graphs and tables, participant IDs (e.g., P001, P002) are used to denote the different algorithms. For instance, "P001" indicates the algorithm submitted by participant P001, since most of the participants submitted two algorithms (one for each category), the same participant ID may refer to the Open category algorithm or to the Light category algorithm, according to the context.

4.1 Overview of the Algorithms

Reporting low-level details about the approaches and techniques adopted by the participating algorithms would be unfeasible since most of the participants are commercial entities and the details of their algorithms are proprietary. For this reason, we asked all the participants to provide a high-level structured description of their algorithms by answering a few questions about:

- Preprocessing: Is segmentation (separation of the fingerprint area from the background) and/or image enhancement performed?
- Alignment: Is alignment carried out before or during comparison? What kind of transformations are dealt with (displacement, rotation, scale, nonlinear mapping)?
- Features: Which features are extracted from the fingerprint images?
- Comparison: Is the algorithm minutiae-based? If so, is minutiae comparison global or local [20]? If not, what is the approach (correlation-based, ridge-pattern-texture-based, ridge-line-geometry-based)?

A total of 29 participants kindly provided the above information. Table 6 compares the corresponding algorithms by summarizing the main information. The two histograms in Fig. 4 highlight the distribution of the features adopted and of the matching approaches, respectively.

4.2 Overall Results

In the following, results from the top algorithms in the Open category are reported. Table 7 reports the average performance indicators over the four databases for the top 10 algorithms (based on average EER).

In Fig. 5, algorithms are sorted by average EER (middle curve): For each algorithm, the best and worst EER among the four databases are plotted. In general, the best algorithms tend to have more stable performance over the different databases, with some noticeable exceptions: P039 is ranked fifth with an average EER of 2.90 percent, in spite of a quite-high EER of 7.18 percent on DB1; P103, with an average EER of 4.33 percent (the 13th in the ranking), shows a performance more stable than most of the algorithms with a lower average EER. The lowest average EER is exhibited by P101 (with a value of 3.56 percent on DB2 and a value of 0.80 percent on DB4); the lowest individual EER is achieved by P071 with 0.61 percent on DB4. Fig. 6 provides

TABLE 6
High-Level Description of the Algorithms from 29 Participants

Preprocessing			Alignment		Features								Comparison				
Participant	*Segmentation*	*Enhancement*	*Before matching, During matching*	*Displacement, Rotation, Scale, Non-linear*	*Minutiae*	*Singular points*	*Ridges*	*Ridge counts*	*Orientation field*	*Local ridge frequency*	*Texture measures*	*Raw/Enh. image parts*	*Minutiae (global)*	*Minutiae (local)*	*Ridge pattern (geometry)*	*Ridge pattern (texture)*	*Correlation*
P002	✓	✓	D	NL	✓				✓				✓	✓			
P009	✓	✓	BD	DRS	✓	✓	✓	✓	✓	✓			✓				
P016		✓	-	-	✓		✓						✓				✓
P026			-	DR	✓			✓	✓				✓				
P027	✓	✓	D	DRS							✓					✓	✓
P039	✓	✓	D	N	✓				✓	✓			✓				
P041	✓	✓	D	DR	✓		✓								✓		
P047	✓		D	DRSN	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	
P049	✓	✓	D	DR	✓	✓	✓		✓	✓			✓	✓			
P050	✓	✓	B	DR	✓				✓				✓				
P051	✓	✓	D	DR	✓	✓		✓	✓				✓	✓			✓
P067	✓	✓	D	DRN	✓				✓				✓				
P068	✓	✓	D	DR	✓		✓						✓				✓
P071	(✓)	✓	D	DR(N)	✓	✓	✓	✓	✓			(✓)		✓	✓		(✓)
P072	✓	✓	D	DR	✓	✓			✓				✓				
P075	✓	✓	B	DR	✓									✓			
P078	✓	✓	D	DRS	✓								✓				
P087	✓	✓	D	DR	✓				✓	✓	✓		✓	✓	✓		
P097	✓		D	DR	✓	✓			✓	✓			✓	✓			
P099	✓	✓	D	DRN	✓								✓				
P101	(✓)	✓	BD	DRS		✓	✓	✓	✓	✓		✓			✓		✓
P103	✓	✓	-	-	✓				✓					✓			
P104	✓	✓	D	DR	✓									✓			
P105	✓	✓	B	DR	✓	✓			✓					✓	✓		
P106			-	-	✓		✓							✓			
P107		✓	D	DRS	✓	✓							✓	✓			
P108	✓	✓	D	DR	✓	✓			✓					✓			
P111	✓	✓	D	DR	✓				✓				✓				
P113	✓	✓	D	N	✓	✓	✓		✓	✓		✓	✓	✓			✓

Notes about P071: Segmentation is performed only in the Light category; alignment type is Displacement + Rotation in the Light category and Nonlinear in the Open; Raw image parts and Correlation are used only in the Open category. Note about P101: Segmentation is performed only on DB1 images.

complementary information to Fig. 5, by plotting, for each algorithm, the EER on the four databases. DB1 has proven to be the most difficult for most of the algorithms, mainly due to the presence of a considerable number of distorted fingerprints (skin distortion was encouraged during the second acquisition session on all scanners, but the scanner used for DB1 allowed more easily for the collection of prints exaggerating this kind of perturbation).

The easiest database for most of the algorithms was DB4 (the synthetic one), but the behavior of the algorithms over DB4 was, in general, comparable to that on the real databases, thus confirming that the fingerprint generator is able to emulate most of the perturbations encountered in the real fingerprint databases.

The graphs in Figs. 5 and 6 are based on the EER, which is an important statistic, indicating an equal trade-off between false match and false nonmatch error rates. But the EER threshold is just one of the possible decision points at which the algorithms can operate. Comparing the algorithms at the ZeroFMR operating point (Fig. 7) and ranking them according to the average ZeroFMR value confirms the excellence of the top two algorithms (P101 and P047 also in this case). Other algorithms show some changes in ranking, the most noticeable being P039. This algorithm shows a reasonable ZeroFMR on three databases (DB1 = 18.00 percent, DB2 = 8.18 percent, and DB4 = 2.71 percent) and is the fifth best algorithm based on average EER (Fig. 5), but it exhibits an extremely high ZeroFMR on DB3 (99.61 percent). This poor performance is caused by three impostor comparisons which resulted in a very high score (i.e., the fingerprints are considered very similar by the algorithm). Fig. SM-5 in Appendix A.1 (see http://computer.org/tpami/archives.htm) shows one such pair; this may have been caused due to the large, noisy area in the middle of both the prints. A more comprehensive view of the results for the top five algorithms is given in Fig. SM-6 of Appendix A.1, (see http://computer.org/tpami/archives.htm) reporting, for each database, the DET curves (which show error rate trade-offs at all possible operating points). The lines corresponding to EER, ZeroFMR, ZeroFNMR, FMR100, and FMR1000 are highlighted in the graphs; the corresponding numerical values are reported in Tables SM-II, SM-III, SM-IV, and SM-V in Appendix A.1 (see http://computer.org/tpami/archives.htm), together with the details of the other nonaccuracy-related indicators (see Section 3.2).

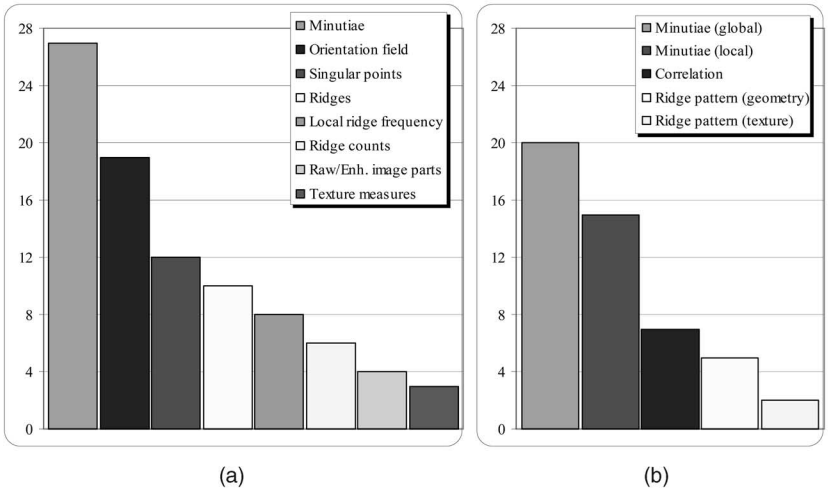


Fig. 4. Histograms of the (a) distribution of the different features exploited by the algorithms and of (b) the comparison approaches. Note that the same algorithm usually exploits several features and often adopts several comparison approaches.

TABLE 7
Open Category—Average Results over the Four Databases: Top 10 Algorithms, Sorted by EER

Algorithm	EER (%)	FMR100 (%)	FMR1000 (%)	ZeroFMR (%)	FTE (%)	FTC (%)	Avg Enroll Time (s)	Avg Comparison Time (s)	Avg Model Size (KB)	Max Model ize (KB)	Max Enroll Memory (KB)	Max Comparison Memory (KB)
P101	2.07	2.54	4.70	6.21	0.00	0.00	0.08	1.48	24	31.5	3204	7752
P047	2.10	2.96	4.61	6.59	0.00	0.00	2.07	2.07	1.3	2.8	5080	5796
P071	2.30	2.73	5.10	10.01	0.00	0.01	0.35	0.67	16.4	31.4	5872	9800
P004	2.45	3.27	5.63	7.34	0.00	0.00	0.69	0.71	2	3.8	7012	7032
P039	2.90	4.57	7.44	32.13	0.00	0.00	1.01	1.19	3.1	4.2	4192	4276
P097	3.13	4.49	7.30	11.85	0.04	0.02	0.47	0.51	14.9	28.1	5564	5780
P049	3.24	5.56	9.25	12.62	0.00	0.00	0.34	0.38	0.5	1.2	2472	2496
P009	3.31	4.93	8.32	11.63	0.00	0.00	0.25	0.24	1.3	2.9	2828	2860
P113	3.71	6.07	8.76	11.29	0.00	0.00	0.45	0.48	49.2	93.2	11936	12260
P068	4.03	6.87	11.08	15.68	0.00	0.00	0.43	0.47	7.9	7.9	4468	4456

4.3 Open Category versus Light Category

Table 8 reports the top 10 participants in the Light category based on average EER (see Table 7 for the corresponding data in the Open category). Fig. 8 compares the performance of the algorithms submitted by participants P101 and P071 to the two categories.

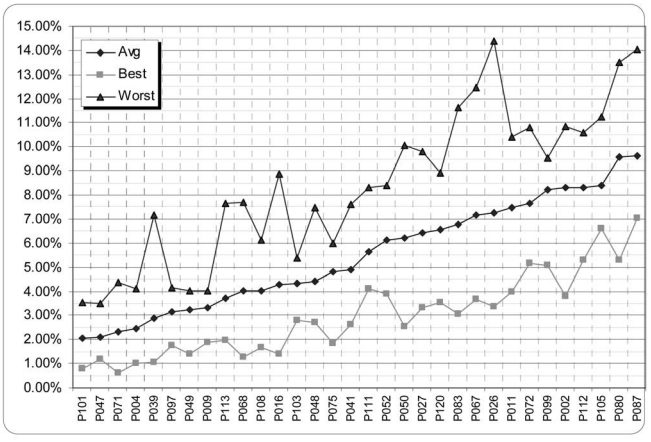


Fig. 5. Open category: EER (average, best, and worst over the four databases). Only algorithms with average EER less than 10 percent are reported.

The performance drop between the Open and Light categories is significant: P101 average EER is about 2.07 percent in the Open (overall best result) and 4.29 percent in the Light category. The overall best average EER in the Light category is 3.51 percent (P009, see Table 8), an error rate which is significantly higher than the best result in the Open category. Almost all the participants submitting two algorithms showed poorer performance in the light category, with the minor exception of P108 (average EER of 4.04 percent in the Open and 3.96 percent in the Light). This means that: 1) most of the participants had to modify their algorithms (or at least adjust some parameters) to meet the Light category constraints and 2) such modifications heavily impacted performance. Table 9 shows that the average performance drop for the top 10 participants (selected according to the average EER in the Open category) is more than 40 percent on EER and more than 35 percent on ZeroFMR.

Such a general performance drop is higher than we expected, considering that the constraints in the Light category (Section 3.1) were not considered "strict" (maximum time for enrollment: 0.5 seconds; maximum time for comparison: 0.3 seconds; maximum template size: 2 KBytes; and maximum amount of memory allocated: 4 MBytes). These are typical of the current constraints on an embedded/standalone

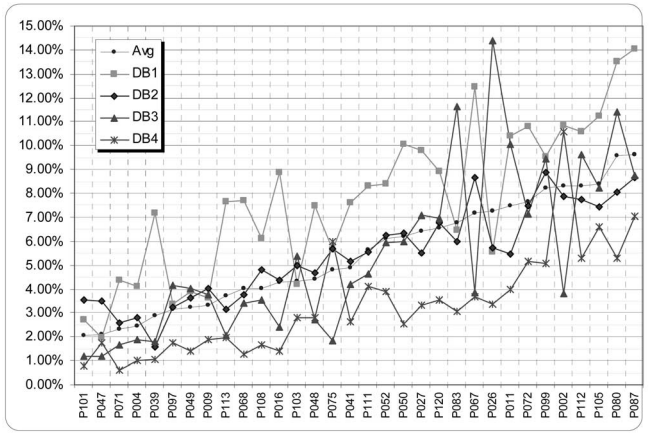


Fig. 6. Open category: Individual EER on each database and average EER. Only algorithms with average EER less than 10 percent are reported.

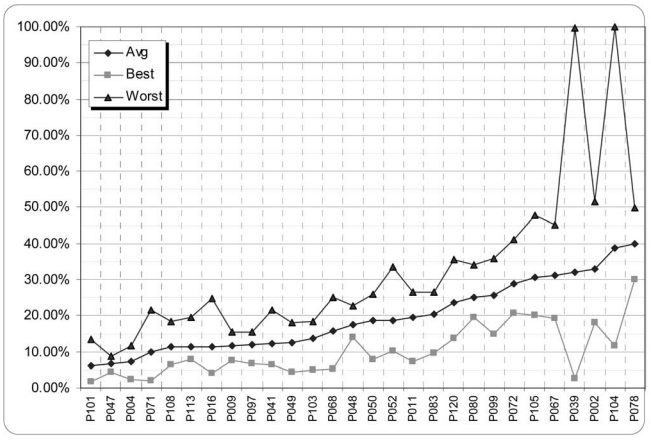
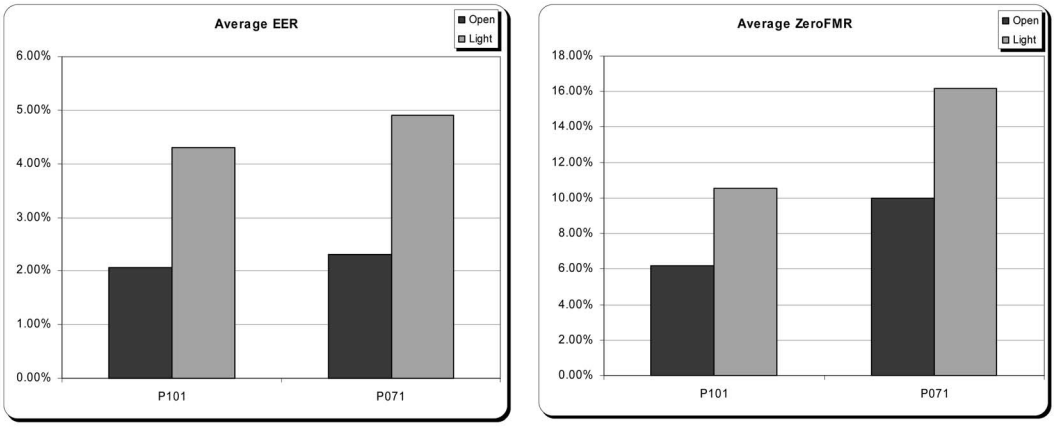


Fig. 7. Open category: ZeroFMR (average, best, and worst over the four databases). Only algorithms with average ZeroFMR less than 40 percent are reported.

TABLE 8
Light Category—Average Results over the Four Databases: Top 10 Algorithms, Sorted by EER

Algorithm	EER (%)	FMR100 (%)	FMR1000 (%)	ZeroFMR (%)	FTE (%)	FTC (%)	Avg Enroll Time (s)	Avg Comparison Time (s)	Avg Model Size (KB)	Max Model Size (KB)	Max Enroll Memory (KB)	Max Match Memory (KB)
P009	3.51%	5.21%	8.71%	12.38%	0.00%	0.00%	0.25	0.22	1.2	2	2844	2568
P107	3.69%	4.68%	6.65%	8.75%	0.00%	0.00%	0.13	0.13	0.2	0.6	1788	1800
P108	3.96%	6.54%	10.64%	13.12%	0.04%	0.05%	0.23	0.23	1.6	1.6	1952	1976
P101	4.29%	6.02%	8.91%	10.57%	0.00%	0.00%	0.09	0.17	1.1	1.2	2228	3044
P103	4.33%	6.66%	9.97%	13.64%	0.00%	0.00%	0.13	0.14	1.2	2	3572	3668
P097	4.86%	6.96%	10.21%	13.12%	0.00%	0.00%	0.19	0.19	2	2	2100	2108
P071	4.91%	8.11%	11.44%	16.16%	0.25%	0.18%	0.19	0.18	1.2	1.7	2552	2424
P016	5.26%	7.68%	10.46%	13.93%	0.00%	0.00%	0.17	0.19	1.4	2	2240	3004
P068	5.29%	9.85%	15.22%	20.18%	0.00%	0.00%	0.16	0.18	2	2	3448	3428
P049	5.64%	10.55%	17.13%	24.12%	0.00%	0.00%	0.12	0.14	0.5	0.9	1956	1980



(a) (b)

Fig. 8. The two top participants in the Open category that also submitted an algorithm to the Light category: (a) Comparison of the average EER over the four databases and (b) of the average ZeroFMR.

TABLE 9
Top 10 Algorithms in the Open Category and Corresponding Algorithms in the Light Category: Average EER, Average ZeroFMR, and the Corresponding Percentage Variations Are Reported: The Last Row Shows the Overall Averages

	Open category (averages)		Light category (averages)		Performance drop (averages)	
	EER	*ZeroFMR*	*EER*	*ZeroFMR*	*EER*	*ZeroFMR*
P101	2.07%	6.21%	4.29%	10.57%	107.25%	70.21%
P071	2.30%	10.01%	4.91%	16.16%	113.48%	61.44%
P097	3.13%	11.85%	4.86%	13.12%	55.27%	10.72%
P049	3.24%	12.62%	5.64%	24.12%	74.07%	91.13%
P009	3.31%	11.63%	3.51%	12.38%	6.04%	6.45%
P068	4.03%	15.68%	5.29%	20.18%	31.27%	28.70%
P108	4.04%	11.25%	3.96%	13.12%	-1.98%	16.62%
P016	4.27%	11.32%	5.26%	13.93%	23.19%	23.06%
P103	4.33%	13.64%	4.33%	13.64%	0.00%	0.00%
P041	4.89%	12.31%	6.21%	18.09%	26.99%	46.95%
				Average	**44%**	**36%**

solution running on a 50-200 MIPS CPU. Match-on-token and match-on-card solutions [26], [30], currently receiving attention due to their privacy and security enhancing characteristics, have much more stringent requirements (i.e., the 0.3 second comparison time limit in our Light category refers to a CPU performing at more than 3000 MIPS, while a typical smart card CPU performs at about 10 MIPS). What would be the performance degradation of these algorithms if adapted to run on a smart card? New evaluation programs with specific protocols for match-on-card algorithms are needed to answer this question.

4.4 Matching Speed

Table 10 reports average comparison times in the Open category. Note that a "comparison" operation consists of the comparison of a given template with a fingerprint image (see Section 3). Hence, the "comparison time" includes the feature extraction time for one of the fingerprints. The overall average

TABLE 10
Open Category: Average Comparison Time on Each Database for Algorithms with Average EER Less than 8 Percent

	Average comparison time (seconds)					Average EER
	DB1	*DB2*	*DB3*	*DB4*	*Avg.*	
P101	3.19	0.81	0.85	1.06	1.48	2.07%
P047	1.87	2.18	2.30	1.93	2.07	2.10%
P071	0.77	0.57	0.81	0.53	0.67	2.30%
P004	0.75	0.62	0.80	0.65	0.71	2.45%
P039	1.32	0.83	1.09	1.53	1.19	2.90%
P097	0.75	0.53	0.22	0.52	0.51	3.13%
P049	0.47	0.34	0.40	0.32	0.38	3.24%
P009	0.23	0.23	0.30	0.22	0.25	3.31%
P113	0.69	0.35	0.50	0.37	0.48	3.71%
P068	0.65	0.39	0.47	0.37	0.47	4.03%
P108	0.35	0.31	0.35	0.26	0.32	4.04%
P016	0.35	0.31	0.39	0.30	0.34	4.27%
P103	0.16	0.12	0.15	0.12	0.14	4.33%
P048	0.40	0.36	0.45	0.35	0.39	4.41%
P075	0.44	0.30	0.68	0.29	0.43	4.79%
P041	0.20	0.14	0.19	0.14	0.17	4.89%
P111	0.64	0.34	0.42	0.32	0.43	5.66%
P052	0.26	0.22	0.28	0.20	0.24	6.12%
P050	0.66	0.51	0.78	0.48	0.61	6.23%
P027	3.26	2.46	2.61	2.17	2.63	6.42%
P120	2.00	1.08	1.26	0.87	1.30	6.56%
P083	0.37	0.43	0.52	0.37	0.42	6.80%
P067	0.08	0.04	0.05	0.04	0.05	7.17%
P026	3.56	3.26	5.10	3.20	3.78	7.27%
P011	0.69	0.37	0.45	0.35	0.47	7.48%
P072	0.12	0.10	0.12	0.09	0.11	7.64%
Avg.	**0.93**	**0.66**	**0.83**	**0.66**	**0.77**	**4.74%**

The last row reports the averages of the above times.

times (bottom row in Table 10) reflect the different image sizes (DB1: 307 KPixels, DB2: 119 KPixels, DB3: 144 KPixels, DB4: 108 KPixels, see Section 2). As can logically be expected, algorithms generally take more time to process and compare larger images. On the other hand, detailed analysis of the time data reveals interesting exceptions with many algorithms (for instance, P047, which took more time for comparison in DB3 and DB2 than in DB1, or P097, which exhibited a much lower time on DB3 than on the other databases). This may indicate, at least in some cases, that the database-specific adjustment of the algorithms (allowed by the FVC protocol, see Section 3) involved operations with a considerable impact on the efficiency (e.g., different enhancement approaches or parameters, different degrees of freedom in the matching, etc.).

The most accurate algorithm (P101) shows a quite high average comparison time: 1.48 seconds is its overall average, about twice the average of all the algorithms in Table 10 (0.77 seconds). The high value is mostly due to the very high comparison time in DB1 (3.19 seconds). It is worth noting that the most accurate algorithm on DB1 was P047 (see Table SM-II in Appendix A.1 http://computer.org/tpami/archives.htm), which exhibits comparison times more consistent across the different databases, but definitely higher overall: an average of 2.07 seconds, which is the highest among the most accurate algorithms. A look at Table 6 shows that the low speed of P047 is probably due to the large number of features it extracts and to the alignment and matching techniques which appear computationally intensive.

Algorithms exhibiting good tradeoffs between speed and accuracy were P071 and P009: The former achieving the third-best average EER, with an average comparison time of 0.67 seconds; the latter coupling a reasonable accuracy to a quite low comparison time. Another interesting result was obtained by P103, with an average time of 0.14 seconds. The highest speed was achieved by P067, but at the cost of a definitely lower accuracy. As shown in Table 6, it appears that fast algorithms like P009, P067, and P103 owe their speed mainly to an efficient implementation of minutiae-based comparison techniques. Combining different comparison approaches, which exploit different features, can definitely improve accuracy (see P047 or P101), but obviously at the cost of lower efficiency.

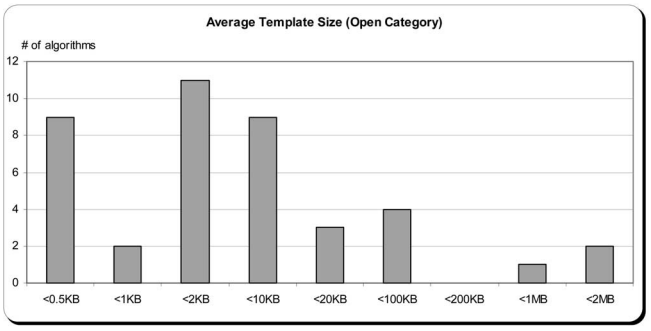


Fig. 9. Open category: Histogram of average template sizes over the four databases.

4.5 Template Size Analysis

The histogram in Fig. 9 reports the distribution of average template sizes among the four databases. Tables SM-II, SM-III, SM-IV, and SM-V in Appendix A.1 (see http://computer.org/tpami/archives.htm) report the per-database averages for the top algorithms.

Template sizes less than 0.5KB are usually indicative of algorithms based only upon minutiae. This is supported by Table 6, where six of the nine algorithms in the left-most column of the histogram are present. All of them adopt a matching technique based only on minutiae (local, global, or both), with the exception of P087, which compares ridge geometry in addition to minutiae.

Template sizes in the range of 1KB to 2KB are likely to contain not only minutiae points but also other features to help the alignment of the two fingerprints and the subsequent comparison process. The most commonly used additional feature for this purpose is the orientation field [20]. All of the algorithms for which there is information in Table 6 (eight out of 12) extract both minutiae points and the orientation field as input features.

Very large template sizes are probably due to the storage of some portions of the fingerprint image itself (e.g., raw or enhanced image blocks to be used for correlation). The seven algorithms which are indicated in Table 6 as exploiting a correlation-based comparison approach have template sizes ranging from 5KB to about 50KB.

The two right-most columns of the histogram in Fig. 9 refer to three algorithms (P079, P109, and P118) with extremely large templates (larger than the 170KB average image size over the four databases). No information was provided by the designers of these algorithms, so it is not possible to understand the reasons for such a huge utilization of storage space. We can speculate that, when the template is larger than the image, it probably contains

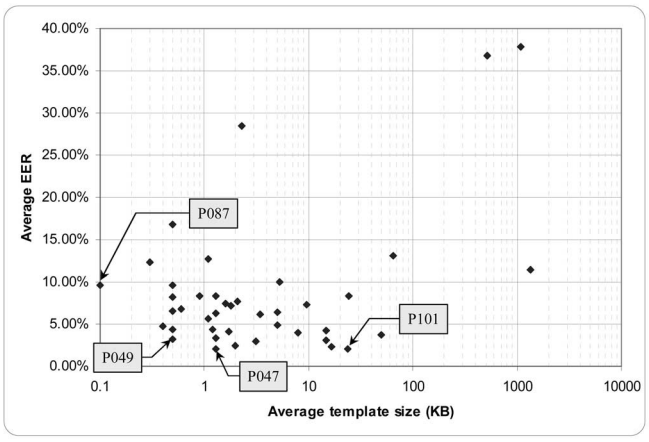


Fig. 10. Open category: Correlation between average template size (x axis, using a logarithmic scale) and average EER over the four databases (y axis); each point corresponds to an algorithm.

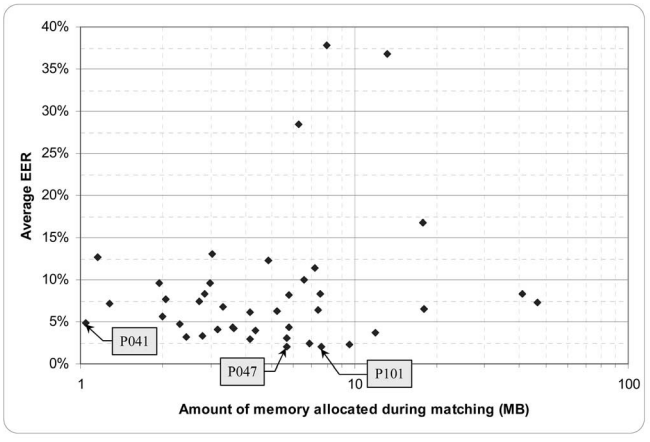


Fig. 11. Open category: Correlation between the maximum amount of memory allocated during comparison (x axis, using a logarithmic scale) and average EER over the four databases (y axis); each point corresponds to an algorithm.

some redundant precomputed data useful in speeding up the comparison process (e.g., rotations of the image).

Fig. 10 plots the average template size versus the average EER over the four databases for all the algorithms. Although correlation exists, the scattered cloud of points testifies that storing more information does not necessarily translate to achieving better performance. The overall best algorithm based on average EER (P101) achieves an average EER of 2.07 percent, with an average template size of 24 KBytes. A comparable result (2.10 percent) is obtained by P047, with a much smaller average template size of 1.3KB. An interesting result is also the 3.24 percent average EER with 0.5KB average template size for algorithm P049. The smallest average template size is exhibited by P087 (0.1KB), with an average EER of 9.62 percent.

4.6 Amount of Memory Used

Table 7 reports the maximum amount of memory allocated by the top performing algorithms over the four databases during comparison and enrollment, respectively. Tables SM-II, SM-III, SM-IV, and SM-V in Appendix A.1 (see http://computer.org/tpami/archives.htm) report the statistics for the top algorithms on each database. The amount of memory considered is the total quantity reported by the Operating System, which includes space allocated for both the code and data.

Fig. 11 correlates the maximum amount of memory to the accuracy (average EER over the four databases). Almost all the algorithms with an average EER below 5 percent use more than 2MB of memory; the only exception being P041, which achieves an EER of 4.89 percent using about 1MB of memory. The two most accurate algorithms (P101 and P047) show fairly high memory usage (7.6MB and 5.7MB, respectively). Judging by the data available in Table 6, almost all the algorithms that use less than 3MB of memory perform comparisons using only minutiae points. Matching techniques based on multiple modalities require greater amounts of memory, especially when image correlation is involved.

5 Comparing Algorithms at Score Level

5.1 Definitions

In general, comparison scores from different algorithms are not directly comparable even if they are restricted to a prescribed range (e.g., FVC protocol requires scores to be in the range [0, 1], see Section 3.1). Moreover, it is not even possible to directly compare scores of the same algorithm on different databases.

A simple but effective a posteriori technique for the comparison of the outputs of different algorithms is proposed here:

- let g_a and g_b be the scores produced by algorithms a and b, respectively, for the same genuine comparison on a given database,
- let FMR(g_a) be the False Match Rate of algorithm a (on that database) when the threshold is set to g_a (that is, the minimum percentage of false match errors that the algorithm would make if it were forced to accept as genuine a comparison with score g_a), and
- let FMR (g_b) be the corresponding value for algorithm b;

then, FMR(g_a) and FMR(g_b) are two directly-comparable measures of how much the given genuine comparison is difficult for algorithms a and b, respectively: The closer to zero, the easier the genuine comparison.

Analogously, the corresponding values of FNMR can be used to compare the difficulty of impostor comparisons. In the following, for a generic algorithm p, we will denote the above-defined *difficulty values* as:

$DV_G(p, x, y)$, for a genuine comparison between fingerprints x and y, and

$DV_I(p, w, z)$, for an impostor comparison between fingerprints w and z.

Some analyses performed exploiting the above approach are described in the rest of this section.

5.2 Average "Difficulty" of Genuine and Impostor Fingerprint Pairs

The average "difficulty" of each fingerprint pair can be simply measured by averaging the difficulty values among all the algorithms:

$$\overline{DV_G}(x, y) = \frac{\sum_{p \in P} DV_G(p, x, y)}{\#P},$$

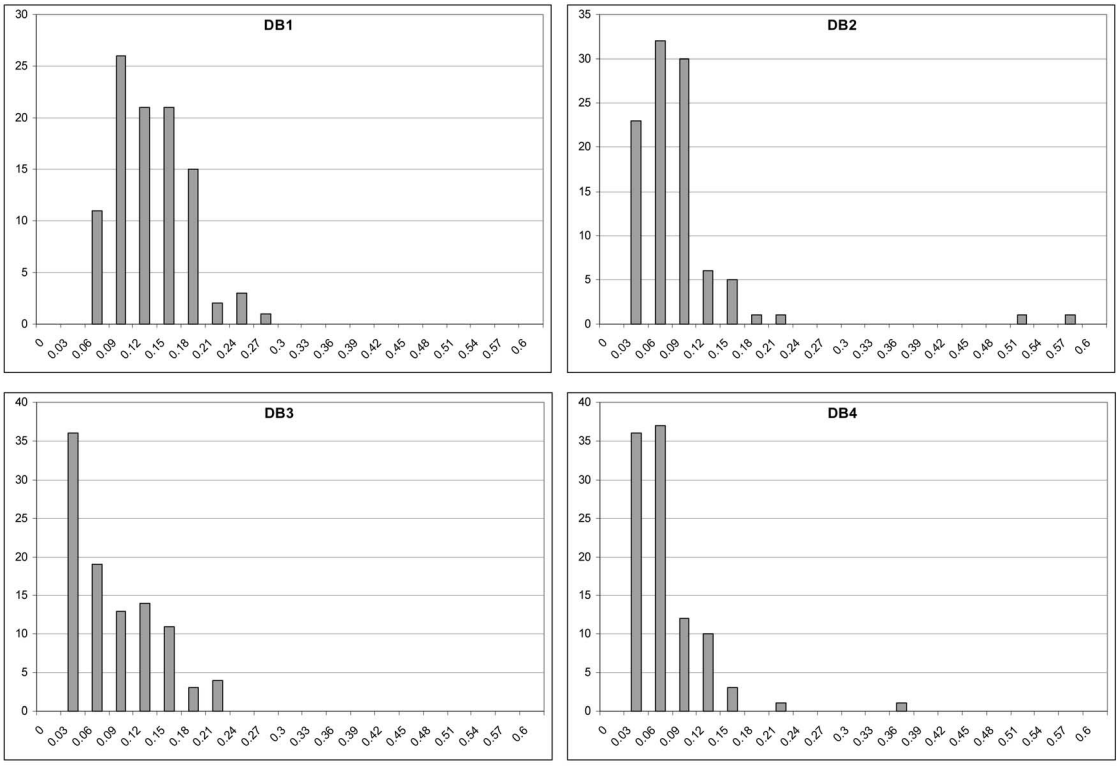


Fig. 12. Open category: Histograms of the $\overline{D_G}(f)$ values on the four databases.

$$\overline{DV_I}(x,y) = \frac{\sum_{p \in P} DV_I(p,x,y)}{\#P},$$

where P is the set containing all the algorithms. Figs. SM-7, SM-8, and SM-9 in Appendix A.1 report (see http://computer.org/tpami/archives.htm), for the three real databases, the number of genuine fingerprint pairs that the algorithms found, on the average, to be the easiest and the most difficult. Analogously, Figs. SM-10, SM-11, and SM-12 in Appendix A.1 report (see http://computer.org/tpami/archives.htm)the impostor pairs. As predictable, the most difficult impostor pairs always consist of two fingerprints belonging to the same class/type (right loop in DB1, whorl in DB2 and left loop in DB3).

5.3 Intrinsic Difficulty of Individual Fingers

From the average difficulty of fingerprint pairs, it is possible to derive a measure of individual "difficulty" of a given finger:

$$\overline{D_G}(f) = \frac{\sum_{(a,b) \in F_G} \overline{DV_G}(a,b)}{\#F_G},$$

where f is a given finger and F_G is the set of all the genuine comparisons involving impressions of f.

$$\overline{D_I}(f) = \frac{\sum_{(a,b) \in F_I} \overline{DV_I}(a,b)}{\#F_I},$$

where f is a generic finger and F_I is the set of all the impostor comparisons involving impressions of f.

A high value of $\overline{D_G}(f)$ indicates that impressions of finger f are likely to be falsely nonmatched against impressions of the same finger (i.e., f is a finger more difficult to be recognized than others); a high value of $\overline{D_I}(f)$ indicates that impressions of finger f are likely to be falsely matched against impressions of other fingers (i.e., f is a finger easier to be mistaken for another).

The previous analysis, performed at the level of comparisons, showed that some genuine fingerprint pairs are more/less difficult than others. This result was expected because of the different perturbations introduced during database collection (Section 2). On the other hand, since all the volunteers were requested to introduce the same perturbations, one could expect a certain amount of uniformity when the difficulty is analyzed at finger level. Actually, as Fig. 12 shows and as other studies have highlighted [6], [27], it is evident that some fingers (whose owners are affectionately referred to as "goats" [6]) are more difficult to recognize than others. This may be particularly marked, as in the case of two fingers in DB2 (Fig. 13 shows the eight impressions of the worst one).

Fig. 14 shows the histograms of finger difficulty with respect to the impostor comparisons. In this case, finger difficulties fall into a narrower interval and the distributions do not exhibit outliers. Therefore, we can conclude that FVC2004 databases do not include fingers (referred to as "wolves/lambs" in [6]) that are more likely to be falsely matched.

6 Fusing Algorithms at Score Level

The matching difficulty values introduced in Section 5 can be used to measure the correlation among different algorithms on each database. A strong correlation of the difficulty values of two algorithms means that they made similar errors (i.e., they consistently found the same

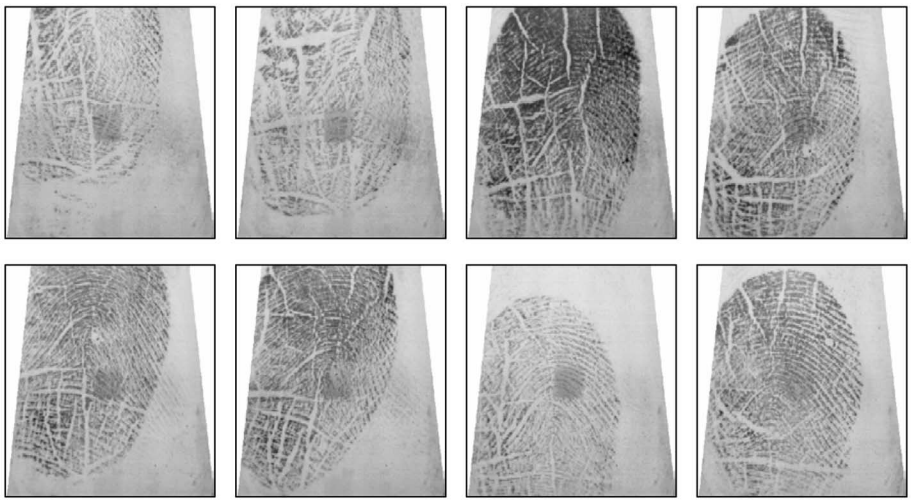


Fig. 13. Database 2: The eight impressions of the finger corresponding to the rightmost bar in the DB2 histogram of Fig. 12.

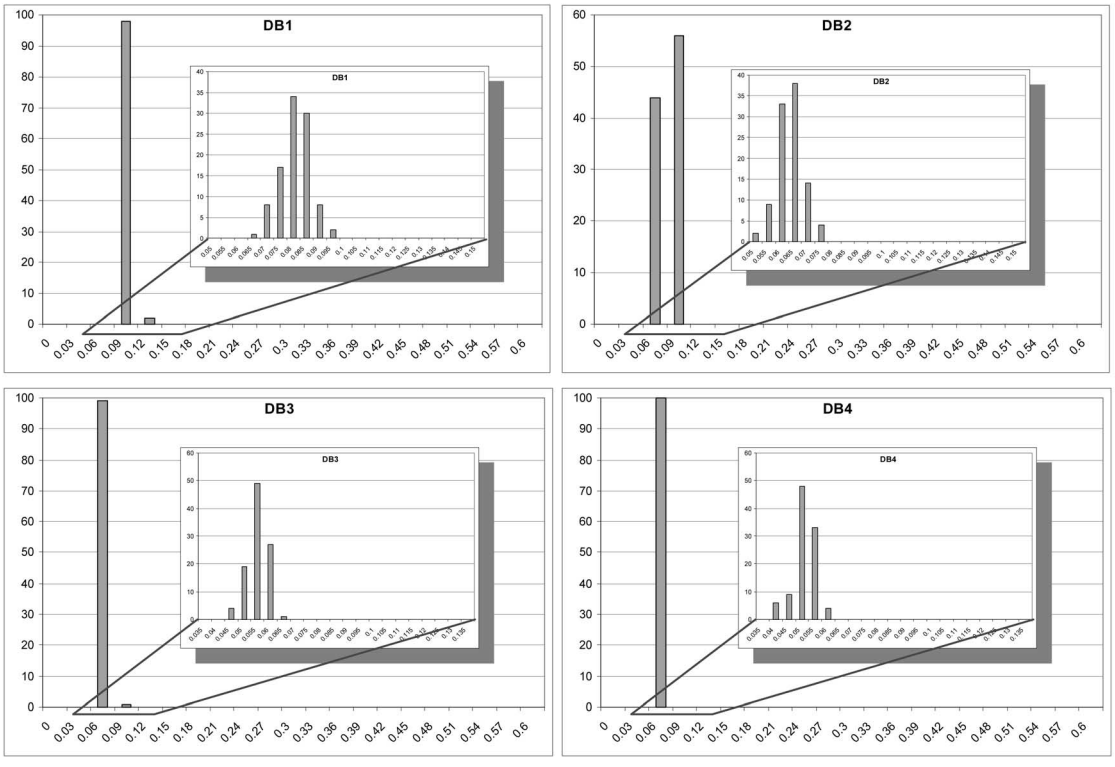


Fig. 14. Open category: Histograms of the $\overline{D_I}(f)$ values on the four databases. The scale of the horizontal axis is the same as in Fig. 12 to allow a direct comparison. The intervals of interest are expanded in the inner graphs.

fingerprint pairs to be particularly difficult); a low correlation indicates that they made different errors.

Tables 11, 12, and 13 report results from the five top algorithms in the Open category for which the high-level description has been provided in Table 6. Table 11 shows the average correlation on genuine comparisons over the four databases and Table 12 shows the average correlation on impostor comparisons. A first observation is that the correlation on impostor pairs is definitely lower than that on genuine pairs. This result is not unexpected because difficulty in genuine-pair comparisons is often caused by well-identifiable perturbations (little commonality of imaged finger area, distortion, severe noise, etc.), whereas difficulty for impostor-pair comparisons is more algorithm-dependent since it is more related to the specific features used and the way they are processed.

The average correlation on genuine comparisons in Table 11 is very low, which is quite surprising, considering that the table reports the results of top algorithms. The database where those algorithms are less correlated is DB3 (Table 13). This may be due to the particular nature of the images, obtained by a sweeping sensor (see Fig. 2) for which most of the algorithms are probably not optimized. Such a low correlation suggests that combining some of the algorithms could lead to improved accuracy [15]. Although studying optimal combination strategies (e.g., using trained combiners [7]) is beyond the aims of this paper, three very simple fusion

TABLE 11
Open Category, Genuine Comparisons: Average Correlation of the Corresponding Difficulty Values for the Top Algorithms over the Four Databases

	P039	*P047*	*P071*	*P097*	*P101*
P039	1.00				
P047	0.16	1.00			
P071	0.25	0.28	1.00		
P097	0.22	0.37	0.30	1.00	
P101	0.26	0.25	0.26	0.23	1.00

TABLE 12
Open Category, Impostor Comparisons: Average Correlation of the Corresponding Difficulty Values for the Top Algorithms over the Four Databases

	P039	*P047*	*P071*	*P097*	*P101*
P039	1.00				
P047	0.01	1.00			
P071	0.14	0.04	1.00		
P097	0.07	-0.01	0.08	1.00	
P101	0.04	-0.01	0.06	0.14	1.00

experiments have been performed by using the sum rule (i.e., the matching score of the combined system is defined as the sum of the scores produced by the individual algorithms): 1) combination of P039 and P071 (the two least correlated), 2) combination of P047 and P101 (the two most accurate), and 3) combination of all the five algorithms. The results are reported in Table 14, together with the individual performance of the algorithms. As expected, performance greatly benefits from the combination: For example, by combining the top five algorithms, the EER on DB3 decreased from 1.18 percent (top single algorithm) to 0.28 percent.

7 Conclusions

Performance evaluation is important for all pattern recognition applications and particularly so for biometrics, which is receiving widespread international attention for citizen identity verification and identification in large-scale applications. Unambiguously and reliably assessing the current state of the technology is mandatory for understanding its limitations and addressing future research requirements. This paper reviews and classifies current biometric testing initiatives and assesses the state-of-the-art in fingerprint verification through presentation of the results of the third international Fingerprint Verification Competition (FVC2004). Results are critically reviewed and analyzed with the intent of better understanding the performance of these algorithms. We can conclude that:

- The interest shown in the FVC testing program by algorithm developers is steadily increasing. In this third edition (FVC2004), a total of 67 algorithms have been evaluated by the organizers. FVC2000 and FVC2002 fingerprint databases, now available to the scientific community, constitute the most frequently used benchmarking databases in scientific publications on fingerprint recognition.
- Performance of top fingerprint algorithms is quite good (best EER over the four databases is 2.07 percent), particularly if we consider that the databases have been intentionally made difficult (more difficult than FVC2002) by exaggerating perturbations such as skin distortion and suboptimal skin conditions (e.g., wet and dry) known to degrade algorithm performance.
- Most of the algorithms tested are based on global minutiae matching, which is still one of the most reliable approaches for fingerprint recognition. However, the use of a larger variety of features (in addition to minutiae) and alternative/hybrid matching techniques is now common, especially for the best performing algorithms. This needs to be carefully considered when defining standards for template storage [14].
- A fingerprint verification algorithm cannot be characterized by accuracy indicators only. Computational efficiency and template size could make an algorithm appropriate or unsuitable in a given application. Measuring and comparing such characteristics among different algorithms is possible only for a strongly supervised independent evaluation such as FVC.
- If restrictions are made on maximum response time, template size, and memory usage, the resulting loss in accuracy can be significant. The algorithm with best EER (2.07 percent) on the Open category exhibits a 4.29 percent EER in the Light category.
- Our results confirm that matching difficulty is not equally distributed among fingerprint pairs, some fingers being more difficult to match than others.
- Surprisingly, error correlation between best performing algorithms is very low. That is, different algorithms tend to make different errors. This indicates that there is still much potential for algorithmic improvement. Our experiments show that simply combining algorithms at the score level allows accuracy to be markedly improved. By combining the top five algorithms, the EER on DB3 dropped from 1.18 percent (for the top single algorithm) to 0.28 percent (for the combination).

TABLE 13
Open Category, Genuine Comparisons on DB3: Average Correlation of the Corresponding Difficulty Values for the Top Algorithms

	P039	*P047*	*P071*	*P097*	*P101*
P039	1.00				
P047	0.15	1.00			
P071	0.07	0.08	1.00		
P097	0.23	0.24	0.14	1.00	
P101	0.36	0.14	0.24	0.14	1.00

TABLE 14
Open Category—DB3: Results of Three Combination Experiments Using the Sum Rule

Algorithm	EER (%)	FMR100 (%)	FMR1000 (%)	ZeroFMR (%)	ZeroFRR (%)
P047	1.18	1.68	2.68	4.89	100.00
P101	1.20	1.32	3.11	3.79	81.66
P071	1.64	1.86	4.96	9.89	100.00
P039	1.78	2.32	5.64	99.61	37.98
P097	4.16	6.36	8.79	15.43	100.0
P039+P071	**0.81**	**0.79**	**3.14**	**8.57**	**19.84**
P047+P101	**0.49**	**0.36**	**0.93**	**2.39**	**75.68**
All five	**0.28**	**0.14**	**0.36**	**0.86**	**20.06**

We are currently planning a new testing initiative (FVC2006) with the intention of leveraging our experience gained in previous editions. We are considering the inclusion of two new categories:

- With the aim of decoupling feature extraction and feature comparison performance, participants will be asked to produce templates in a standard format (e.g., [14]). This would allow us to evaluate interoperability and interchange of templates across algorithms.
- With the aim of better understanding the degradation in accuracy for "very-light" architectures, a match-on-card category will be introduced, enabling computational constraints typical of a smart card.

APPENDIX

Appendices A.1 and A.2 are included in the supplemental material which can be found at http://computer.org/tpami/archives.htm.

ACKNOWLEDGMENTS

This work was partially supported by the European Commission (BioSecure NoE; FP6 IST-2002-507634).

REFERENCES

[1] R. Cappelli, "Synthetic Fingerprint Generation," *Handbook of Fingerprint Recognition,* D. Maltoni, D. Maio, A.K. Jain, and S. Prabhakar, eds. New York: Springer, 2003.
[2] R. Cappelli, A. Erol, D. Maio, and D. Maltoni, "Synthetic Fingerprint-Image Generation," *Proc. 15th Int'l Conf. Pattern Recognition,* pp. 475-478, Sept. 2000.
[3] R. Cappelli, D. Maio, and D. Maltoni, "Modelling Plastic Distortion in Fingerprint Images," *Proc. Second Int'l Conf. Advances in Pattern Recognition,* pp. 369-376, Mar. 2001.
[4] R. Cappelli, D. Maio, and D. Maltoni, "Synthetic Fingerprint-Database Generation," *Proc. 16th Int'l Conf. Pattern Recognition,* vol. 3, pp. 744-747, Aug. 2002.
[5] Y. Dit-Yan et al., "SVC2004: First International Signature Verification Competition," *Proc. Int'l Conf. Biometric Authentication,* pp. 16-22, July 2004.
[6] G. Doddington et al., "Sheep, Goats, Lambs and Wolves: A Statistical Analysis of Speaker Performance," *Proc. Int'l Conf. Language and Speech Processing,* pp. 1351-1354, Nov. 1998.
[7] J. Fierrez-Aguilar, L. Nanni, J. Ortega-Garcia, R. Cappelli, and D. Maltoni, "Combining Multiple Matchers for Fingerprint Verification: A Case Study in FVC2004," *Proc. 13th Int'l Conf. Image Analysis and Processing,* Sept. 2005.
[8] C. Wilson et al., "Fingerprint Vendor Technology Evaluation 2003: Summary of Results and Analysis Report," NISTIR 7123, Nat'l Inst. of Standards and Technology, http://fpvte.nist.gov, June 2004.
[9] D.M. Blackburn, J.M. Bone, and P.J. Phillips, "Facial Recognition Vendor Test 2000 Evaluation Report," http://www.frvt.org/FRVT2000, Feb. 2001.
[10] P.J. Phillips, P. Grother, R.J Micheals, D.M. Blackburn, E Tabassi, and J.M. Bone, "Facial Recognition Vendor Test 2002 Evaluation Report," http://www.frvt.org/FRVT2002, Mar. 2003.
[11] *Fingerprint Verification Competition (FVC2000),* http://bias.csr.unibo.it/fvc2000, 2000.
[12] *Fingerprint Verification Competition (FVC2002),* http://bias.csr.unibo.it/fvc2002, 2002.
[13] *Fingerprint Verification Competition (FVC2004),* http://bias.csr.unibo.it/fvc2004, 2004.
[14] ISO/IEC JTC1 SC 37 WG 3 Final Committee Draft 19794-2: Finger Minutiae Pattern Format, 2004.
[15] A.K. Jain, S. Prabhakar, and S. Chen, "Combining Multiple Matchers for a High Security Fingerprint Verification System," *Pattern Recognition Letters,* vol. 20, nos. 11-13, pp. 1371-1379, 1999.
[16] A.K. Jain and A. Ross, "Fingerprint Mosaicking," *Proc. Int'l Conf. Acoustic Speech and Signal Processing,* vol. 4, pp. 4064-4067, 2002.
[17] D. Maio, D. Maltoni, R. Cappelli, J.L. Wayman, and A.K. Jain, "FVC2000: Fingerprint Verification Competition," *IEEE Trans. Pattern Analysis Machine Intelligence,* vol. 24, no. 3, pp. 402-412, Mar. 2002.
[18] D. Maio, D. Maltoni, R. Cappelli, J.L. Wayman, and A.K. Jain, "FVC2002: Second Fingerprint Verification Competition," *Proc. 16th Int'l Conf. Pattern Recognition,* vol. 3, pp. 811-814, Aug. 2002.
[19] D. Maio, D. Maltoni, R. Cappelli, J.L. Wayman, and A.K. Jain, "FVC2004: Third Fingerprint Verification Competition," *Proc. Int'l Conf. Biometric Authentication,* pp. 1-7, July 2004.
[20] D. Maltoni, D. Maio, A.K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition.* New York: Springer, 2003.
[21] A. Mansfield, G. Kelly, D. Chandler, and J. Kane, "Biometric Product Testing Final Report," Issue 1.0, U.K. Nat'l Physical Lab, Mar. 2001.
[22] A. Martin, M. Przybocki, and J. Campbell, "The NIST Speaker Recognition Evaluation Program," *Biometric Systems Technology, Design and Performance Evaluation,* J. Wayman, A. Jain, D. Maltoni, and D. Maio, eds. London: Springer-Verlag, 2004.
[23] J. Matasetal, "Comparison of Face Verification Results on the XM2VTS Database," *Proc. 15th Int'l Conf. Pattern Recognition,* vol. 4, pp. 858-863, Sept. 2000.
[24] K. Messeretal, "Face Authentication Competition on the BANCA Database," *Proc. Int'l Conf. Biometric Authentication,* pp. 8-15, July 2004.
[25] Nat'l Inst. of Standards and Technology Speaker Recognition Evaluation, http://www.nist.gov/speech/tests/spk/index.htm, 2004.
[26] S.B. Panetal, "An Ultra-Low Memory Fingerprint Matching Algorithm and Its Implementation on a 32-Bit Smart Card," *IEEE Trans. Consumer Electronics,* vol. 49, no. 2, pp. 453-459, May 2003.
[27] S. Pankanti, N.K. Ratha, and R.M. Bolle, "Structure in Errors: A Case Study in Fingerprint Verification," *Proc. 16th Int'l Conf. Pattern Recognition,* 2002.
[28] P.J. Phillips, A. Martin, C.L. Wilson, and M. Przybocky, "An Introduction to Evaluating Biometric Systems," *Computer,* vol. 33, no. 2, Feb. 2000.
[29] P.J. Phillips, H. Moon, S.A. Rizvi, and P.J. Rauss, "The FERET Evaluation Methodology for Face-Recognition Algorithms," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 10, pp. 1090-1104, Oct. 2000.
[30] R. Sanchez-Reillo and C. Sanchez-Avila, "Fingerprint Verification Using Smart Cards for Access Control Systems," *IEEE Aerospace and Electronic Systems,* vol. 17, no. 9, pp. 12-15, Sept. 2002.
[31] "Best Practices in Testing and Reporting Performance of Biometric Devices," U.K. Government's Biometrics Working Group, v2.01, Aug. 2002.

Raffaele Cappelli received the Laurea degree cum laude in computer science in 1998 from the University of Bologna, Italy. In 2002, he received the PhD degree in computer science and electronic engineering from the Department of Electronics, Informatics, and Systems (DEIS), University of Bologna, Italy. He is an associate researcher at the University of Bologna, Italy. His research interests include pattern recognition, image retrieval by similarity, and biometric systems (fingerprint classification and recognition, synthetic fingerprint generation, face recognition, and performance evaluation of biometric systems).

Dario Maio received a degree in electronic engineering from the University of Bologna, Italy, in 1975. He is a full professor at the University of Bologna, Italy. He is the chair of the Cesena Campus and director of the Biometric Systems Laboratory (Cesena, Italy). He has published more than 150 papers in numerous fields, including distributed computer systems, computer performance evaluation, database design, information systems, neural networks, autonomous agents, and biometric systems. He is the author of books: *Biometric Systems, Technology, Design and Performance Evaluation*, Springer, London 2005, and the *Handbook of Fingerprint Recognition*, Springer, New York 2003 (received the PSP award from the Association of American Publishers). Before joining the University of Bologna, he received a fellowship from the CNR (Italian National Research Council) for working on the Air Traffic Control Project. He is a member of the IEEE. He is with DEIS and IEII-CNR. He also teaches database and information systems.

Davide Maltoni is an associate professor at the University of Bologna (Department of Electronics, Informatics, and Systems-DEIS). He teaches computer architectures and Pattern Recognition in computer science at the University of Bologna, Cesena, Italy. His research interests are in the area of pattern recognition and computer vision. In particular, he is active in the field of biometric systems (fingerprint recognition, face recognition, hand recognition, performance evaluation of biometric systems). He is the codirector of the Biometric Systems Laboratory (Cesena, Italy), which is internationally known for its research and publications in the field. He is the author of two books: *Biometric Systems, Technology, Design and Performance Evaluation*, Springer 2005, and the *Handbook of Fingerprint Recognition*, Springer 2003, which received the PSP award from the Association of American Publishers. He is currently an associate editor of the journals: *Pattern Recognition* and the *IEEE Transactions on Information Forensic and Security*. He is a member of the IEEE.

James L. Wayman received the PhD degree in engineering in 1980 from the University of California, Santa Barbara. He is the director of the Biometric Identification Research Program at San Jose State University. In the 1980s, under contract to the US Department of Defense, he invented and developed a biometric authentication technology based on the acoustic resonances of the human head. He joined San Jose State University in 1995 to direct the Biometric Identification Research Program, which became the US National Biometric Test Center from 1997 to 2000. He is a coeditor with J.Wayman, A. Jain, D. Maltoni, and D. Maio of the book *Biometric Systems*, (Springer, London, 2005). He holds four patents in speech processing and is a "Principle UK Expert" for the British Standards Institute on the ISO/IEC JTC1 Subcommittee 37 on biometrics. He is a member of the US National Academies of Science/National Research Council Committee "Whither Biometrics?" and previously served on the NAS/NRC "Authentication Technologies and their Implications for Privacy" committee.

Anil K. Jain received the BTech degree from the Indian Institute of Technology, Kanpur, in 1969 and the MS and PhD degrees from The Ohio State University in 1970 and 1973, respectively. He is a University Distinguished professor in the Departments of Computer Science and Engineering at Michigan State University. He served as the department chair during 1995-1999. His research interests include statistical pattern recognition, data clustering, texture analysis, document image understanding, and biometric authentication. He received awards for best papers in 1987 and 1991, and for outstanding contributions in 1976, 1979, 1992, 1997, and 1998 from the Pattern Recognition Society. He also received the 1996 *IEEE Transactions on Neural Networks* Outstanding Paper Award. He was the Editor-in-Chief of the *IEEE Transactions on Pattern Analysis and Machine Intelligence* (1991-1994). He is a fellow of the IEEE, the ACM, and the International Association of Pattern Recognition (IAPR). He has received a Fulbright Research Award, a Guggenheim fellowship, and the Alexander von Humboldt Research Award. He delivered the 2002 Pierre Devijver lecture sponsored by the International Association of Pattern Recognition (IAPR) and received the 2003 IEEE Computer Society Technical Achievement Award. He holds six patents in the area of fingerprint matching, and he is the author of a number of books: *Biometric Systems, Technology, Design and Performance Evaluation* (Springer 2005), *Handbook of Face Recognition* (Springer 2005), *Handbook of Fingerprint Recognition* (Springer 2003) (received the PSP award from the Association of American Publishers), *BIOMETRICS: Personal Identification in Networked Society* (Kluwer 1999), *3D Object Recognition Systems*, (Elsevier 1993), *Markov Random Fields: Theory and Applications* (Academic Press 1993), *Neural Networks and Statistical Pattern Recognition* (North-Holland 1991), *Analysis and Interpretation of Range Images* (Springer-Verlag 1990), *Algorithms For Clustering Data* (Prentice-Hall 1988), and *Real-Time Object Measurement and Classification* (Springer-Verlag 1988). He is an associate editor of the *IEEE Transactions on Information Forensics and Security* and is currently serving as a member of the study team on Whither Biometrics being conducted by the National Academies (CSTB).