# An Intelligent Sensor for Fingerprint Recognition

Salvatore Vitabile[2,3], Vincenzo Conti[1], Giuseppe Lentini[1], and Filippo Sorbello[1,3]

[1] Dipartimento di Ingegneria Informatica, Universita' di Palermo
Viale delle Scienze, Edificio 6, 90128, Palermo, Italy
{conti, sorbello}@unipa.it, {lentini}@csai.unipa.it

[2] Dipartimento di Biotecnologie Mediche e Medicina Legale, Universita' di Palermo
via del Vespro, 90127, Palermo, Italy
{vitabile}@unipa.it

[3] Istituto di CAlcolo e Reti ad alte prestazioni, Italian National Research Council
Viale delle Scienze, Edificio 11, 90128, Palermo, Italy

**Abstract.** In this paper an intelligent sensor for fingerprint recognition is proposed. The sensor has the objective to overcome some limits of the fingerprint recognition software systems, as elaboration time and security issues related to fingerprint transmission between sensor and processing unit. Intelligent sensor has been prototyped using the Hamster Secugen sensor for image acquisition and the Celoxica RC1000 board, employing a Xilinx VirtexE2000 FPGA, for image processing and analysis. Resources used, elaboration time as well the recognition rates in both verification and identification modes are reported in the paper. To the best of our knowledge, this is the first implementation for a full hardware implemented fingerprint recognition system.

## 1 Introduction

Biometric based systems for personal identification are always an open research issue. In literature many approaches have been proposed to develop fingerprint recognition systems. Generally, they are characterized by three main steps: image acquisition, 'biometric signature' extraction, matching between the acquired biological signature and the stored one.

Fingerprint minutiae extraction task is a very critical and complex step, so, different dedicated software algorithms have been proposed in literature [1], [2], [4], [5], [6], [7], [8], [9].

In this paper an intelligent hardware sensor for fingerprint recognition is proposed. Sensor prototype has been developed using the Celoxica RC1000 board [12]. The board employs a 2M gates Xilinx VirtexE FPGA [13]. The sensor implements ad hoc image processing algorithms selected evaluating both their performance when implemented in fixed point arithmetic and their requested hardware resources.

The proposed intelligent sensor is composed by a *Sensor Acquisition Module*

(SAM) and a *Sensor Processing Module* (SPM). The first one is based on Hamster Secugen sensor [19] for fingerprint image acquisition. The second one is an FPGA based prototype implementing the whole fingerprint recognition chain. Modules have been installed on a standard workstation and their communication exploits standard PCI bus.

The proposed system has been tested using 384 fingerprints belonging to 96 different people, and the F.A.R. (False Acception Rate) and the F.R.R (False Rejection Rate) have been used to verify its performance. Experimental trials shows that an interesting working point could be reached by the system with a F.A.R. of about 1% with the related F.R.R. of 8%.

The proposed system can be employed as an automatic fingerprint discriminator, too. The system has been evaluated with an identification test where each fingerprint has been compared with each database item in order to find a similarity index. The obtained results show that the processed image is in the subset composed by the 5 most similar fingerprints with a percentage of 84%.

The paper is organized as follow. Some related works are briefly described in section 2, whilst in section 3 some guidelines for algorithms profiling in terms of execution time and FPGA resources are presented. The proposed system as well as each processing phase are described in section 4. In section 5, both system elaboration times and recognition rates are presented. Finally, in section 6 the conclusion of this work is reported.

## 2 Related Works

In literature many approaches have been proposed and many software systems have been implemented to develop fingerprint based recognition system [1], [2], [5], [6], [8]. Generally, these systems exploit filters and image enhancement algorithms [21], classification algorithms and matching techniques and they are developed with standard high level programming languages on general purpose computers.

In [10] an hardware fingerprint recognition system is presented. However, in the system the fingerprint matching phase has not been developed. The rest of the fingerprint processing tasks were implemented in a FPGA device with a clock frequency of 27,65 Mhz, and a processing time of 589,46 ms.

## 3 The Hardware Design Guidelines

FPGA devices are widely used for rapid system prototyping. However, an efficient image processing algorithms implementation on FPGA requires an algorithms profiling phase before their implementation. Fingerprint processing algorithms have been analyzed and evaluated in order to optimize FPGA used resources, system elaboration time and result accuracy.

With the respect the FPGA requested resources, each image processing algorithm must be evaluated through the number of loop, the presence of recursion, the number of divisions, square roots operators, and powers different from 2, the presence and the dimension of typical high level language structures as "union", "array", and "circular lists" [3]. Further analysis on image processing algorithms concerns their inclination for parallel and/or pipeline implementation.

The above points are very critical for an high efficient system implementation and they concern each phase of the identification system. The Quantitative Index, *QI*, to individualize the *a priori* performance of the fingerprint processing algorithms for the embedded solution is now introduced:

$$QI = X + 0.6 * Y + 0.3 * Z + 0.05 * W \tag{1}$$

where $X$ is the number of the loops in the algorithm, Y is the number of recursion and dynamic structures in the algorithm, Z is the number of division and square root operators and powers different from 2 in the algorithm, and W is the number of union and array data structure in the algorithm.

For each phase of the fingerprint processing tasks, several algorithms have been profiled, modified and re-profiled in order to optimize system performance with respect to the hardware resources, processing time and recognition rate.

## 4 The Intelligent Sensor

As pointed out before, most of the proposed solutions are developed with standard high level programming languages on general purpose computers. In this paper the authors present an intelligent sensor that is able to acquire a fingerprint image, process it and select the corresponding database item for person identification. The sensor is composed by a *Sensor Acquisition Module* (SAM) and a *Sensor Processing Module* (SPM). The first one is based on Hamster Secugen sensor [19] for fingerprint image acquisition. The second one is a FPGA based prototype implementing the whole fingerprint recognition chain.

With more details the SPM is based on five sequential phases: the normalization phase; the binarization phase; the thinning phase; the minutiae extraction phase and the matching phase. In Figure 1 the SAM, the SPM and their relative connections with the Host buses are depicted. The SAM, based on Hamster Secugen sensor [19], acquires a fingerprint image. Successively the SAM transfers the acquired image to the SPM using both the host expansion bus and the host PCI bus. The SPM prototype has been developed on the RC1000 Celoxica board [12] equipped with a 2M gates Xilinx VirtexE FPGA [13]. SPM communications use only the host PCI bus and its clock has been set to 90/4 MHz in order to guarantee the correct data exchange between FPGA and board RAM. The RAM is used to store the fingerprint database, i.e. fingerprint image coding.

Exploiting the high data parallelism of the application, different algorithms have been parallelized. In addition, fingerprint processing phases have been pipelined in order to increase execution time as well as the final throughput. In what follows, the FPGA implementation of the five sequential phases for fingerprint recognition will be described.

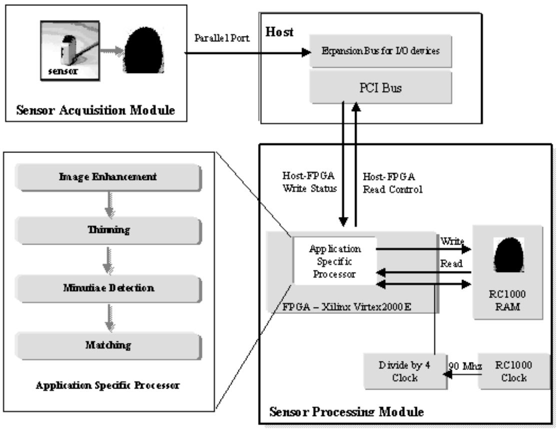

**Fig. 1.** In the figure is depicted the Sensor acquisition Module, the Sensor Processing Module and their communication with Host and Celoxica board

### 4.1 Fingerprint normalization on FPGA

In this phase the undesirable fingerprint faults are reduced, because they can produce analysis mistakes [15]. The sensor fingerprint images could have low quality due to the non-uniform contact between user finger and sensor. Consequently, an adaptive normalization algorithm [14] based on local property of the fingerprint image considered is adapted for its efficient digitalization.
The sensor fingerprint image has been sub-sampled to reduce processing time eliminating the redundant information about the thickness of the ridges. So, the 300X260 pixels sensor image have been sub-sampled obtaining a 150x260 pixels image. Exploiting data parallelism of the adaptive normalization algorithm [14], each sensor image have been divided in four 75X130 sub-images for parallel processing, since the RC1000 board is equipped with four RAM memory banks. Sensor images are codified with 256 grey levels. We use 8 bit for each pixel storing four pixels in each RAM cell (32 bit).
The adaptive normalization algorithm [14] is based on four parameters: $M_0$, $VAR_0$, M and VAR. Experimental trials conducted on our sensor images show that they range around fixed values: 100, 250, 38, 5190, respectively. Fixing the above values, the equations to calculate the normalized pixel becomes more simple without the use of the square root and division operators:

$$G(i,j) = M_0 + (I(i,j) - M) << 2 \qquad if \qquad I(i,j) \geq M \tag{2}$$

$$G(i,j) = M_0 - (I(i,j) - M) << 2 \qquad if \qquad I(i,j) < M \tag{3}$$

where I(i,j) is the intensity of the pixel at the $i_{th}$ row and $j_{th}$ column and G(i,j) the relative normalized value.

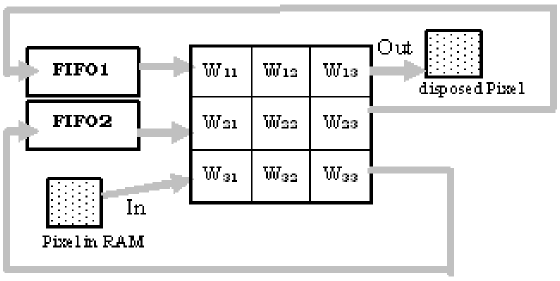

**Fig. 2.** An example of a pipelined 3x3 moving window implemented to perform the image binarization process

### 4.2 Fingerprint binarization on FPGA

The main problem of this phase is a good threshold value selection for image binarization, because a wrong threshold value could erase real minutiae in the fingerprint image. In our solution two algorithms are used to improve the binarization step: the median filter algorithm [14], and the iterative thresholding algorithm [15].

The median filter is a non-linear digital filter that is able to preserve sharp signal and to remove impulse noise. With respect iterative thresholding algorithm, the analysis reported in [1], [2], shows that the best results are obtained using two thresholds S1 and S2. The thresholding operation is applied using an iterative method with a moving 3x3 kernel that find a solution for pixels whose intensity is greater than S1 and less than S2, with S1¡S2.

In the FPGA algorithms implementation, the main problem is the median filter implementation for its high computational cost. A 7x7 moving window was used to implement the median filter in the FPGA. The moving window was stored in a FIFO buffer and FPGA execution was pipelined to improve performance.

Successively, the median filter output pixels were processed by the thresholding operation. Experimental trials had shown that S1=140 and S2=170 give good binarization results. Following the median filter approach, a 3x3 moving window was implemented to complete the image binarization process.

In Figure 2 the pipelined 3x3 moving window elaboration is depicted: at each time the processed pixel is in the position (2,2). However each pixel starts its elaboration from the position (3,1) and two FIFO buffers are used to move it among the rows until the position (1,3).

### 4.3 Fingerprint thinning on FPGA

Most of recent thinning algorithms were designed to optimize the execution time since they usually require a very high processing time. The techniques proposed in literature are based on morphological algorithms, sequential algorithms, parallel algorithms. Morphological thinning algorithms present good elaboration time but too many resources are requested for their implementation. Sequential thinning algorithms show bad elaboration performances. This limit is overcome by the parallel thinning algorithms.

Following the proposed guidelines, the MB2 algorithm [20] and the Zhang-Suen algorithm [16] were profiled. The first algorithm uses the recursive operation, with an high resources consumption. The second algorithm is based on moving kernels without any recursion. In addition, the algorithm can be parallelized since for a faster execution.

In the FPGA implementation, the binary image was split in four sub-images (we exploit the RC1000 memory banks). Sub-images thinning based on moving kernels was pipelined as shown for the binarization phase.

### 4.4 Fingerprint minutiae extraction on FPGA

The minutiae, i.e. ridge endings and ridge bifurcations, are usually detected in the thinned fingerprint image [11]. Due to the presence of either original noise or preprocessing caused noise, the thinned image contains a large number of false minutiae.

Among the profiled algorithms, the Tico and Kuosmanen one [17] extracts every true minutia in one step without any noise reduction step. The same authors in [18] present a new version of their algorithm to remove the false minutiae caused by broken ridges into a 9x9 window.

The Tico and Kuosmanen algorithm works on adjacent matrixes of pixels so it was immediately pipelined in the related FPGA implementation. Algorithm performance were improved adding the false minutiae elimination in the boundary of the image. However, the new version of the Tico and Kuosmanen algorithm leaves an high number of false minutiae caused by broken ridges. Using the direction of every ridge-ending, the number of the erased false minutiae increases significantly.

### 4.5 Fingerprint matching on FPGA

Several matching algorithms proposed in literature have been profiled. Algorithms based on pattern matching give the best performance due their possibility of parallel implementation. With more details, the algorithms proposed in [1], [2], were adapted for their efficient digitalization.

A fingerprint minutiae descriptor is used to implement the pattern matching. With more details, a list of both minutiae spatial coordinates and minutiae direction are used as descriptor to codify the fingerprint image. The algorithm receives as input two descriptors extracted from the enrolled fingerprint image and from the on-line processed image. As result, the algorithm gives a matching score with the similarity degree of the two fingerprints. A threshold is applied to the above matching score to decide if the processed fingerprints belong to the same person.

# 5 Experimental Results

## 5.1 Hardware and Software Environment

The proposed intelligent sensor is composed by a *Sensor Acquisition Module* and a *Sensor Processing Module*. The first one is based on Hamster Secugen sensor [19] for fingerprint image acquisition. The second one is an FPGA based prototype implementing the whole fingerprint recognition chain. Modules have been installed on a standard workstation and their communication exploits standard PCI bus. Sensor prototype has been developed using the Celoxica RC1000 board [12]. The board employs a 2M gates Xilinx VirtexE FPGA [13]. FPGA programming has been performed using both Celoxica DK2 [12] and Xilinx ISE [13] development environments. Algorithms have been described using the an algorithmic-like hardware programming language: the Handel-C language [12].

## 5.2 Elaboration Time

The board work frequency is limited by the work frequency of the RAM memory banks (see Figure 1). Celoxica RC1000 board manual suggests 25 MHz as maximum work frequency for board memory. We have performed several reading/writing data tests with the board memory banks. A clock frequency of 90/4 MHz, with 90 MHz the main board frequency, has assured proper and correct memory operations.
In the Table 1 some information about the implemented pipelined steps are illustrated in detail. With more details, the processed pixels mask dimension, the latency and running clock cycles, the real board work frequency and the obtained execution time are reported for each of the 5 processing steps.

**Table 1.** Processed mask dimension, latency and running clock cycles, real board work frequency and obtained execution time for each of the 5 processing steps

| Processing Steps | Latency Clock Cycles | Running Clock Cycles | Working Freq.(MHz) | Execution Time(ms) |
|---|---|---|---|---|
| Normalization | 8 | 6 | 20.9 | 12.5 |
| Binarization | 60 | 10 | 20.9 | 18.6 |
| Thinning | 83 | 73 | 21.1 | 39.8 |
| Minutiae Detection | 95 | 13 | 20.6 | 24.3 |
| Matching | 23 | 21 | 20.5 | 6.75 |

## 5.3 Resources Analysis

As pointed out before, both Celoxica DK2 [12] and Xilinx ISE [13] development environments have been used for FPGA programming. Algorithms have been described using the Handel-C language [12]. The Map Report tool inside the

Xilinx ISE [13] development environment gives an output with the FPGA resources used by the fingerprint elaboration chain. Table 2 summaries the used resources for the 2M gates Xilinx VirtexE FPGA [13].

**Table 2.** FPGA resources used by the fingerprint elaboration chain

| **Resources Type** | **FPGA Total Resources** | **Used Resources** | **Used Resources (%)** |
|---|---|---|---|
| GCLK | 4 | 2 | 50 |
| GCLKIOB | 4 | 1 | 25 |
| IOB | 404 | 273 | 67 |
| SLICE | 19200 | 16178 | 84 |
| LUT | 38400 | 27027 | 70 |
| BLOCKRAM | 160 | 6 | 3 |

### 5.4 Recognition Rates

In the registration phase, a sequence of 4 fingerprints of the same finger was acquired in different days. So for each person 4 biological signatures or descriptors were extracted and stored in the 4 board memory banks.

In the matching phase, the on-line extracted biological signature is compared with the registered signatures for a typical verification. With more details, the proposed sensor has been evaluated in both verification tasks (a username is used to select the database item and perform a 1-¿1 match) and identification tasks (no username is used so 1-¿all matches must be performed in order to select the highest matching score candidate).

Due the sensor design features, four security levels could be chosen on the basis of the matched biological signature number: the *basic security level*, if the on-line extracted biological signature matches one of the four stored signatures; *normal security level*, if the on-line extracted biological signature matches two of the four stored signatures; *high security level*, if the on-line extracted biological signature matches three of the four stored signatures; *very high security level*, if the on-line extracted biological signature matches each of the stored signatures. Exploiting both Celoxica RC1000 memory banks and the *par* feature of the Handel-C language, the four fingerprint matches are parallel executed. Experimental trials have demonstrated that the *normal security level* is the best trade-off between security and performance.

Intelligent sensor recognition performance were evaluated using the following indexes: F.A.R. (False Acception Rate) and the F.R.R (False Rejection Rate). Experimental tests were executed on a sample of 384 fingerprints taken by the Secugen sensor [19] belonging to 96 different individuals.

The first set of experiments was conducted to evaluate the performance of the proposed intelligent sensor in verification mode. So with a matching score of 95%, intelligent sensor performance are summarizes by FAR=1,07% and FRR=8,33%. In Figure 3 are plotted FAR and FRR vs the matching score for the normal

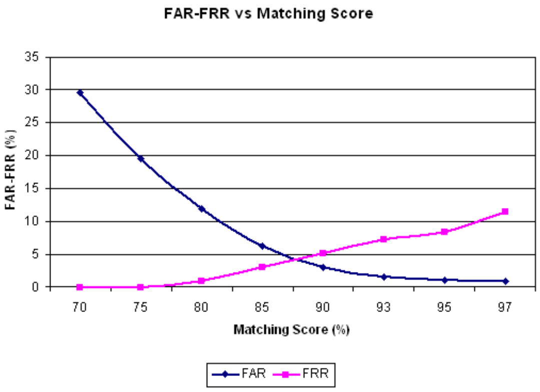

**Fig. 3.** FAR and FRR vs the matching score for the normal security level

security level. With the second set of experiments we address the problem of individual identification in large fingerprint database. Experiments aim with a database subset selection, composed by only 5 fingerprints, that always contains the processed fingerprint. In the best case, a recognition rate of 84% with a maching score of 93% is obtained, i.e. the processed fingerprint is among the 5 selected ones.

## 6 Conclusion

In this paper an hardware intelligent sensor for fingerprint recognition is proposed. The sensor has been tested with 384 fingerprints belonging to 96 different people. Sensor performance have been evaluated with F.A.R. and F.R.R indexes in both verification and identification modes. In the verification mode, a F.A.R. of about 1% and a F.R.R. of about 8% are obtained. The sensor has been used as fingerprint discriminator, too. Each processed fingerprint is among the set of 5 similar fingerprints with recognition rate of 84%. The obtained experimental results are comparable with the full software recognition systems. In Bonato at al. [10] an incomplete hardware implemented recognition system (the matching process was not implemented) was proposed. System elaboration time was about 590 ms, while the proposed sensor elaboration time is about 102 ms, considering the full fingerprint matching process.

## References

1. V. Conti, G. Pilato, S. Vitabile, F. Sorbello, *A Robust System for Fingerprints Identification*, Knowledge-Based Intelligent Information Engineering System and Allied Technologies, Crema September 2002, pp. 1162-1166

2. V. Conti, G. Pilato, S. Vitabile, F. Sorbello, *Verification of Ink-on-paper Fingerprints by Using Image Processing Techniques and a New Matching Operator*, AI*IA September 2002, pp. 594-601
3. S. Vitabile, A. Gentile, S.M. Siniscalchi, F. Sorbello, *Efficient Rapid Prototyping of Image and Video Processing Algorithms*, Proc. of EUROMICRO Symposium on Digital System Design - Architectures, Methods and Tools, Rennes 2004, pp. 452-457, IEEE Computer Society Press.
4. Jain A., *A Multichannel Approach to FingerPrints Classification*, IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol.21, n.4, 1999, pp. 348-358
5. Jain A., *On-Line Fingerprint Verification*, IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol.19, n.4, 1997, pp. 302-314
6. H. Lin, *Fingerprint Image Enhancement*, IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol.20, n.8, 1998, pp. 777-789
7. Prabhakar S., Jain A., W. Jianguo, *Minutiae Verification and Classification*, Department of Computer Engineering and Science, University of Michigan State, East Lansing 1998
8. Miklos, Zsolt, Vajna, Kovacs, *A Fingerprint Verification System based on Triangular Matching and Dynamic Time Warping*, IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol.22, n.11, 2000, pp. 1266-1276
9. Cappelli R., Lumini A., Mario D., Maltoni D., *Fingerprint Classification by Directional Image Partitioning*, IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol.21, n.5, 1999, pp. 402-421
10. V. Bonato, R.F. Molz, J.C. Furtado, M.F. Ferro, F.G. Moraes, *Propose of a hardware implementation for fingerprint systems*, UNISC - Departamento de Informatica Santa Cruz-Brazil, PUCRS - Faculdade de Informatica porto Alegre - Brazil
11. D. Maio, D. Maltoni, *Direct Gray-Scale Minutiae Detection in Fingerprints*, IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol.19, n.1, 1997
12. Celoxica Ltd.[on line], http:// www.celoxica.com
13. Xilinx Inc. [online], http:// www.xilinx.com
14. L. Hong, Y. Wan, A. Jain, *Fingerprint Image Enhancement: Algorithm And Performance Evaluation*, IEEE Transactions On Pattern Analysis And Machine Intelligence, 1998, Vol.20, N8, pp. 777-789
15. I. Emiroglu, M.B. Akhan, *Pre-Processing of FingerPrint Images*, European Conference on Security and detection, Conference Publication n.437, IEEE 1997
16. T.Y. Zhang, C.Y. Suen, *A fast parallel algorithm for thinning digital patterns*, Comm. ACM., 27(3) pp. 236-239, 1984
17. M. Tico, P. Kuosmanen, *An Algorithm for Fingerprint Image Postprocessing*, IEEE Transactions On Pattern Analysis And Machine Intelligence, pp.1735-1739, 2000
18. M. Tico, P. Kuosmanen, *Fingerprint Matching using an orientation-based Minutia Descriptor*,IEEE Transactions On Pattern Analysis And Machine Intelligence, Vol.25, no.8, 2003
19. *SecuGen FDx Developer's Gui*, Copyright 1998-2001 SecuGen Corporation and NITGen Co., Ltd. http://www.secugen.com/
20. T.M. Bernard, A. Manzanera, *Improved Low Complexity Fully Parallel Thinning Algorithm*, International Conference on Image Analysis and Processing, pp. 215-220, Venice, Italy, September 1999
21. V. Conti, G. Milici, G. Vetrano, S. Vitabile, F. Sorbello, *Fingerprint Registration Using Specialized Genetic Algorithms*, accepted in the 8th International IEEE EUROCON Conference, Belgrade, November 21-24, 2005