Fingerprint Recognition System With Hardware Acceleration

R. Leitão, A. Ferreira, T. Dias
ADEETC, ISEL – Instituto Superior de Engenharia de Lisboa, Instituto Politécnico de Lisboa, Portugal
ricardo_leitao@outlook.com aferreira@deetc.isel.ipl.pt tdias@cc.isel.ipl.pt

***Abstract*— The use of biometric systems for human authentication and identification is well known. Under the use of biometric traits for these purposes, *Fingerprint Recognition* (FR) systems are an adequate choice for many applications. In this paper, we propose a FR system based on a *Field Programmable Gate Array* (FPGA) device, an optical sensor, an image processing algorithm, and a pattern recognition module. A representation of the data with a fixed-point scheme is proposed to increase the efficiency of the FR system. The building blocks of the proposed system are analyzed in detail, regarding their performance. For the most demanding blocks, based on Gabor filtering, an efficient implementation is carried out using dedicated hardware modules; this leads to a speedup of 731.99 on the Gabor filtering module and 2.72 on the overall algorithm. For an authentication task, the proposed system attains an *Equal-Error Rate* (EER) of 7.18%.**

Keywords: fingerprint recognition, image processing, Gabor filtering, pattern recognition, FPGA, hardware accelerator, MicroBlaze.

I. Introduction

The use of *Fingerprint Recognition* (FR) systems for identification and authentication tasks is nowadays widely known. We can find *Automatic Fingerprint Identification Systems* (AFIS) in multiple and distinct hardware and software platforms, such as cell phones, laptops, access control points, or forensics tasks [2], [4]. Considering different hardware and software platforms, it is important to have FR systems with low error rates for both identification and authentication tasks. Moreover, the number of false positives (or false matches) and false negatives (or false non-matches) should also be placed on a range that is suitable for the task at hand. Other meaningful aspects are the time efficiency (for scalability issues) and the energy efficiency (in terms of consumption) of the entire FR system. For recent surveys on FR systems and techniques, please see [2] and [4] as well as the references therein.

In this paper, we propose an embedded FR system that is based on a *Field Programmable Gate Array* (FPGA) device, an optical sensor, an optimized image processing algorithm and a pattern recognition module [1], [3]. We start with a baseline implementation of the system running on a MicroBlaze softcore [6]. Then, we profile the performance of all its building blocks, which leads to a final implementation including dedicated and optimized hardware blocks. The system is then evaluated with standard procedures and measures.

The remainder of this paper is organized as follows. Section II describes the key stages of the FR system. A study on its efficiency is provided in Section III. Some experimental results and their discussion are reported in Section IV.

II. Fingerprint Recognition System

The presented FR recognition system [1], [3] operates on grayscale fingerprint images, acquired by a contact optical sensor GT-511C3 from ADH Technology. These images have a spatial resolution of 256×256 pixels, with 450 *Dots Per Inch* (DPI) and an 8 bit pixel/depth.

Each acquired image is first processed using the following 3-steps image processing algorithm:

1. Locate the *Reference Point* (RP) coordinates, i.e. the image pixel yielding the largest derivative. This implies that many of the fingerprint ridges converge to this point.
2. Establish the circular *Region Of Interest* (ROI), centered on the RP. Decompose this ROI into sectors and normalize the pixels therein, in terms of mean and variance.
3. Filter the pixels contained in the ROI, with eight band-pass and spatial oriented Gabor filters, with fine-tuned parameters. Each filter results in a different filtered image, leading to eight versions of the input image.

Fig. 1 shows the first two steps of the image processing algorithm, with the RP placed at the center and the circular sector decomposition defining the ROI. The ROI pixels are then normalized, for zero mean and unit variance.

Then, the resulting filtered images are applied to the pattern recognition module, which works as follows:

1. For each sector of each filtered image, compute the average of its pixels.
2. Place the resulting average values of each sector into a feature vector. For each filtered version, we get a feature vector with length equal to the number of sectors. Our *FingerCode*, a pattern that represents the fingerprint image, is composed of the features from the eight images.
3. For the final decision step, apply an Euclidean distance-based classifier, to compare the acquired *FingerCode* with the ones stored in the considered database.

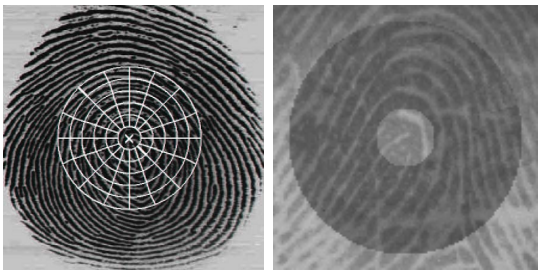

Fig. 1. Fingerprint image and its decomposition into sectors (left). Image resulting from the normalization of the ROI (right).

III. Speeding-up the Recognition System

The proposed FR system was modeled in MATLAB and implemented in software using the 'C' programming language. This 'C' code was written for the MicroBlaze processor [6], considering its implementation on a Xilinx Virtex-7 FPGA device with the configuration depicted in Fig. 2. Using both approaches, the image processing and pattern recognition modules were profiled in terms of computation resources and running time. We ran the algorithm on the MicroBlaze processor, using the Xilinx Software Development Kit (XSDK) and its profile tool. We used a counter and an interrupt controller to count the number of times the processor carried out the FR operations.

The analysis of the profile reports generated by the XSDK tool allowed us to identify the most demanding stages of the FR system and to choose the modules to be implemented using dedicated hardware blocks. In particular, the obtained profiling reports showed that about 67.11% of the total execution time was taken by the Gabor filtering procedure, followed by the floating-point arithmetic operations with 21.02%. The remaining operations took much less time. Thus, we decided to perform the following improvements on the system [1]:

1. Split the two-dimensional Gabor filter kernels into two one-dimensional kernels, through a *Singular Value Decomposition* (SVD) procedure [5], allowing a faster filtering. Also, the two-dimensional filter kernels were reduced from 33×33 to 9×9, with comparable results.
2. Replace the floating-point scheme by a fixed-point scheme, using 22 bits per filter coefficient.

Fig. 2 depicts the architecture of the proposed FR system, where the devised Gabor hardware accelerator (named filter_Gabor) is highlighted.

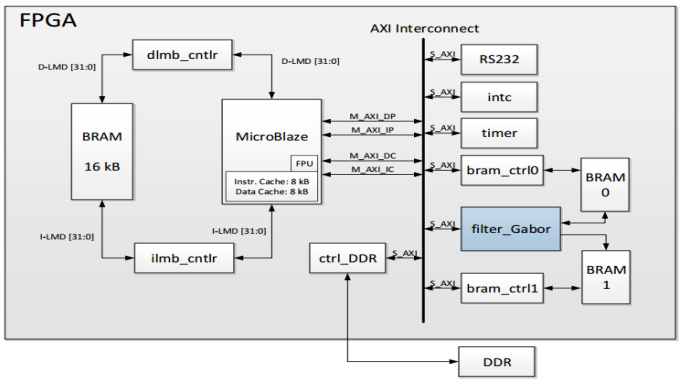

Fig. 2. Block diagram of the proposed system.

IV. Experimental Results and Discussion

The considered FR system was implemented in a Xilinx Virtex-7 FPGA VC707 Evaluation Kit [7] and evaluated using a fingerprint database (built using the optical sensor GT-511C3) with 195 images of all types of fingerprint: *left loop*, 72; *right loop*, 60; *whorl*, 54; *arch*, 6; *tented arch*, 3. Table I summarizes the FPGA implementation results.

TABLE I
Implementation results of the proposed FR system on a Xilinx Virtex-7 XC7VX485T-2FFG1761C FPGA.

Resources	FR System		filter_Gabor	MicroBlaze
Slices	4576	(6%)	477	-
LUTs (6-input)	10798	(14%)	1158	-
Flip-Flops	9377	(1.5%)	1231	-
BRAMs (36 Kb)	64	(6%)	22	10
DSP48E1	23	(< 1%)	18	5
Max. Freq.	-		266 MHz	100 MHz

The obtained experimental results allow to conclude that the two proposed key improvements (Section III) have yield a more efficient version of the FR system. Regarding the running time, our hardware implementation of the Gabor filter module has lead to a speedup of 731.99 in this filtering stage, which resulted in an overall algorithm speedup of 2.72. Also, the use of fixed-point arithmetic does not impose any loss in the attained *Equal-Error Rate* (EER), *False-Match Rate* (FMR), and *False-Non Match Rate* (FNMR) metrics. In fact, the EER attained by the proposed FR system is of 7.18%, which is adequate for many FR scenarios [2], [4]. This is shown in Fig. 3, where the EER, FMR, and FNMR are functions of the classification threshold t, considered by the classifier.

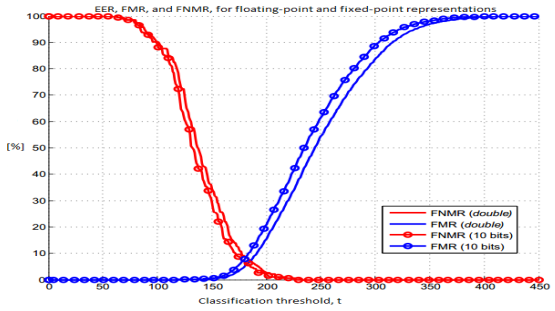

Fig. 3. The obtained EER, FMR, and FNMR, for floating-point and fixed-point representations, as functions of the threshold, t, on the Euclidean distance-based classifier.

References

[1] R. Leitão. Sistema de Reconhecimento de Impressões Digitais Baseado em FPGA, Instituto Superior de Engenharia de Lisboa, MSc thesis, November, 2014.

[2] D. Peralta, M. Galar, I. Triguero, D. Paternain, S. Garca, E. Barrenechea, J. Bentez, H. Bustince, and F. Herrera. A survey on fingerprint minutiae-based local matching for verification and identification: Taxonomy and experimental evaluation. *Information Sciences*, 315:67 – 87, 2015.

[3] S. Prabhakar. *Fingerprint classification and matching using a filterbank*. PhD thesis, Michigan State University, January 2001.

[4] A. Sankaran, M. Vatsa, and R. Singh. Latent fingerprint matching: A survey. *IEEE Access*, 2:982–1004, 2014.

[5] G. Strang and K. Borre. *Linear algebra, geodesy, and GPS*. Wellesley-Cambridge Press, 1997.

[6] Xilinx, Inc. MicroBlaze Processor Reference Guide - Embedded Development Kit EDK 14.7, UG081 (v14.7). October 2, 2013, Xilinx, Inc., Natick, San Jose, United States.

[7] Xilinx, Inc. VC707 Evaluation Board for the Virtex-7 FPGA - User Guide, UG885 (v1.3). August 22, 2013, Xilinx, Inc., Natick, San Jose, United States.