

**BAND-LIMITED PHASE-ONLY CORRELATION (BLPOC)
USING FPGA FOR FINGER VEIN RECOGNITION
SYSTEM**

TAN CHEE WEI

**UNIVERSITI SAINS MALAYSIA
2013**

**BAND-LIMITED PHASE-ONLY CORRELATION (BLPOC)
USING FPGA FOR FINGER VEIN RECOGNITION
SYSTEM**

by

TAN CHEE WEI

**Thesis submitted in fulfilment of the requirements
for the degree of
Master of Science**

September 2013

ACKNOWLEDGEMENTS

I would like to take this opportunity to express my greatest respect and appreciate to all who guided and inspired me through the duration of this thesis.

First of all, I would like to thanks to my supervisor, Dr. Bakhtiar Affendi Rosdi, who had provided and supported me all the facilities and resources required for this thesis. This work would not be done without the encouragement and guidance from him. The feedback and comments from him had helping me to improved the quality of this thesis.

Next, I would like to thank to Universiti Sains Malaysia who had provided the grants to support this research, where this work is supported by Universiti Sains Malaysia Research University Grant No. 1001/PELECT/814116 and Short Term Grant No. 304/PELECT/6039021.

Then, I would like to thank Usains Infotech Sdn. Bhd. and MyMaster program which had helping me to cover the require fees through my study.

Lastly, I would like to thanks to my parent and my family for their constant support and love. Their support and love is my greatest strength to complete my thesis.

TABLE OF CONTENTS

Acknowledgements	ii
Table of Contents	iii
List of Tables	vi
List of Figures	vii
List of Abbreviations	x
List of Symbols	xii
Abstrak	xiii
Abstract	xiv
CHAPTER 1 – INTRODUCTION	
1.1 Background	1
1.2 Problem Statement	4
1.3 Objectives	5
1.4 Research Scope	6
1.5 Research Contribution	6
1.6 Thesis Outline	6
CHAPTER 2 – LITERATURE REVIEW	
2.1 Introduction	8
2.2 Finger Vein Recognition Algorithm	9
2.3 POC and BLPOC	12
2.3.1 Fundamental of POC	12
2.3.2 Fundamental of BLPOC	13
2.3.3 Algorithm of BLPOC	14
2.4 Previously Proposed Hardware Implementation of POC and BLPOC	16
2.4.1 An FPGA-based Embedded System For Fingerprint Matching Using Phase-Only Correlation Algorithm	16

2.4.2	An embedded multi-core biometric identification system	18
2.5	2D-DFT Algorithm	20
2.5.1	Multidimensional Systolic Arrays for the Implementation of Discrete Fourier Transforms	21
2.5.2	Design of a Fully-Pipelined Systolic Array for Flexible Transposition-Free VLSI of 2-D DFT	22
2.5.3	Comparison of The Proposed 2D-DFT Algorithms	26
2.6	COordinate Rotation Dlgital Computer (CORDIC)	28
2.7	Summary	32

CHAPTER 3 – METHODOLOGY

3.1	Introduction	34
3.2	Research Methodology	34
3.3	Hardware Architecture Design	36
3.3.1	Architecture Overview	37
3.3.2	Multiplexer	41
3.3.3	2D-DFT	41
3.3.3(a)	Stage 1 of 2D-DFT	41
3.3.3(b)	Stage 2 of 2D-DFT	44
3.3.3(c)	Combined 2D-DFT	45
3.3.4	Band Limit	47
3.3.5	Proposed Reduced 2D-DFT Architecture	48
3.3.6	CORDIC: Vector Rotation	50
3.4	Performance Evaluation	54
3.5	Summary	56

CHAPTER 4 – RESULTS AND DISCUSSION

4.1	Introduction	58
4.2	Experimental Setup	58

4.3	Validation of The Proposed Architecture	59
4.3.1	Compilation	60
4.3.2	Simulation	61
4.4	FPGA Implementation	66
4.5	Performance Evaluation	68
4.6	Summary	71
CHAPTER 5 – CONCLUSION AND RECOMMENDATION		
5.1	Conclusion.....	72
5.2	Limitation and Recommendation	74
	References	75
	APPENDICES	77
	APPENDIX A – ALTERA CYCLONE III FPGA BOARD	78
	APPENDIX B – VERILOG CODE OF DFT BLOCK.....	79
	APPENDIX C – VERILOG CODE OF CORDIC BLOCK.....	83
	APPENDIX D – VERILOG CODE OF SUBTRACTER.....	88
	APPENDIX E – VERILOG CODE OF DFT FUNCTION IN BLPOC	89
	APPENDIX F – VERILOG CODE OF CORDIC FUNCTION IN BLPOC.....	118
	APPENDIX G – VERILOG CODE OF BLPOC OUTPUT	121

LIST OF TABLES

		Page
Table 2.1	Value of α_n in radian and degree	30
Table 3.1	Value of constant C (in <i>Rad.</i>) in first quadrant used in Stage 1 of 2D-DFT	43
Table 3.2	Value of constant D (in <i>Rad.</i>) in first quadrant used in Stage 2 of 2D-DFT	45
Table 3.3	Equations of general, vectoring and rotation modes of CORDIC	52
Table 3.4	Data conversion for pre- and post-processing of CORDIC	52
Table 4.1	Usage of FPGA on-chip resources	60
Table 4.2	Number of combinational functions used for each process in the proposed architecture	60
Table 4.3	Estimation of power consumption given from Quartus II software compilation	61
Table 4.4	Number of required clock cycles and processing time for each block in the proposed architecture	67
Table 4.5	Performance comparison of proposed hardware and software-based BLPOC for finger vein recognition	70

LIST OF FIGURES

		Page
Figure 1.1	(a) Finger vein imaging device proposed by Miura et al. (2004), (b) captured finger vein image	3
Figure 2.1	Dark line detection diagram by Miura et al. (2004)	10
Figure 2.2	Proposed finger vein recognition algorithm by Lee et al. (2009)	10
Figure 2.3	Flow diagram of the proposed algorithm by Mahri et al. (2010)	11
Figure 2.4	Examples of Genuine and Impostor matching using POC and BLPOC functions (Mahri et al. (2010)): (a) and (b) are finger vein images from different finger, (c) and (d) are the POC and the BLPOC results of the impostor matching for image (a) and (b), respectively; (e) and (f) are finger vein images from same finger, (g) and (h) are the POC and the BLPOC results of the genuine matching for image (e) and (f), respectively.	15
Figure 2.5	Overall flow to calculate the similarity of two images using BLPOC function	16
Figure 2.6	Proposed 2D-FFT hardware system block diagram by Danese et al. (2009)	17
Figure 2.7	Proposed matching algorithm architecture by Danese et al. (2011)	19
Figure 2.8	(a) Type 1 semi-systolic multiplication and (b) Type 2 semi-systolic multiplication by Lim and Swartzlander (1999)	22
Figure 2.9	(a) PE structure and (b) Systolic array for 2D-DFT by Lim and Swartzlander (1999)	23
Figure 2.10	(a) Dependence graph (DG1) pertaining to computation of Stage 1 of recursions for n^{th} column of the input image, (b) Mathematical representation for each node by Meher (2005)	24
Figure 2.11	(a) Dependence graph (DG2) pertaining to computation of Stage 2 of recursions for k^{th} row of the intermediate array, (b) Mathematical representation for each node by Meher (2005)	25
Figure 2.12	Dependence graph of combination of DFT Stage 1 and Stage 2 by Meher (2005)	26
Figure 2.13	Processing Element of 2D-DFT by Meher (2005)	27
Figure 2.14	Example of 2D-DFT processing elements for a 3×3 image by Meher (2005)	27
Figure 2.15	Iterative CORDIC structure by Andraka (1998)	29

Figure 3.1	Research flow of hardware implementation of BLPOC for finger vein recognition system	36
Figure 3.2	Overall view of the proposed hardware architecture to calculate the similarity of two finger vein images based on BLPOC.	38
Figure 3.3	The operation flowchart of the proposed hardware architecture.	39
Figure 3.4	The operation flowchart of the Registration mode.	39
Figure 3.5	The operation flowchart of the Matching mode.	40
Figure 3.6	Overall view of the 2D-DFT block without band limit (Number of PEs is 12288 (192×64)).	42
Figure 3.7	Block diagram for the first stage of 2D-DFT	44
Figure 3.8	Block diagram of the multiplier	44
Figure 3.9	Block diagram for the second stage of 2D-DFT	46
Figure 3.10	Block diagram for the combination of first and second stages of 2D-DFT (A PE of DFT block).	47
Figure 3.11	Overall view of the 2D-DFT block with band limit, where the number of PEs is reduced to 2496. By using the band limit ratio of 0.2, the number of required PEs in a row can be reduced from 192 to 39 ($20 + 19$).	48
Figure 3.12	Overall view of the proposed 2D-DFT block	49
Figure 3.13	Overall view of the CORDIC block which is used to convert the format of DFT coefficients between real/imaginary and phase/magnitude	50
Figure 3.14	Overall view of the CORDIC block which is used to convert the format of DFT coefficients between real/imaginary and phase/magnitude	53
Figure 3.15	Examples of the enhanced finger vein images used in the experiments adopted from Rosdi et al. (2011)	55
Figure 4.1	(a) Preprocessed finger vein image; (b) 2D-DFT result of finger vein image in (a) from MATLAB software; (c) Simulation waveform of 2D-DFT on finger vein image in (a) by DFT block.	62
Figure 4.2	(a) CORDIC Vectoring mode simulation waveform with first quadrant input; (b) CORDIC Vectoring mode simulation waveform with fourth quadrant input;	63
Figure 4.3	(a) CORDIC Rotation mode simulation waveform with positive input; (b) CORDIC Rotation mode simulation waveform with negative input;	64
Figure 4.4	Examples of (a) the finger vein images from the same finger and their genuine matching result using (b) MATLAB and (c) Quartus II.	65

Figure 4.5	Examples of (a) the finger vein images from two different fingers and their imposter matching result using (b) MATLAB and (c) Quartus II.	66
Figure 4.6	Architecture designed overview for performance evaluation purpose.	67
Figure 4.7	Board's view of proposed architecture implemented on Altera Cyclone III FPGA.	68
Figure 4.8	(a) Matching score graph of FPGA-based system; (b) Matching score graph of MATLAB-based system.	69
Figure 4.9	(a) EER graph of FPGA-based system; (b) EER graph of MATLAB-based system.	70
Figure A.1	Board view of proposed architecture implemented on Altera Cyclone III FPGA.	78

LIST OF ABBREVIATIONS

POC	Phase Only Correlation
BLPOC	Band Limit Phase Only Correlation
2D-DFT	Two Dimension Discrete Fourier Transform
2D-FFT	Two Dimension Fast Fourier Transform
DFT	Discrete Fourier Transform
iDFT	Inverse Discrete Fourier Transform
1D-DFT	One Dimension Discrete Fourier Transform
2D-iDFT	Two Dimension Inverse Discrete Fourier Transform
FPGA	Field-programmable gate array
PC	Personal computer
CPU	Central processing unit
ROI	Region of interest
CLAHE	Contrast limited adaptive histogram equalization
DSP	Digital signal processing
EER	Equal error rate
FRR	False rejection rate
FAR	False acceptance rate
PE	Processing element

- DG** Dependence graph
- CORDIC** Coordinate Rotation Digital Computer
- HDL** Hardware description language
- RTL** Register-transfer level
- GUI** Graphical user interface
- CAD** Computer-aided design
- EDA** Electronic design automation
- LCD** Liquid crystal display
- USB** Universal Serial Bus
- LED** Light-emitting diode

LIST OF SYMBOLS

π ρi

θ angle in Radians

KOLERASI FASA JALUR TERHAD MENGGUNAKAN FPGA UNTUK SISTEM PENGECAMAN VENA JARI

ABSTRAK

Kebelakangan ini, disebabkan corak vena jari mempunyai keselamatan yang tinggi dan boleh dipercayai, ia telah menjadi salah satu topik hangat dalam penyelidikan biometrik. Dalam tahun-tahun kebelakangan ini, beberapa algoritma pengesanan vena jari telah dicadangkan. Namun begitu, algoritma-algoritma tersebut dilaksanakan dengan menggunakan perisian yang berasaskan komputer. Pelaksanaan ini mempunyai kelemahan dalam kelajuan pemrosesan, saiz dan penggunaan kuasa. Untuk mengatasi kelemahan ini, tesis ini membentangkan satu seni bina bagi sistem pengesanan vena jari berasaskan kaedah pemadanan BLPOC. BLPOC adalah kaedah pemadanan yang berasaskan fasa yang berketepatan tinggi dan kurang dipengaruhi oleh perubahan kedudukan imej dan juga perubahan tahap pengcahayaan. Ia melibatkan proses pengiraan yang tinggi, iaitu 2D-DFT, maka, ia perlu dilaksanakan melalui perkakasan seperti FPGA. Seni bina ini telah berjaya dilaksanakan dengan menggunakan Verilog HDL dan disahkan dengan menggunakan Altera Cyclone III EP3C120F780 FPGA. Blok DFT yang dicadangkan telah berjaya mengurangkan saiz sebanyak 97% daripada saiz asal. Sebuah pengkalan data yang mengandungi imej vena jari daripada 204 kelas telah digunakan untuk menilai prestasi seni bina yang dicadangkan. Keputusan menunjukkan bahawa seni bina yang dicadangkan berupaya memproses pemadanan tunggal dua imej vena jari dalam masa 1.15 ms, yang merupakan kira-kira sembilan kali lebih pantas daripada pelaksanaan berasaskan perisian, dengan ketepatan yang setanding dengan pelaksanaan berasaskan perisian. Kesimpulannya, sistem pengesanan vena jari berasaskan BLPOC telah berjaya direkabentuk menggunakan FPGA, dimana ia berupaya memproses dengan lebih pantas berbanding dengan teknik berasaskan perisian.

BAND-LIMITED PHASE-ONLY CORRELATION (BLPOC) USING FPGA FOR FINGER VEIN RECOGNITION SYSTEM

ABSTRACT

Nowadays, due to the high security and reliable of finger vein pattern, it had become one of the major interests in the biometric research. In the last few years, a number of finger vein recognition algorithms have been proposed. Most of the proposed methods were implemented in software-based on a general-purpose processor, which have limitations on the processing speed, size and power consumption. To overcome these limitations, this thesis presents an architecture for finger vein recognition system based on BLPOC matching method. The BLPOC is a phase-based matching method which have benefits of high accuracy and less affected by image shifted or brightness changed. It involves a high computation process, which is 2D-DFT, therefore, it is necessary to implement on a hardware device such as FPGA. It consists of two types of multiplexer blocks, one DFT block, one CORDIC block, seven types of memory blocks, one subtracter block, one divider block and one comparator block; and is implemented using Verilog HDL and verified using the Altera Cyclone III EP3C120F780 FPGA board. The proposed DFT block had contributed to reduce the area used by 97% of the previously proposed DFT block. A finger vein image database of 204 classes has been used to evaluate the performance of the proposed architecture. Results show that the proposed architecture can process a single matching of two finger vein images in 1.15 ms, which is about nine times faster than the software-based implementation, while the accuracy is similar with the software-based implementation. In conclusion, the finger vein recognition system based on BLPOC is successfully implemented on a FPGA board with better processing time as compared with the software-based implementation.

CHAPTER 1

INTRODUCTION

1.1 Background

Identity verification had been used by human for ages; i.e., human used it to recognize others by their name, face or voice. Nowadays, identity verification had been applied in computer systems, which can be accomplished by three methods: two conventional methods of identification based on some things the one has (i.e., ID card or key), or knowledge of secret (i.e., PIN or password) (Cavoukian et al. (2012)); and biometrics method. However, there are disadvantages on the two conventional methods, such as, ID card and key tend to get stolen or loss and password are often forgotten. To achieve more reliable identity verification (Dahiya and Kant (2012)), the third method, which is biometrics are introduced.

Biometrics is a process of automatically recognizing a person using the measurable physiological or behavioral trait that can be captured and subsequently compared with another (Austin Jay Harris (2002)). Physiological biometrics (P.K. Janbandhu (2001)) is a biometric characteristic based on individual's anatomy or detailed physiology, such as, fingerprint (Sheng et al. (2012)), iris (Proenca and Alexandre (2012)) and face (Wagner et al. (2012)). While, behavioral biometrics (Revett (2008)) is biometric characteristic based on acquired behavior over time by an individual, such as, speech (Inthavisas and Lopresti (2012)) and signatures (Ferrer et al. (2012)).

Biometrics mainly apply on verification and identification system (Sutrop and Laas-Mikko (2012)). Verification is a task where a system attempts to confirm individuals claimed identity, while the identification is a task where a system search from database and returns a correspond-

ing identity (Dahiya and Kant (2012)). There are some biometric systems, which were successfully developed, i.e., fingerprint (Ito et al. (2004)), palmprint (Ito et al. (2008)), iris (Miyazawa et al. (2008)), face (Ding et al. (2012)), voice (Drygajlo (2009)) and signature (Ferrer et al. (2012)). Notwithstanding this great and increasing variety of biometrics, no biometric has yet been developed that is perfectly reliable or secure. For example, fingerprints and palm prints are usually frayed; voice, signatures, hand shapes, and iris images are easily forged; face recognition can be made difficult by occlusions or face-lifts (Liu et al. (2010)). Besides that, biometrics such as fingerprints, iris and face recognition are susceptible to spoofing attacks, i.e., the biometric identifiers can be copied and used to create artifacts that can deceive many currently available biometric devices (Liu et al. (2010)). Finger vein pattern is just a promising qualified candidate for biometric-based personal identification (Liu et al. (2010)).

Nowadays, vascular pattern, which commonly referred as the vein pattern, has become one of the major interests in biometric research (Mahri et al. (2010)). Vein is the blood vessel inside our human body. Researchers have determined that the vessel pattern of the human body is unique to a specific individual, not affected by race and skin discolorations, and does not change as people age (Mulyono and Jinn (2008)). Due to the human vein is hidden inside skin and cannot be seen by human, thus, it requires a specific device to capture the vein pattern. Figure 1.1 shows the idea of finger vein imaging device proposed by Miura et al. (2004) and the sample of captured finger vein image. Normal visible light is not able to pass through human tissues, thus, infrared light is needed in-order to capture the vein pattern. It attributes to be a high security and reliability as a biometric due to the following benefits (Liu et al. (2010)):

- High security level: Due to the vein are hidden inside human skin, thus it is difficult to be forged and replicated.
- User acceptance: The non-invasive and contactless capture ensures both convenience

and cleanliness for the users.

- Living body identification: It requires blood to flow in veins in-order to be registered.

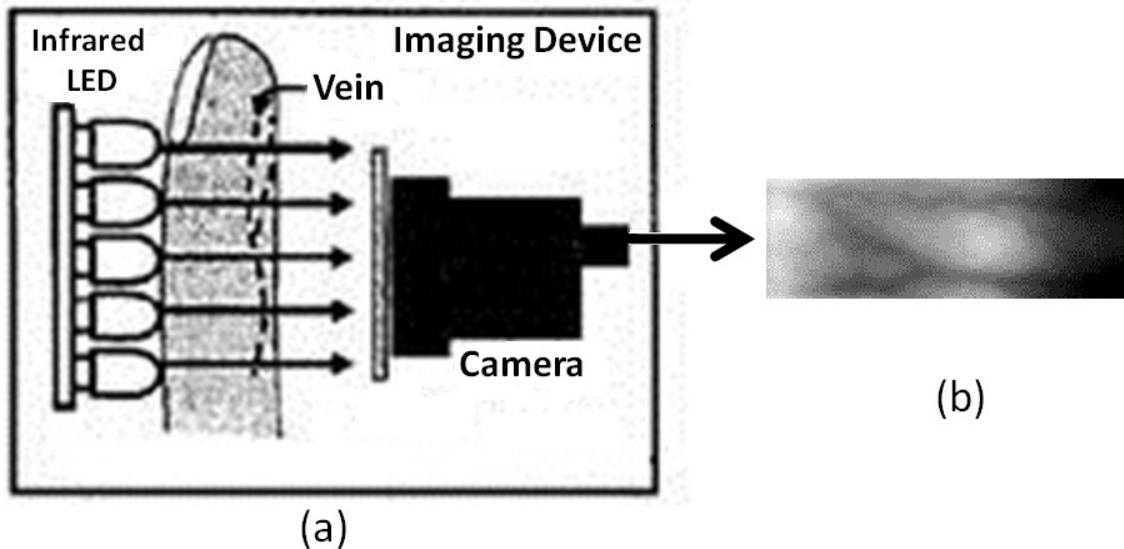


Figure 1.1: (a) Finger vein imaging device proposed by Miura et al. (2004), (b) captured finger vein image

Due to the advantages, some research had been conducted on the finger vein biometric system. Finger vein recognition system proposed by Liu et al. (2010), Lee et al. (2009) and Miura et al. (2004) have similarity on the fundamental algorithm which includes enrollment, template creation and finger vein pattern extraction in the preprocessing stage. However, according to Mahri et al. (2010), when the vein pattern is not clear, pattern extraction might become inconvenient and prone to extract inaccurate vein patterns. To overcome this problem, Mahri et al. (2010) proposed the Band-Limited Phase-Only Correlation (BLPOC) method.

BLPOC is an enhanced method of Phase-Only Correlation (POC) (Takita et al. (2003)). POC is a matching method which is using the phase components of the Two Dimension Discrete Fourier Transform (2D-DFT) of given images to check their similarities. This matching method does not include pattern extraction technique, and it provides better accuracy compare to the ordinary correlation methods. POC method has benefits which it is less affected by image shifting

and brightness change. BLPOC is a POC method which has limits the band of phase components to remove most of the meaningless components. Since most of the meaningless components are removed from matching, hence, the accuracy is increased. This phase-based image matching method already successfully applied on some biometric systems, i.e., fingerprint by Ito et al. (2004) and iris recognition by Miyazawa et al. (2008).

The matching method of BLPOC had been proved to be able to be implemented on a finger vein recognition system by Mahri et al. (2010). In this thesis, the finger vein recognition system based on BLPOC is implemented on a hardware device, which had proven to improve the speed and reduce the area used and the power consumption.

1.2 Problem Statement

Nowadays, with the growth of technology, the accuracy of a biometric system is no longer the only consideration in biometric research. With the higher security and reliable biometrics had been discovered, the speed, size, power consumption and cost become the major interest in the biometrics research.

Most of the current proposed finger vein biometrics systems are in software-based implementation, i.e finger vein biometric system proposed by Song et al. (2011) and Mahri et al. (2010) are implemented in software. According to Rakvic et al. (2009), a software-based implementation applied the algorithm on a general-purpose machine which designed for all types of applications, therefore, it will results to be slower speed, larger area, higher power consumption and cost. The previously proposed finger vein recognition system by Mahri et al. (2010) implemented the BLPOC algorithm using MATLAB software. In the proposed algorithm of BLPOC matching technique, it includes 2D-DFT process, which is a high and complex computation. Therefore, it requires a longer processing time to complete the computation task, and this results to slower

processing speed. To overcome these problems faced on software-based implementation of biometric, hardware implementation of BLPOC is proposed. Danese et al. (2011) proved that the speed of the hardware-based implementation of BLPOC significantly improved and the area and the power consumption are reduced, as compared with the software-based implementation.

There are some biometric systems that had been implemented into a hardware device; i.e., iris recognition system by Rakvic et al. (2009) and fingerprint recognition system by Danese et al. (2011). Rakvic et al. (2009) proved that the implementation of iris recognition on FPGA board had increased the speed by 19 times as compared to CPU, while Danese et al. (2011) proved that the speed of FPGA-based implementation of fingerprint recognition is 20 times faster than the software-based implementation.

Motivated by the benefits of hardware implementation and the problems of software-based implementation, this research proposed the FPGA-based implementation of finger vein recognition system based on BLPOC function.

1.3 Objectives

The objectives of this research are:

- To study the algorithm of BLPOC for finger vein recognition system.
- To develop an architecture of the finger vein recognition system based on the BLPOC and implement it on a FPGA board.
- To investigate the performance of the proposed architecture by comparing with the software-based implementation in term of accuracy and speed.

1.4 Research Scope

In this research, a finger vein recognition system based on BLPOC function is implemented on a FPGA board. The focus of this research is the BLPOC function, thus, the steps in the preprocessing stage, which are region of interest (ROI) extraction, image resizing and image enhancement will be excluded. Those three preprocessing steps are decided to be performed using MATLAB software for the whole images in the database. The finger vein image database used in this research is adopted from Mahri et al. (2010). The ROI extraction, image resizing and image enhancement of all vein images in the database are preprocessed by MATLAB software. The architecture of BLPOC function is designed according to the algorithm proposed by Mahri et al. (2010) and implemented on a FPGA board. To verify the proposed architecture, the performance is compared with the MATLAB-based implementation system in term of speed and the accuracy.

1.5 Research Contribution

In general, the focus of this research is the hardware architecture of finger vein recognition system based on BLPOC function. The main contribution of this research is the proposed hardware architecture, which had improved the speed and reduced the area and the power consumption, as compared with the software-based implementation. The proposed architecture is validated using a FPGA board.

1.6 Thesis Outline

The remainder of this thesis is organized as follows:

In Chapter 2, the overview of the finger vein biometric system and the matching methods are presented. The focus of this chapter is the phase-based matching method. The algorithm of the

phase-based matching is reviewed. In addition, the surveys on hardware implementation of the phase-based matching method are given.

Next, the methodology of the research is explained in Chapter 3. The method consists of architecture design, FPGA implementation and performance evaluation. The tools used in this research are explained in details.

Then, the architecture design of the BLPOC function is discussed in Chapter 4. The flow and the function of each component in the architecture are explained in details.

Besides that, the experimental results and analysis are shown in Chapter 5. The experiments include the simulation and FPGA implementation. The performance on the speed and the accuracy of the proposed architecture are evaluated by comparing with the MATLAB-based implementation.

Chapter 6 concluded the research and discussed the recommendation for future work. The conclusions are drawn from the experimental results and analysis. Finally, the recommendation for future work is discussed.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Biometric system is a computerized system, which recognizing an individual based on the measurable biological or behavioral characteristics. It mainly applies on security and personal identification system. There are some biometric systems, which were successfully developed, i.e., fingerprint, palmprint, iris, face, voice and signature. There is no biometric has yet been developed that is perfectly reliable or secure, i.e, fingerprints and palm prints are usually frayed; voice, signatures, hand shapes, and iris images are easily forged; face recognition can be made difficult by occlusions or face-lifts; and biometrics such as fingerprints, iris and face recognition are susceptible to spoofing attacks. Finger vein pattern is just a promising qualified candidate for biometric-based personal identification.

This chapter presents the literature review on finger vein recognition and matching methods. The main component to be reviewed for this research is the matching method, which is BLPOC function. The algorithm of the BLPOC function that is used to measure the similarity of two finger vein images is shown. The function involved the high computation component, 2D-DFT, which is the highest demand component in BLPOC architecture. The algorithm and the architecture of 2D-DFT are reviewed in details. Besides, the hardware implementation of BLPOC function in other biometric systems are reviewed in-order to compare the benefits and the method of implementation.

2.2 Finger Vein Recognition Algorithm

Due to the high security and reliability of the finger vein as biometric system, a lots of researches had been conducted to achieve a better way of processing the finger vein image for recognition system.

Miura et al. (2004) proposed finger vein feature extraction using repeated line tracking. The applying of line tracking repeatedly is proven to be able to extract a better finger vein pattern. The steps of the repeated line tracking are as follows: First, they determine the start point for line tracking and the moving-direction attribute. Next, they detect the direction of the dark line and movement of the tracking point as shown in Figure 2.1. Then, they update the number of tracked times points in the locus space and repeat the previous steps for N times. Finally, they acquire the finger vein pattern from the locus space. For the matching method, they introduced an enhanced method of template matching, which is to identify the "ambiguous region" around the veins and the slight misalignments between vein patterns in these regions are ignored. The steps of the proposed enhanced method of template matching are as follows: First, they binarize the locus space by using a threshold, which pixel with values smaller than the threshold is set to 0 and greater or equal than the threshold is set to 255. Next, spatial reduction is performed by taking the average of all non-overlapping 3×3 pixels. Then, they relabel the locus space by converting the pixel values according to a range, which pixel values in the range from 0 to 85 is converted to 0, range from 86 to 170 is converted to 128 and range from 171 to 255 is converted to 255. Finally, they calculate the mismatch ratio, which defined as the different between two sets of data to be matched.

Another method of finger vein recognition which proposed by Lee et al. (2009) is using minutia-based alignment and local binary pattern-based feature extraction. The flow of the proposed algorithm (as shown in Figure 2.2) of the finger vein recognition is as follows: First, they

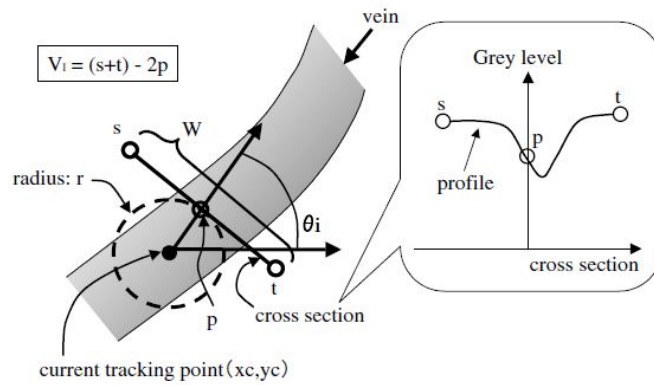


Figure 2.1: Dark line detection diagram by Miura et al. (2004)

capture the finger vein image and extract the vein pattern. Second, they localize the finger region in the captured image using a mask. Next, they stretch and subsample the vein image to improve the processing time. Then, the subsampled image is aligned using the affine model based on the extracted minutia points of a finger vein. After that, the finger vein code is extracted based on local binary pattern (LBP). Finally, they match the extracted code with the enrolled ones by calculating the Hamming distance (HD), where the HD is used to measure the dissimilarity between two images.

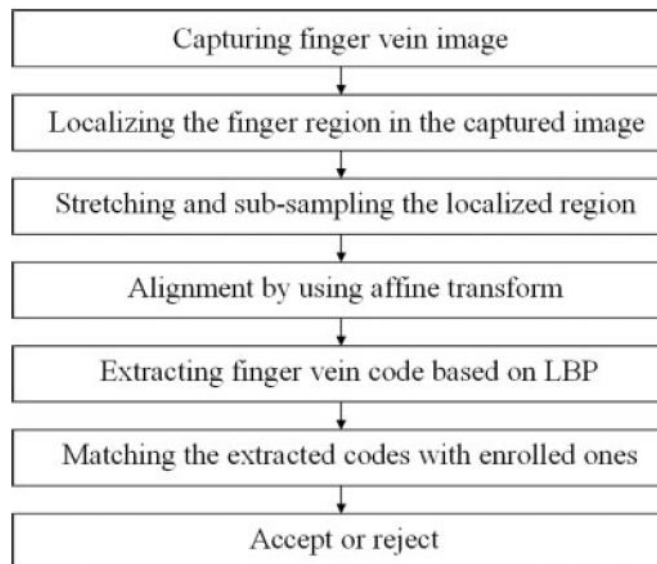


Figure 2.2: Proposed finger vein recognition algorithm by Lee et al. (2009)

Algorithms of finger vein recognition that proposed by Miura et al. (2004) and Lee et al. (2009) have a similarity on the fundamental algorithms which include the finger vein pattern extraction in preprocessing stage. However, when the vein pattern is not clear, the pattern extraction will become inconvenient and prone to extract inaccurate vein pattern (Mahri et al. (2010)). To overcome this problem, Mahri et al. (2010) had proposed a new matching technique, which using the phase components of 2D-DFT to calculate the matching score. This technique is known as BLPOC. This matching technique did not involve vein pattern extraction, thus, it results to reduce and simplified the processing steps. They proposed the algorithm (as shown in Figure 2.3) of matching process as follows: First, finger vein images are captured by their own designed capturing device. Next, they apply image normalization and ROI extraction. Then, they applied image enhancement using contrast limited adaptive histogram equalization (CLAHE) technique. Finally, the two finger vein images are matched by BLPOC matching method.

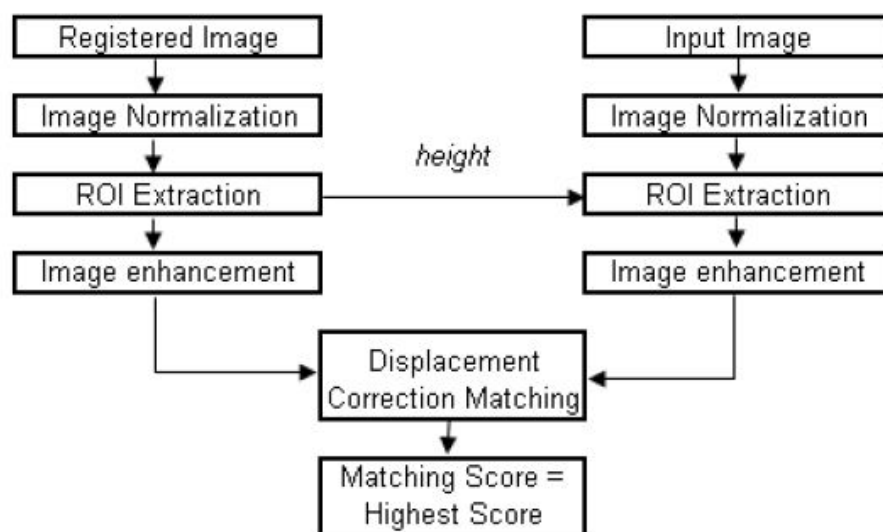


Figure 2.3: Flow diagram of the proposed algorithm by Mahri et al. (2010)

By comparing the three matching techniques of Miura et al. (2004), Lee et al. (2009) and Mahri et al. (2010), the technique proposed by Mahri et al. (2010) has benefits on simple processing step, which is excluded the step of finger vein pattern extraction. Therefore, the matching

method proposed by Mahri et al. (2010), which is BLPOC, is chosen to be used in this research for the hardware implementation.

2.3 POC and BLPOC

POC is a matching method which uses only the phase components from 2D-DFT to check the similarity of two images. This matching technique has an advantage on provide more accuracy compare with the ordinary correlation methods. Besides, it also has benefits on less affected by image shifting and brightness changing. From experiments, the accuracy of the POC matching result is low, thus, an enhanced method of POC is proposed, which known as BLPOC.

BLPOC is an enhanced method of POC, which it limits the band of the matching region to remove most of the meaningless components. In the phase components of an image, most of the meaningless components contains in the high frequency band, thus, by removing the high frequency band, the accuracy is increased (Ito et al. (2004); Miyazawa et al. (2008); Mahri et al. (2010)). The details of the POC and BLPOC are explained in the next sub-section.

2.3.1 Fundamental of POC

The definition of the POC function is described in this section. Considering two images $f(m, n)$ and $g(m, n)$, each with $(M \times N)$ fixed size, where $M = 2m + 1$ and $N = 2n + 1$. Let $F(u, v)$ and $G(u, v)$ denote the 2D-DFT of the two images. $F(u, v)$ and $G(u, v)$ are given by (Ito et al. (2004); Mahri et al. (2010)),

$$\begin{aligned}
 F(u, v) &= \sum_{x=-m}^m \sum_{y=-n}^n f(x, y) e^{-i\frac{2\pi}{M}(ux)} \times e^{-i\frac{2\pi}{N}(vy)} \\
 &= A_F(u, v) e^{i\theta_F(u, v)}
 \end{aligned} \tag{2.1}$$

$$\begin{aligned}
G(u, v) &= \sum_{x=-m}^m \sum_{y=-n}^n g(x, y) e^{-i\frac{2\pi}{M}(ux)} \times e^{-i\frac{2\pi}{N}(vy)} \\
&= A_G(u, v) e^{i\theta_G(u, v)}
\end{aligned} \tag{2.2}$$

where $A_F(u, v)$ and $A_G(u, v)$ are the amplitude of the input images and $e^{j\theta_F(u, v)}$ and $e^{j\theta_G(u, v)}$ are the phase of the input images. The cross phase spectrum, \widehat{R}_{FG} of the two images are given by,

$$\begin{aligned}
\widehat{R}_{FG}(u, v) &= \frac{F(u, v) \times \overline{G(u, v)}}{|F(u, v) \times G(u, v)|} \\
&= e^{i\theta_{FG}(u, v)}
\end{aligned} \tag{2.3}$$

where $\overline{G(u, v)}$ is the conjugate of $G(u, v)$ and the phase different, $\theta_{FG}(u, v) = \theta_F(u, v) - \theta_G(u, v)$.

The POC function of the two images is the 2D inverse DFT (2D-iDFT) of the cross phase spectrum and is given by,

$$\widehat{r}_{fg}(x, y) = \frac{1}{MN} \sum_{u=-m}^m \sum_{v=-n}^n \widehat{R}_{FG}(u, v) e^{-i\frac{2\pi}{M}(ux)} e^{-i\frac{2\pi}{N}(vy)} \tag{2.4}$$

2.3.2 Fundamental of BLPOC

The previous research, Ito et al. (2004), states that most of the meaningless phase components contain in the high-frequency domain of an image. In the POC based image matching method, all the frequency components are involved, included the meaningless components. Hence, the calculation of the similarity between two finger vein images using POC function may have less reliability. To eliminate the meaningless components, Ito et al. (2004) proposed band limited phase only correlation (BLPOC), which limits the range of the frequency bands of the images to match.

Assuming that the effective frequency band of finger vein images are given by $k_1 = -K_1, -K_1 +$

$1, \dots, K_1$ and $k_2 = -K_2, -K_2 + 1, \dots, K_2$. Hence, the size of the effective frequency spectrum is given by $L_1 = 2K_1 + 1$, and $L_2 = 2K_2 + 1$. BLPOC function is given by (Ito et al. (2004)),

$$\hat{r}_{fg}^{K_1 K_2}(n_1, n_2) = \frac{1}{L_1 L_2} \sum_{k_1=-K_1}^{K_1} \sum_{k_2=-K_2}^{K_2} \hat{R}_{FG}(k_1, k_2) \times e^{-i\frac{2\pi}{L_1}(k_1 n_1)} e^{-i\frac{2\pi}{L_2}(k_2 n_2)} \quad (2.5)$$

where $L_1 \times L_2$ is the size of BLPOC.

Figure 2.4 shows the genuine (matching of two images from the same finger) and impostor (matching of two images from two different fingers) matching examples using the POC and BLPOC functions. The matching scores of two images are determined by the peak value of the POC and BLPOC results. The higher the peak value, the higher the similarity of the two images. The maximum of the peak value of POC and BLPOC output is 1, while the minimum is 0. As we can see from the examples, the discrimination capability between the genuine and impostor matching of the BLPOC function is much higher than the POC function. This is due to the removing of most of the meaningless components in BLPOC matching method.

2.3.3 Algorithm of BLPOC

The algorithm used to calculate the similarity between two input images using BLPOC function is shown in Figure 2.5. This algorithm is based on the mathematical function of BLPOC by Ito et al. (2004) and Mahri et al. (2010); and modified from the POC algorithm which proposed by Morikawa et al. (1999). The flow of the BLPOC matching process is as follows: First, two images are inserted, which are input and registered images. The input image is the image from the input device, while the registered image is the image that registered in the database. Next, the two images are transformed from the time domain to the frequency domain by 2D-DFT. There are two types of output from 2D-DFT, which are magnitude and phase components. As BLPOC

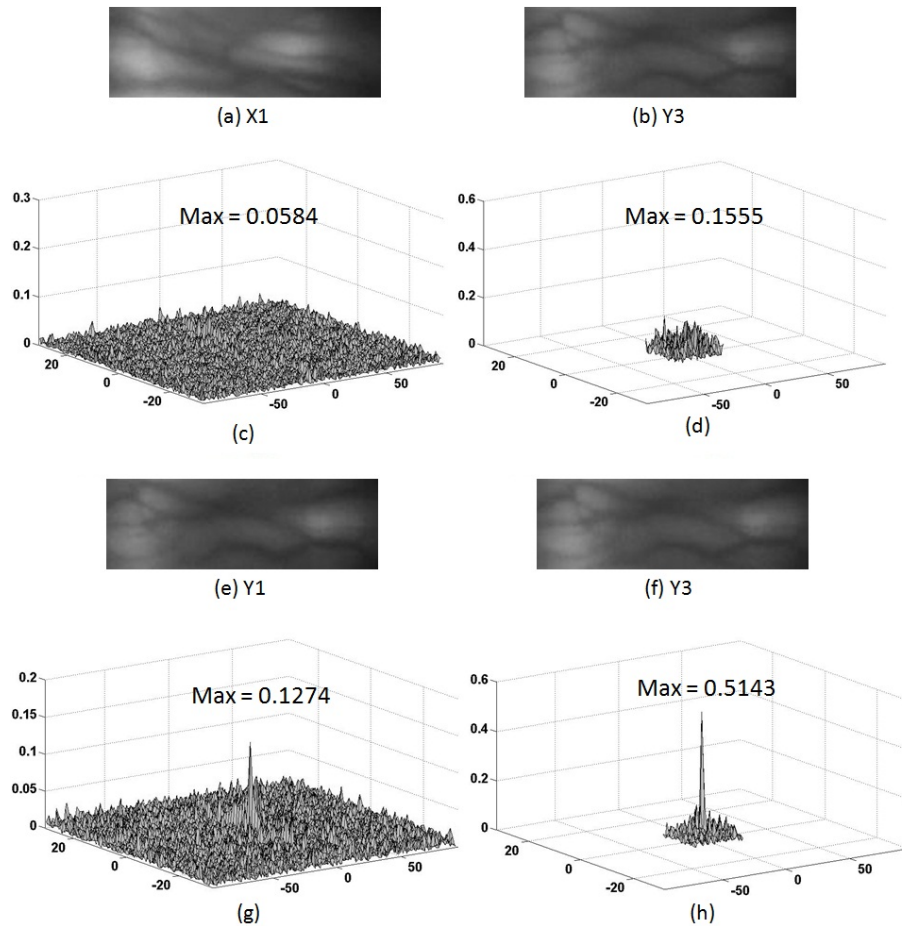


Figure 2.4: Examples of Genuine and Impostor matching using POC and BLPOC functions (Mahri et al. (2010)): (a) and (b) are finger vein images from different finger, (c) and (d) are the POC and the BLPOC results of the impostor matching for image (a) and (b), respectively; (e) and (f) are finger vein images from same finger, (g) and (h) are the POC and the BLPOC results of the genuine matching for image (e) and (f), respectively.

function only needs the phase components, thus, the magnitude components can be ignored. Then, apply the band limit by limiting the frequency band to only low frequency part, which is to remove most of the meaningless components contains in the high-frequency band. After that, subtract the phase of the input image with the registered image to get the phase different values. Finally, the phase different values are inserted into 2D-iDFT to calculate the BLPOC matching score.

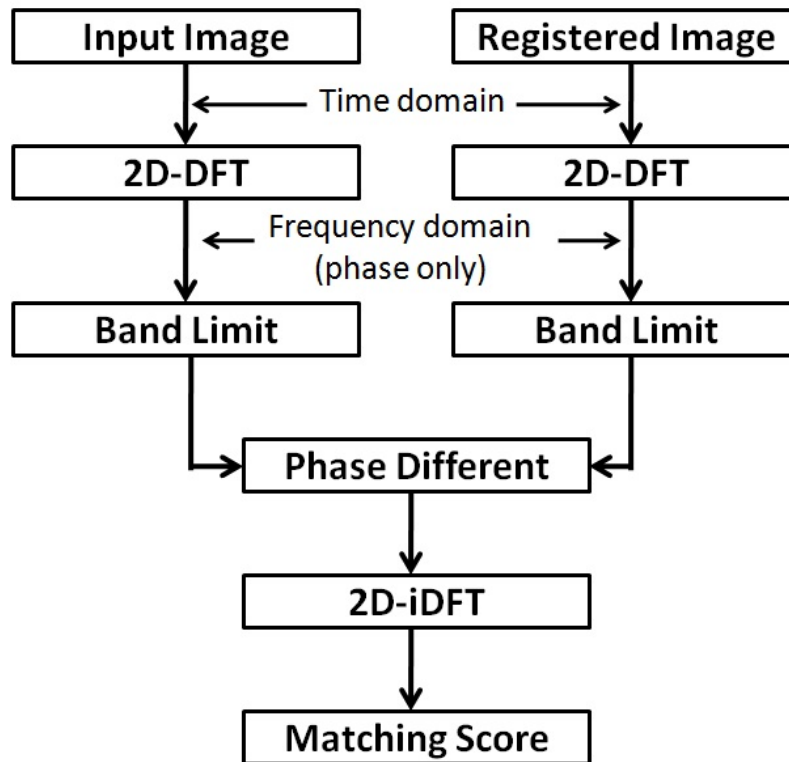


Figure 2.5: Overall flow to calculate the similarity of two images using BLPOC function

2.4 Previously Proposed Hardware Implementation of POC and BLPOC

Nowadays, with the growing of technology, human become more concerns about the speed, area and power. The software-based implementation of POC and BLPOC functions able to provide the function on identification and security system, but it might not be able to give a fast speed, small area and low power consumption. Due to these matching method involves 2D-DFT, which is a complex and high computation process, it results a slow speed for software-based implementation. Therefore, the hardware-based implementation of POC and BLPOC are proposed by Danese et al. (2009) and Danese et al. (2011), respectively, to improve the speed.

2.4.1 An FPGA-based Embedded System For Fingerprint Matching Using Phase-Only Correlation Algorithm

Danese et al. (2009) implemented fingerprint matching system on Altera Stratix II FPGA. Their proposed matching algorithm is based on POC function. For the Fourier Transform step in

POC function, they proposed to use two dimension Fast Fourier Transform (2D-FFT). Due to the Fourier Transform step is the most demanding step in the POC algorithm, they decided to implement an *ad hoc* hardware for it, and the steps other than 2D-FFT are implemented on Altera Nios II processor. Their proposed hardware block diagram of the 2D-FFT algorithm is shown in Figure 2.6.

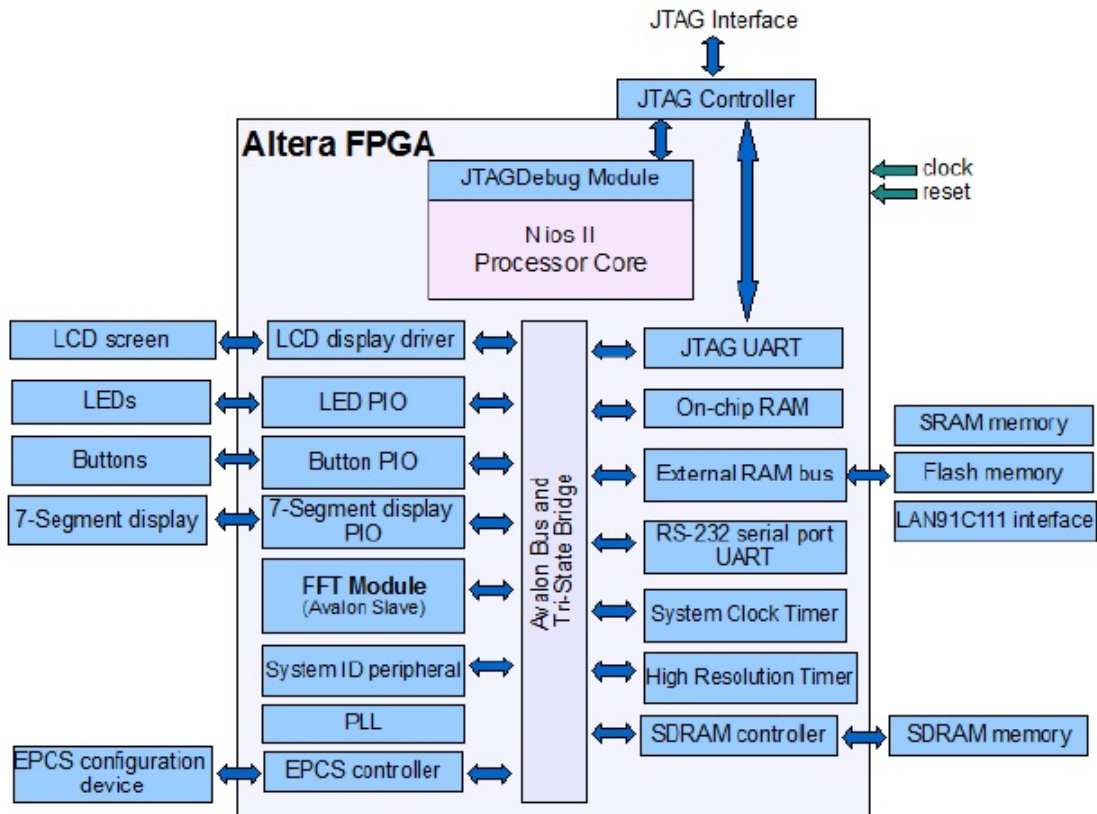


Figure 2.6: Proposed 2D-FFT hardware system block diagram by Danese et al. (2009)

In the performance evaluation, they compare the output of the proposed architecture with software-based implementation at the precision of two floating points. For the results, the errors of the phase components are usually less than 6%, with a maximum value of 11.5% and mean value of 0.72%. For the FPGA device, the resources used on logic elements is 32%, 9% of the available memory, 18% of the pins and 44% of the DSP blocks. In the performance of speed, they proved that the proposed hardware implementation had speedup at factor of 22, which is 12.86 μ s of the execution time for proposed hardware system and 292.20 μ s for the software-

based implementation, which is implemented on a computer with Pentium IV processor at 2.4 GHz.

2.4.2 An embedded multi-core biometric identification system

Danese et al. (2011) implemented an embedded multi-core fingerprint identification system based on the BLPOC function on Altera Stratix II FPGA. Their proposed BLPOC function for fingerprint identification system having steps as follows:

1. Image enhancement includes background elimination and contrast augmentation;
2. Image rotation in several angle to generate a number of fingerprints to compare the other with;
3. Transform the enhanced fingerprints using 2D-FFT.
4. Discard the high-frequency components, so to keep only the signal which compatible with the physiological characteristics of fingerprint;
5. Remove the magnitude components and keep only the phase components for matching;
6. Calculate the phase different between two fingerprints;
7. Back-transform the phase different values using 2D-iDFT;
8. Search the peak value from the back-transformed values;
9. Compare the peak value to the threshold and decide whether the two fingerprints belong to the same fingerprint.

In the proposed system, they implemented only the matching part, which are step 6, 7 and 8, in hardware, while the others are implemented on a PC Intel Core2Quad Q6600 (2.4 GHz).

The proposed matching architecture is shown in Figure 2.7, which includes the phase different module, 2D-iDFT module and peak value finder module.

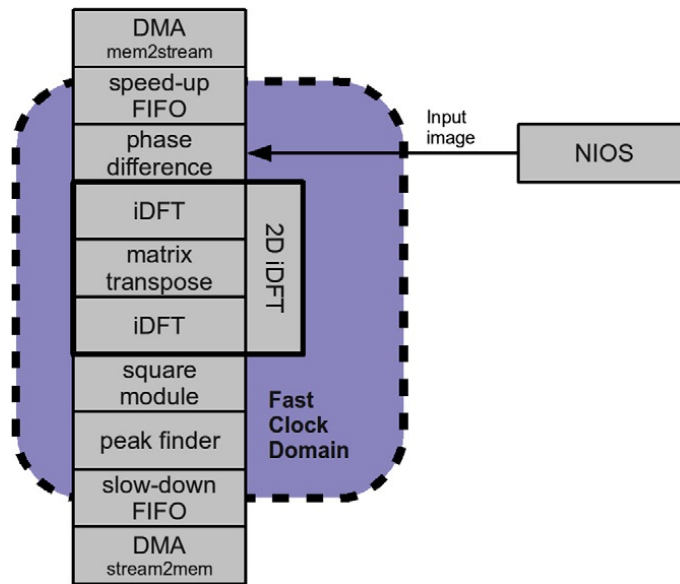


Figure 2.7: Proposed matching algorithm architecture by Danese et al. (2011)

The architecture used 86% of the total resources on the chosen FPGA board, which includes 59% of the total logic elements, 13% of total pins, 50% of total memory bits and 100% of the DSP blocks. In terms of speed, the proposed hardware implementation is proven to speed up at a factor of 20, which the execution time for hardware is $0.660\mu s$ while for software is 13ms. For performance evaluation, they developed a software-based implementation system, which is able to compare only the matching (hardware implementation) part. The Equal Error Rate (EER) of the proposed architecture is 6.16%.

According to Danese et al. (2009) and Danese et al. (2011), both had successfully implemented phase based matching method for biometric system. The speed of the hardware-based implementation is proven to significantly improve as compared with the software-based implementation. Danese et al. (2011) implement only the matching part of the algorithm on hardware, which the 2D-FFT is excluded in the hardware implementation. The architecture proposed by Danese et al. (2009) implemented the 2D-FFT on the FPGA board, while for the non-Fourier

Transform steps are implemented on Altera Nios II processor. Both of the proposed methods implement only part of the matching system in hardware, thus, the advantages of hardware implementation may not be optimized. The method of Fourier Transform used in both proposed architecture is 2D-FFT, which is suitable for a square size ($N \times N$) image. Since the sizes of finger vein images normally are not a square size, thus, the 2D-FFT algorithm is not suitable to be applied on finger vein recognition system.

Due to the 2D-FFT algorithm is not suitable to apply on finger vein recognition system, the 2D-DFT algorithm is chosen to be used in this research. The 2D-DFT algorithms are reviewed and compared in the following section.

2.5 2D-DFT Algorithm

Fourier Transform is one of the processes needed in BLPOC system, which function to convert data from the time domain to the frequency domain. This step is necessary in-order to get the phase components for BLPOC matching system. Due to the high computation calculation of 2D-DFT, it becomes the highest demand component in the BLPOC system. The using of 2D-FFT, as in paper Danese et al. (2011, 2009), able to perform faster speed compare with ordinary 2D-DFT method, but it limited to be only able to process image with square size ($N \times N$). The finger vein images are normally not be in a square size, thus, a flexible image size input of the Fourier Transform system is more suitable.

Among the existing hardware implementation of DFT, systolic architectures have been extensively popular due to the simplicity of the design having regular and local interconnection and rhythmic processing, and also the potential of using high level pipelining in small area and low power consumption (Meher (2005)). The traditional systolic design of 2D-DFT able to process flexible size of images. It consists of three stages of process, which are two stages of 1D-DFT

and one stage of transposition of the intermediate matrix. The stage of transposition the intermediate matrix is unattractive for hardware implementation, due to it involves global routing that causing the significant delay and consume larger area. Lim and Swartzlander (1999) and Meher (2005) proposed systolic array 2D-DFT architecture designs without involving the transposition operation.

2.5.1 Multidimensional Systolic Arrays for the Implementation of Discrete Fourier Transforms

Lim and Swartzlander (1999) proposed a multidimensional fully systolic array for implementing multidimensional DFT. They proposed to produce a fully systolic system by using a regular array of simple processors which is operated in two semi-systolic manners on alternating times. Equations 2.6 and 2.7 represent the equation of 2D-DFT, where P_0 is the input data to perform 2D-DFT, P_1 is the intermediate data and P_2 is the 2D-DFT output.

$$P_1(n_1, k_0) = \sum_{n_0=0}^{N-1} P_0(n_1, n_0) W_N^{k_0 n_0} \quad (2.6)$$

$$P_2(k_1, k_0) = \sum_{n_1=0}^{N-1} P_1(n_1, k_0) W_N^{k_1 n_1} \quad (2.7)$$

where $k_1, k_0 = 0, 1, \dots, N-1$, for input size $N \times N$.

The key idea of their proposed systolic array for the 2D-DFT is to combine the two types of semi-systolic arrays, which are shown in Figure 2.8(a) and (b), into one array. The H_{in} , H_{out} , V_{in} and V_{out} represent the horizontal input, the horizontal output, the vertical input and the vertical output respectively. The type 1 semi-systolic multiplication is used to compute Equation 2.6, while the type 2 semi-systolic multiplication is used to compute Equation 2.7. The input for type 1 array are the values from a two dimensional input (P_0), while the input for type 2 array are the

output of type 1 array (P_1). To avoid transposition step for the process of type 1 and type 2 arrays, a combination array of type 1 and type 2 is proposed as shown in Figure 2.9. Figure 2.9(a) and (b) shows the Processing Element (PE) and the combination for systolic array of 2D-DFT respectively. The number of PEs equal to the size of input data, where Figure 2.9(b) showing an example of input data size of 4×4 , therefore, it needs 16 PEs. For a 4×4 example as in Figure 2.9(b), the PE operated as type 1 in the first 4 cycles; for the next 4 cycles, the PE operated as type 2; then the process is repeated on the neighbor PEs. The D in the PE represent a delay, and the multiplexer switches every 4 cycles. This systolic array required $2N$ cycles to compute the first output of $N \times N$ 2D transform.

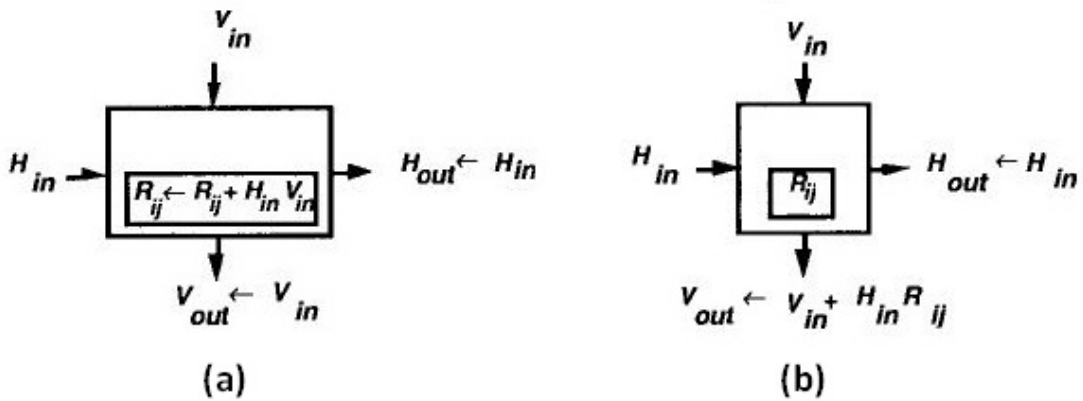


Figure 2.8: (a) Type 1 semi-systolic multiplication and (b) Type 2 semi-systolic multiplication by Lim and Swartzlander (1999)

2.5.2 Design of a Fully-Pipelined Systolic Array for Flexible Transposition-Free VLSI of 2-D DFT

Meher (2005) proposed the architecture of 2D-DFT in a fully-pipelined systolic array, without the transposition operation. The 2D-DFT is made from two dimension of 1D Fourier Transform, thus, the 2D-DFT equation as shown in Equation 2.1 can be decomposed into 2 stage of computation as shown in Equations 2.8 and 2.9. Considering an image $x(m,n)$ of size $M \times N$. As shown in Equation 2.8, the first stage of the 2D-DFT computes the 1D-DFT of each column of the

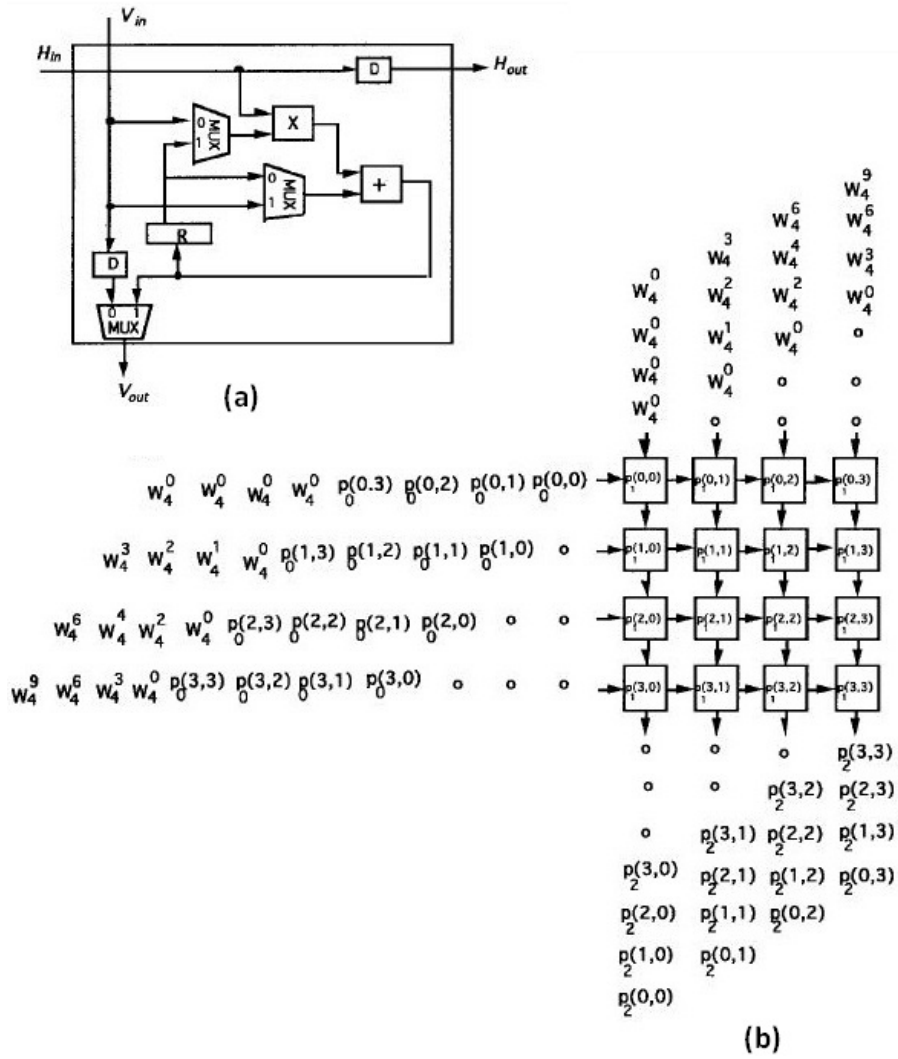


Figure 2.9: (a) PE structure and (b) Systolic array for 2D-DFT by Lim and Swartzlander (1999)

input image and the intermediate matrix, $V(k, n)$ is obtained. Then, as shown in Equation 2.9, the second stage of the 2D-DFT computes the 1D-DFT of each row of the intermediate matrix $V(k, n)$. The output for the second stage which is $X(k, l)$ is the 2D-DFT coefficients.

$$V(k, n) = \sum_{m=0}^{M-1} x(m, n) e^{-i \frac{2\pi}{M} (km)} \quad (2.8)$$

$$X(k, l) = \sum_{n=0}^{N-1} V(k, n) e^{-i \frac{2\pi}{N} (ln)} \quad (2.9)$$

The computation of Stage 1 as given in Equation 2.8 can be represented by a dependence

graph (DG1) as shown in Figure 2.10(a). The mathematical representation for each node in the DG1 is shown in Figure 2.10(b). For an input image $x(m,n)$ of size $M \times N$, the output of the graph is an intermediate matrix $V(k,n)$ of size $M \times N$. The first node from the first column of the graph processes the last pixel value from a row of the input image. Whereas, the first pixel value is processed by the last node of the first column. x_{in} and y_{in} are the inputs and x_{out} and y_{out} are the outputs for each node. As from Figure 2.10(a), the output x_{out} from a node will be the input x_{in} for the next horizontal node. While, the output y_{out} for a node will be the input y_{in} for the next vertical node. As for the calculation of y_{out} from the input y_{in} , the multiplying coefficient C is equal to $e^{-i \times 2\pi \times k/M}$ where k is $(k+1)^{th}$ column. Hence, the value of C will be the same for all nodes in a column. While, different columns will have a different value of C .

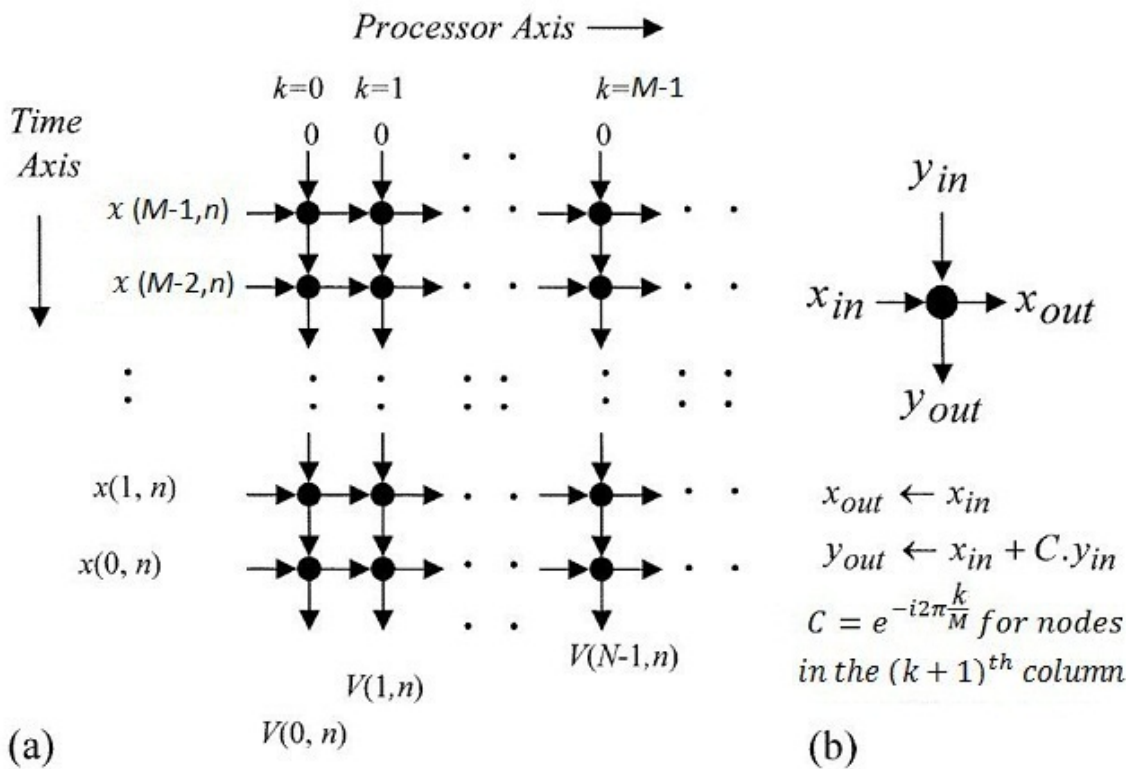


Figure 2.10: (a) Dependence graph (DG1) pertaining to computation of Stage 1 of recursions for n^{th} column of the input image, (b) Mathematical representation for each node by Meher (2005)

The computation of Stage 2 as given in Equation 2.9 can be represented by a dependence graph (DG2) as shown in Figure 2.11(a). The mathematical representation for each node in the