ABSTRACT

This paper presents two novel architectures for two-dimensional (2-D) Haar wavelet transform (HWT) of transform block in face recognition systems. The proposed architectures comprises 2-D HWT with transpose-based computation and dynamic partial reconfiguration (DPR) that have been synthesised using VHDL and implemented on Xilinx Virtex-5 FPGAs. To evaluate the proposed architecture, comparison for both configurations and a detailed performance analysis in terms of area, power consumption and maximum frequency are also addressed in this paper.

INTRODUCTION

In recent years, the demand for sophisticated security systems has risen significantly. Both commercial and governmental organisations require methods of protecting people and property. Varieties of biometric approaches have been investigated or adopted such as fingerprint [1], voice scans [2] and face recognition [3].

Face recognition has received a large amount of attention from researchers in recent years [3]. It has the potential to provide a robust biometric which, although unlikely to exceed the accuracy of techniques like iris or fingerprint scanning, could fulfill the needs of many scenarios.

Much of the interest in face recognition has been prompted by humans’ own remarkable ability to recognise faces [4]. This ability encompasses recognition of faces from thousands of known individuals, even in cases where there is partial occlusion of the face, poor illumination, or there has been a change in appearance.

The nature of face recognition systems applications involves performing complex task on a large set of database and often under real-time requirements [3]. Therefore, it is computationally intensive and an efficient hardware implementation appears as viable solution to be considered.

To accelerate the systems, this study deals with the hardware implementation of transform block in the face recognition system using field programmable gate array (FPGA). Xilinx FPGA device with dynamic partial reconfiguration (DPR) [5] technique has been selected to prototype the proposed architectures. With the ultimate goal to speed up the process of transforming input images into the wavelet coefficients, FPGA with the availability of advanced embedded resources such as soft cores, dedicated logic and block multipliers [6] is well suited.

The rest of the paper is organised as follows. An overview of the algorithm and methodology are presented in Section 2. Section 3 explains the hardware implementation: proposed systems applications and architecture. FPGA implementation results and an overview of the advantages offered with DPR technique are described in Section 4. Finally, concluding remarks are given in Section 5.

ALGORITHM AND METHODOLOGY

An overview of the algorithm and methodology are discussed in the following subsection.

2.1 Discrete Wavelet Transform (DWT)

Recently, hybrid multi-resolution approaches have received much attention. The discrete wavelet transform (DWT) [7] has been used along with a number of techniques, including principal component analysis (PCA) [4], independent component analysis (ICA) [8] and support vector machines (SVM) [9]. DWT is able to extract features that are localised in both space and frequency by convolving a bank of filters with an image at various locations.
However, to date, no systematic examination has been performed which determines how to best employ DWT for face recognition. The effect of employing different filters and scales has not been examined. A part of this research attempts to investigate these issues and has been published in [1].

The study is then widened to examine another interesting part on the hardware implementation. Among the filters that have been evaluated in [1], this paper highlights the contributions that can be achieved using Haar wavelet transform (HWT) as well as the advantages offered by DPR [10].

2.2 Haar Wavelet Transform (HWT) using Pipelined Direct Mapping

HWT is selected because of its simplistic nature, and mathematical features [11]. The mathematical features of the basis are as follows: the most simplistic wavelet basis, can be implemented using pairwise averaging and differencing, both unitary and orthogonal, and also it has compact support [11]. Calculation for both processes are described in Equations 1 and 2, where \( i = 0 \cdots (N/2 - 1) \).

\[
H_i = \frac{a_{2i} + a_{2i+1}}{2}
\]

\[
H_{(2i+1)} = a_{2i} + a_{2i+1}
\]

From implementation point of view, the one-dimensional (1-D) HWT flow diagram with \( N \)-inputs sample for pipelined direct mapping is shown in Figure 1, with ‘Avg.’ and ‘Diff.’ refer for average and differencing processes, respectively.

3 HARDWARE IMPLEMENTATION

An overview of the hardware implementations including the proposed system applications and architectures is explained in the following subsections.

3.1 Proposed System Applications

Figure 2 illustrates an overview of the proposed system for both the trained and after the training phase using the AT&T database. To accelerate the processes involved in face recognition system, two FPGA-based architectures of two-dimensional (2-D) HWT have been proposed to transform an image \( x^0 \) scale.

![Figure 2. Proposed system applications](image)

(a) Trained phase (b) After the training phase.

A high-level overview of the recognition approach adopted is given in Figure 3 (a), whilst the generic proposed 2-D HWT architecture is illustrated in Figure 3 (b). The whole chain to calculate the 2-D HWT gets an input as a 2-D image with \( N \times N \) point, and outputs the coefficients of the \( N \times N \) point.

To simplify the hardware design, the 2-D HWT is split into two 1-D HWT calculation cascaded together with transpose modules in between. This is achieved by performing the first 1-D HWT along the rows (columns) of the array followed by 1-D HWT along the columns (rows) of the transformed array. Transposition module stores the transposed coefficients into memory with a fetch unit module that reads back the coefficients for the next calculation.

3.2 Proposed Architectures

Both proposed architectures implementation on the FPGA are given in Figure 4 (a) and (b). The implementation of 2-D HWT without DPR defined the entire FPGA devices as one module. On the other hand, the implementation with DPR method and its framework consists of:

1. Two reconfigurable areas – for the 1-D HWT and transposition module; and
2. A static area – for the data fetch unit and the memory controller (Wishbone compliant).
3.3 2-D Haar Wavelet Transform (HWT) and Transpose-based Computation

The proposed 2-D HWT implementation works as follows. The input to the first 1-D HWT is read row by row, the 1-D HWT is performed on each input vector as they are provided and the calculated values are sent to the transpose module, which calculated the memory addresses for the transposition and stores the data into memory.

The transpose acts as a memory forwarder and performs matrix transpose, since row vectors are provided by the 1-D HWT. After transposition of the resultant matrix, another 1-D HWT is performed on the coefficients which are stored in memory to yield the two-dimensional 2-D HWT coefficients. Algorithm 1 gives the description of the 2-D HWT process.

Algorithm 1 The 2-D HWT pseudo-code

```plaintext
for row = 1 to norows do
    Apply a 1-D HWT column-wise
end for
for col = 1 to nocols do
    Apply a 1-D HWT row-wise
end for
```

3.4 2-D Haar Wavelet Transform (HWT) with Dynamic Partial Reconfiguration (DPR)

In this study, the ISE Design Suite 9.2PR and PlanAhead 10.1 [12] are used. With module-based DPR [10], this method has the limitation that all design files and reconfigurable modules must be available to the build environment to build partial modules.

Reconfigurable architectures using DPR technique comprises of several reconfigurable processing modules (RPM), a reconfigurable interface, an off-chip memory and micro blaze ($\mu$blaze). The system is connected to the host personal computer (PC) via peripheral component interconnect (PCI) express [10]. $\mu$blaze is a soft processor core designed for Xilinx FPGAs [12].

The reconfigurable processing modules allow hardware acceleration and can be reconfigured based on the system demand, whilst the communication interface is used to build the interconnection between RPM and the other components.

4 RESULTS AND ANALYSIS

FPGA implementation results for both architectures, analysis and an overview of the advantages offered with DPR technique are presented in the following subsections.
4.1 FPGA Implementation

In this study, Xilinx early access partial reconfiguration (EAPR) design flow [5] is used as a design flow reference, and these two architectures are implemented on the Xilinx Virtex-5 (XC5VLX110T-3FF1136).

In the face recognition system, the inputs are various size of images, hence different transform sizes ($N = 8, 16, 32, 64$ and $128$) have been used to examine the relationship of the transform sizes on the area (slices), power consumption (mW) and maximum speed (MHz).

In Table 6, results for both architectures are listed. As an example, for $N = 128$, the implementation with DPR technique yields a significant achievement with better resources used for area as well as better power consumption by 46.67% and 15.96%, respectively. On top of that, DPR technique also gives 4.59% better maximum frequency than without DPR.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Proposed 2-D HWT Without DPR</th>
<th>With DPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N = 8$</td>
<td>2,180</td>
<td>1,376</td>
</tr>
<tr>
<td>$N = 128$</td>
<td>38,261 (3.15%)</td>
<td>17,723</td>
</tr>
<tr>
<td>$N = 8$</td>
<td>20,403 (29.51%)</td>
<td>9,026.44</td>
</tr>
<tr>
<td>$N = 128$</td>
<td>36,723 (55.35%)</td>
<td>17,723</td>
</tr>
</tbody>
</table>

4.2 Discussions

To underline the influence of different transform size on area, power consumption and maximum frequency, Figures 5 – 7 illustrate the relationship for each performance indicator. Results obtained are clearly shown that the proposed 2-D HWT without DPR consumes more area and power. Using DPR technique, better area and power saving can be achieved between 36.68% to 46.67% and 6.78% to 15.96%, respectively. Additionally, to visualise the impact of non-partial and partial reconfiguration chip layouts for $N = 16$ and $64$ are given in Figure 8.

DPR is a promising technique for reducing the hardware required as well as improving the performance of the system. With this technique, the design can be divided into sub-designs that fit into the available hardware resources and can be uploaded into the reconfigurable hardware when needed [10].

In SRAM-based FPGAs, full-device reconfiguration is required upon power up [13]. The process of initialisation involves the FPGAs to be programmed with a configuration bitstream file. Partial reconfiguration concept appears after initialisation and works to modify a fraction of the resources by programming the FPGA with a partial bitstream file. Obviously, a full bitstream size is very massive whereas a partial bitstream may represents only 2% of the full bitstream [6], [13]. With smaller bitstreams, several advantages can be achieved: reduced reconfiguration time, reduced storage requirements, and dynamic allocation of functionality.

The application of face recognition requires several building blocks for its computationally intensive processes to perform matrix transformation operations. Moreover, complexity in addressing and accessing large databases have resulted in vast challenges from a hardware implementation point of view. To cope with these issues, an FPGA-based architecture with efficient reconfigurability techniques is a promising solution to meet the demands of these applications in terms of speed, size (area), power consumption and throughput.
Figure 8. Comparison of chip layout for different transform sizes on XC5VLX110T-3FF113.

5 CONCLUSIONS

This paper presents two architectures for 2-D HWT have been proposed for the transform in the proposed face recognition system based on transpose computation and partial reconfiguration.

To sum up, comparative study for both non-partial and partial reconfiguration processes has shown that DPR offers many advantages and lead to a promising solution for implementing computationally intensive applications such as face recognition systems. Using DPR, several large systems are mapped to small hardware resources and the area, power and maximum frequency are optimised and improved.

REFERENCES