Journal of Information Technology and Engineering
Vol. 3 No. 2 (December, 2018)

An FPGA Prototyping of Gabor-wavelets Transform for Motion Detection

Bing XIONG, Ce ZHU, Charayaphan CHAROANSAK and Wei YU
Nanyang Technological University, School of EEE, Nanyang Avenue, Singapore 639798
E-mail: xiongbing@pmail.ntu.edu.sg, eczhu@ntu.edu.sg, ecchara@ntu.edu.sg,yuwei@pmail.ntu.edu.sg

Received: 03rd June 2017 Revised: 14th August 2017 Accepted: 01st March 2018

Abstract: *Many motion detection algorithms have been developed for various applications such as machine vision, face recognition and content based video indexing and retrieval techniques. In this paper, motion estimation based on Gabor-wavelets transform is reviewed, which can be used to detect the motion information in image sequences accurately. In order to overcome the high computation load involved in Gabor transform for different orientations and scales, a parallel structure is developed to speed up the calculation. System level approach to the design of FPGA (Field Programmable Gate Array) for the computation of Gabor wavelets transform is described.*

Key words: *Gabor-wavelets, Motion detection, FPGA.*

1. INTRODUCTION

Motion detection and estimation in image sequences is very useful in many machine vision applications including automatic surveillance, traffic monitoring and object segmentation based on movement [1, 3, 6, 11]. Accordingly, a large number of motion detection and estimation algorithms have been developed [2, 4, 5].These algorithms can be divided into two broad classes, filter-based method and matching-based method.

A typical example of a filter-based method is the gradient method, where a motion constraint analysis is used to estimate the image flow. The well-known optical flow constraint (OFC) equation is [2]:

$$\frac{\partial I}{\partial x}\frac{\partial x}{\partial t}+\frac{\partial I}{\partial y}\frac{\partial y}{\partial t}+\frac{\partial I}{\partial t}=0 \qquad (1)$$

where I(x, y, t) is the pixel located at x = (x ,y) and at time t in the image sequences. This method estimates motion locally in a small region of image to achieve low computation, with the risk of suffering from the aperture problem. It provides the velocity in the direction of the gradient only. To obtain full velocity fields, larger space-time support is required.

The matching technique, on the other hand, suffers from the correspondence problem, i.e. the ambiguity as to which feature point in one frame is to be matched with the one from the previous frame.

The motion estimation method studied here is based on complex Gabor wavelets transform of image sequences. Gabor wavelets have a large support and are sensitive to texture components of different orientations [1, 5, 7, 8], and can overcome the aperture problem. However, with many different scales and orientations, the computation load of the Gabor wavelets transform is very high. In this paper, we develop a parallel structure to accelerate the computation of the Gabor transform. A system level design of FPGA (Field Programmable Gate Array) for the computation of Gabor wavelets transform is described. Implementing the algorithm in FPGA allows optimal parallelism, which is needed to handle the high computation load. Although FPGA does not offer an optimized hardware implementation when compared to ASIC (Application Specific Integrated Circuit), it allows short development time and enables verification of algorithms in hardware at low cost.

The paper is organized as follows. The following section reviews the motion estimation method based on Gabor-wavelets transform which is introduced in [1]. Section 3 develops a parallel structure to accelerate the computation of the Gabor-wavelets transform and describes its fast prototyping on FPGA. Section 4 demonstrates the motion detection results and conclusions are given in section 5.

2. GABOR-WAVELETS TRANSFORM FOR MOTION DETECTION

The system studied here is a phase-based motion detection, or disparity calculation, which has attracted attention recently due to its robustness. The main concept of this approach is based on the Fourier shifting property. For a given signal, $s(\boldsymbol{x})$, and its Fourier transform, $S(\boldsymbol{k})$, the following relation holds:

$$S(x+d)\mapsto S(k)e^{ikd} \qquad (2)$$

The phase shift in the spectrum, kd, can be used to calculate the spatial translation, d. However, Fourier transform operation is global on the entire signal, thus it can detect only the uniform global translation. Generally in a video sequences, only some parts of the image are moving while the rest is not. Therefore, local detection of motion is more practical.

The concept of local motion detection introduced here is based on the phase information in a local representation of the image sequence that can be produced by a family of Gabor wavelets. The complex Gabor filter implemented on images with spatial translations leads to local phase shifts that can be used to compute the displacement of each image point. The algorithm for image flow estimation based on Gabor wavelets transform will be explained in more details in the following.

2.1 Gabor-wavelets Transform

The first step is to compute the image flow field. The method used here is based on a convolution with Gabor wavelets. Gabor wavelets have the shape of localized plane waves, bounded by a Gaussian envelope function [5, 8, 9]. It is used in many early vision related tasks such as texture discrimination and feature extraction. A family of Gabor wavelet kernels are defined as:

$$\Psi_j(x) = \frac{k_j^2}{\sigma^2}\exp\left(-\frac{k_j^2 x^2}{2\sigma^2}\right)\exp\left(ik_j \bullet x\right) \qquad (3)$$

wh ere $\boldsymbol{k}_j$ is the wave vector of the kernel's main frequency and $\bullet$ indicates dot product of two vectors, and $\boldsymbol{k}_j$ and $\boldsymbol{x}$ indicate the norm of vectors $\boldsymbol{k}_j$ and $\boldsymbol{x}$ respectively. Hence, the Gabor wavelets are in the shape of plane waves, whose magnitudes are modulated by a Gaussian envelope function, changing with wave vector $\boldsymbol{k}_j$ defined as:

$$\boldsymbol{k}_j = \begin{pmatrix} k_{jx} \\ k_{jy} \end{pmatrix} = \begin{pmatrix} k_v \cos\varphi_\mu \\ k_v \sin\varphi_\mu \end{pmatrix}, \quad k_v = 2^{-\frac{v+2}{2}\pi}, \; \varphi_\mu = \mu\frac{\pi}{8} \qquad (4)$$

where $j = \mu + v$. In this paper, a discrete set of 5 different frequencies, v = 0 to 4, and 8 orientations, μ = 0 to 7, are implemented. The width of the Gaussian envelope is chosen as $\delta = 2\pi$.

The Gabor wavelets transformation of an input image I(x) is defined as a convolution:

$$J_j(x) = \int I(x')\Psi_j\left(x - x'\right)dx' \qquad (5)$$

Gabor wavelets transform of a sample image is shown in Fig. 1.

After the transformation using Gabor wavelet with different orientation and scales, each image point corresponds to 40 complex coefficients, which is defined as a jet $\boldsymbol{J}$. They can be written as:

$$J_j(x) = a_j(x)\exp\left(i\phi_j(x)\right) \qquad (6)$$

with amplitudes $a_j(x)$, which varies slowly with position, and phases, $\phi_j(x)$. The phase $\phi_j(x)$ rotates with a rate set by the spatial frequency or wave vector of the kernels.

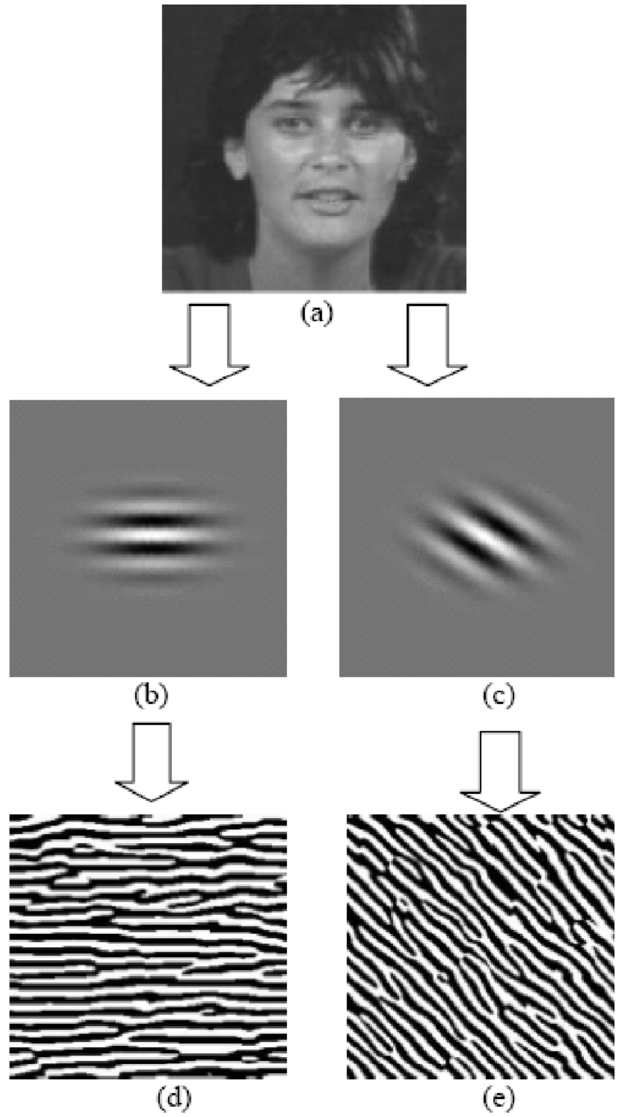

Figure 1: Gabor Wavelets Transform of an Image: (a) the Original Image, (b) and (c) the Gabor Wavelets with Orientation of 0 and 45 Degree Respectively, (d) and (e) the Corresponding Real Parts of the Transform.

2.2 Displacement Estimation

In order to estimate the flow vectors from successive frames of a video sequence, we consider the method used for disparity estimation in [1]. The filter output $\boldsymbol{J}$ is most sensitive to translations of $\boldsymbol{I}$(x) in directions ϕ_u and least sensitive in direction $\phi_u \pm \pi/2$. If one region in the sequential frame is a version of the previous frame, ($I'(\boldsymbol{x}) = I(\boldsymbol{x}+\boldsymbol{d})$), a phase shift of the complex filter output results by the convolution theory:

$$\Psi_j(x) * I(x+d) = F^{-1}\left\{G(w)I(w)e^{ikd}\right\} \qquad (7)$$

where * denotes the convolution operation and $F^{-1}(\cdot)$ is the inverse Fourier transform. A space shift $\boldsymbol{d}$ in the successive frames can be recognized as a space phase shift of the Gabor transform since each single Gabor transform is only affected by the spatial support of the kernel. Therefore, proportion between the local phase shift ***kd*** and the disparity ***d*** is established. The computation of this disparity between pairs of successive images is similar to the task of estimating the small positional displacements in a matching procedure. The idea here is to maximize the similarity function with respect to ***d***:

$$S_\phi = \frac{\sum_j a_j a_j' \cos\left(\Delta\phi_j - \boldsymbol{d}\cdot\boldsymbol{k}_j\right)}{\sqrt{\sum_j a_j^2 \sum_j a_j'^2}} \tag{8}$$

By setting $\frac{\partial}{\partial d_x} = \frac{\partial}{\partial d_y} = 0$ in its Taylor expansion, we have

$$d(x) = \begin{pmatrix} d_x \\ d_y \end{pmatrix} = \frac{1}{\Gamma_{xx}\Gamma_{yy} - \Gamma_{xy}\Gamma_{yx}} \times \begin{pmatrix} \Gamma_{yy} & -\Gamma_{yx} \\ -\Gamma_{xy} & \Gamma_{xx} \end{pmatrix} \begin{pmatrix} \phi_x \\ \phi_y \end{pmatrix} \tag{9}$$

where $\phi_x = \sum_j a_j a_j' k_{jx} \Delta\phi_j$, $\Gamma_{xy} = \sum_j a_j a_j' k_{jx} k_{jy}$ and ϕ_y, Γ_{xx}, Γ_{yy}, Γ_{yx} can be defined similarly. Using this equation, the displacement vectors can be estimated from two jets taken from the same pixel position in two successive frames. The motion vectors estimated here are used for motion detection.

3. FPGA PROTOTYPING FOR GABOR WAVELETS TRANSFORM

3.1 Parallel Gabor Wavelets Transforms

The drawback of the above motion estimation method is that, with many different scales and orientations, the computation load of the Gabor wavelets transform is very high. Hence, acceleration of the transform process is necessary to achieve a real-time motion estimation system using Gabor wavelet transform. It is noticed from the analysis of the Gabor wavelet transform that the transform for different orientations and scales are independent of each other. Therefore, a hardware implementation structure can be developed to accelerate the calculation of the whole transform by parallelizing the transforms for different orientations and scales. The block diagram of our parallel hardware structure for the computation of Gabor wavelets is shown in Fig. 2. The Gabor filter masks at different orientations and scales are stored in ROM. The input image is acquired (from a prerecorded video or real-time camera) and passed through the parallel system of convolution masks to produce all the wavelets transforms. Thus, the computation of Gabor-wavelets is greatly accelerated.

A similar system has been proposed in [12] but the architecture is based on using multiple DSPs. We propose the implementation using FPGA as will be described in the next sub-section. Using an advanced system level design

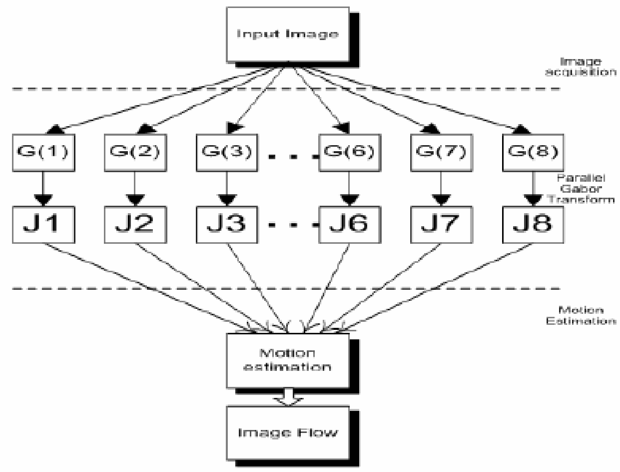

Figure 2: Top-level Block Diagram of the FPGA Hardware for the Computation of Gabor Wavelets Transform

called System Generator from Xilinx, the process of FPGA design becomes much easier.

3.2 Hardware Architecture for Gabor Wavelets Motion Detection

As can be seen in the block diagram in Fig. 2, the parallel FPGA architecture is made up of three main parts, image acquisition, parallel Gabor transform, and motion estimation. In our experimentation, the video sequence is stored as a sequence of bitmap images in computer hard disk. The size of each gray scale image is 256×256 and each pixel is represented by 8 bits. For FPGA simulation, each image file is read into MATLAB workspace, raster scanned, and the image data is fed into the FPGA circuit one pixel at a time.

In our simulation, the FPGA design tools used were Xilinx™ System Generator version 3.1 [10], Simulink™, and MATLABTM version 6.5. The FPGA synthesis tool used was Xilinx™ ISE 5.2i. System Generator from Xilinx provides a bit-true and cycle-true simulation of the Simulink™ models under MATLAB environment. Note that the FPGA design using System Generator is different from the more typical approach of using HDL (Hardware Description Language) or schematics. Using System Generator, the FPGA is designed by means of Simulink models. Thus, the FPGA functional simulation can be carried out easily right inside Simulink environment. After the successful simulation, the synthesizable VHDL (VHSIC HDL where VHSIC is Very High Speed Integrated Circuit) code is automatically generated from the models. As a result, one can define an abstract representation of a system-level design and easily transform it into a gate-level representation in FPGA.

Using System Generator for FPGA implementation might not give the optimized performance/gate count result as compared to more conventional approaches such as using hardware description language or schematic entry. However, this system level approach allows simple design flow, verification, and implementation of the algorithms.

The top-level design of Gabor wavelets transform using System Generator under MATLAB Simulink is shown in Fig. 3. The design is made up of many sub-systems. More detailed information for one sub-system that performs the convolution is shown in Fig. 4.

Table 1
Gate Requirement of the FPGA Design for Gabor Wavelets Motion Detection

Number of Slice for Logic	650
Number of Slice for Flip Flops	305
Number of 4 input LUTs	3002
-used as LUTs	2030
-used as a route-thru	450
-used as Shift registers	522
Total equivalent gate count	100213

Table 2
Maximum Combinational Path Delay and Operating Frequency of the FPGA Design

Maximum path delay from/to any node	15.8 nsec
Maximum operating frequency	71.2 MHz

Table 1 details the gate requirement of the FPGA design. The total gate requirement reported by the ISE is approximately 100 Kgates. The price of the device in this gate size range (for example 300 Kgates) is very low which confirms the advantage of using FPGA. Table 2 shows the reported maximum path delay and the highest FPGA clock frequency of 71.2 MHz.

After successful simulation, the VHDL codes were automatically generated from the design using System Generator special block set. The VHDL codes were then used to synthesize the FPGA using Xilinx ISE 5.2i development tool. Xilinx Virtex-E family of FPGA was chosen for synthesis with optimization set for speed. The testing of the FPGA was done using a prototype board equipped with a 600,000 gates Virtex-E FPGA. A digital input/output interfacing adapter was used for transferring data between the computer and the prototype board.

In this experiment, no analog-to-digital conversion was implemented. MATLAB SimulinkTM program was used to control the operation of the FPGA and to read the data stream of the Gabor wavelets transform from the prototype board using digital input/output adapter. The rate of the data stream is thus less than real-time operating frequency. This means that the testing was not in real-time but with real functioning of the FPGA. The estimated achievable maximum processing speed of 20 frames/second is reported by ISE 5.2i after the FPGA routing. The motion detection results achieved are shown in next section.

4. MOTION DETECTION RESULTS

Fig. 5 illustrates some of the simulation results. The first two columns are two consecutive images in tested image sequences. The third column is the motion detection result. The brightness of the motion detection results represents the magnitude of the estimated motion vector, that is, the brighter pixel in the result image denotes large movement. For ease of evaluation, it is normalized to 0 to 255 for display.

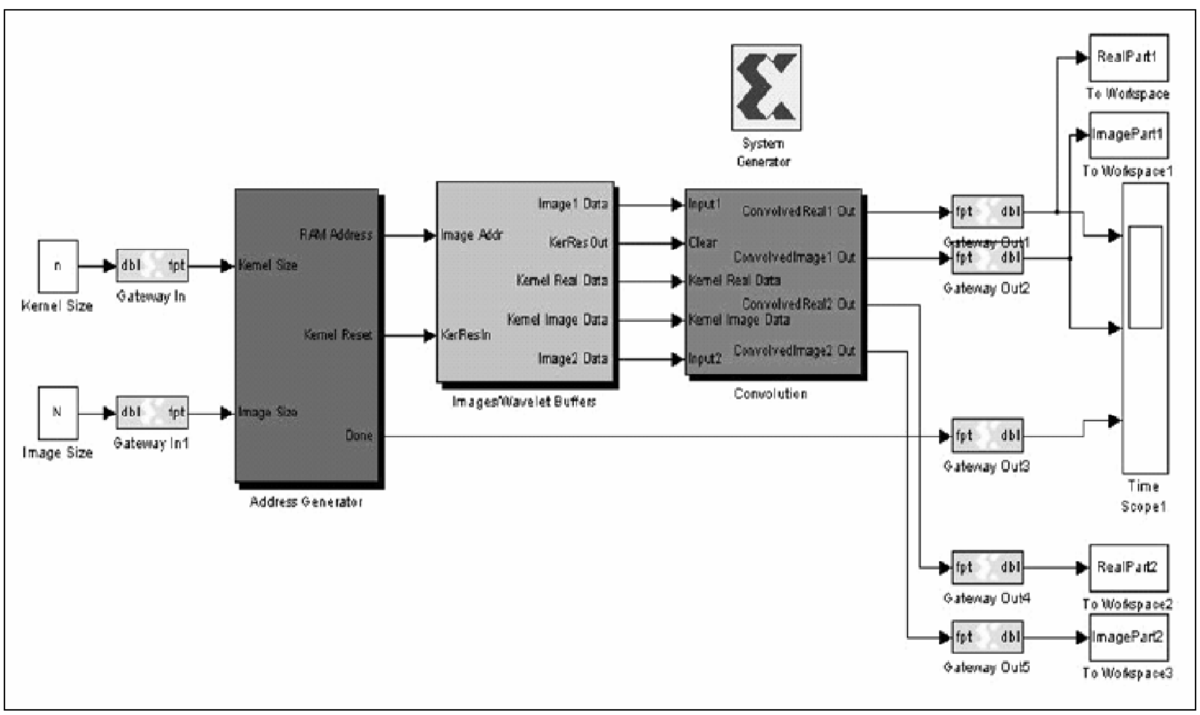

Figure 3: Top-level Design of Gabor Wavelets Transform using System Generator.

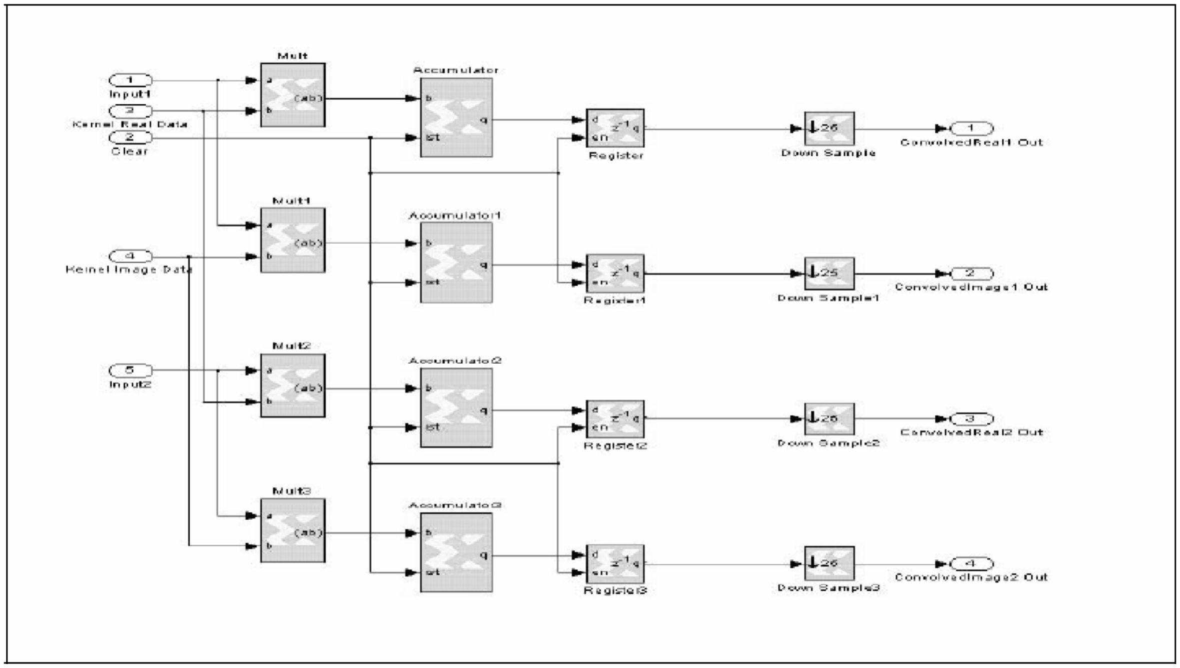

Figure 4: Detail Design of Convolution Sub-system.

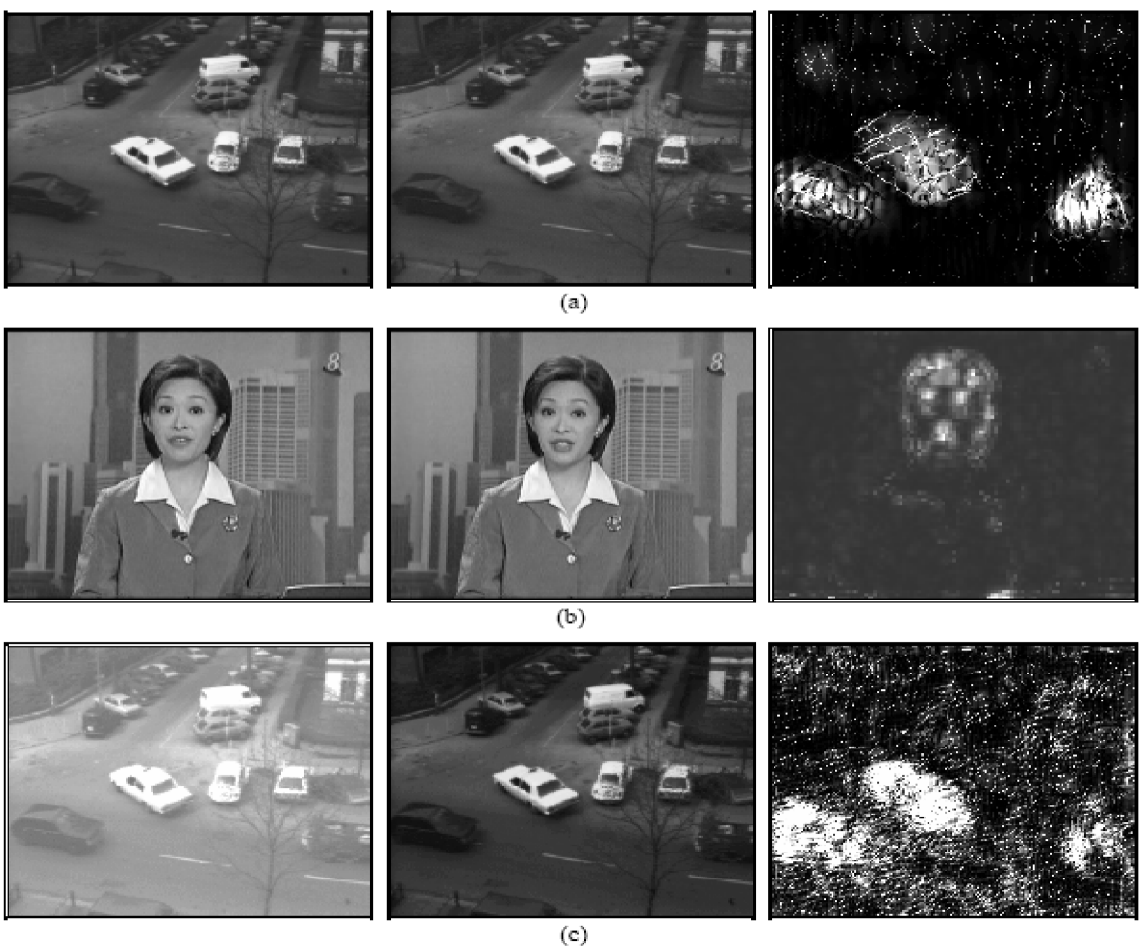

Figure 5: The Implemented Motion Estimation Result for three Image Sequences. (a) Hamburg Taxi Sequence, (b) Newswoman Sequence, (c) Hamburg Taxi with Contrast Change.

Fig. 5(a) shows the motion detection results of two frames in a video sequence "Hamburg Taxi" which is commonly used for evaluating motion detection algorithms. There are four moving objects, the turning taxi, a black car in the left, a truck in the right and a pedestrian in upper-left corner. From the estimated movement in the sequence shown in the third column, it can be seen that the three large moving vehicles are detected while the pedestrian's motion is ambiguous due to the relatively small size and the obscure motion.

The newswoman sequence, shown in Fig. 5(b), is captured from TV news broadcasting. In this video sequence, the newswoman is reporting while moving her head with small motion in front of the camera. In the motion estimation results, the unconstrained problem is very obvious. In the moving regions without distinct texture, such as the cheek, only the moving edges are prominently detected.

Fig. 5(c) illustrates the robustness of the implemented motion estimation method. In this test, the brightness of the first image has been increased. The motion estimation result in the third column shows that the motion of vehicles is still detected, which demonstrates that the method is robust against brightness changes.

5. CONCLUSIONS

In this paper, we have presented a parallel scheme to compute the Gabor-wavelets transform with different scales and orientations. It has been used in a phase-based motion estimation for successive frames from a video sequence. The system level FPGA design for parallel computation of the Gabor transform has been presented.

The design requires only a small size FPGA (100 K gates) and so is low in device cost. This FPGA implementation scheme may find applications involving Gabor-wavelets for real-time face recognition and object tracking.

REFERENCES

[1] L. Wiskott, "Segmentation from Motion: Combining Gabor-and Mallat-wavelets to Over-come the Aperture and Correspondence Problems", *Pattern Recognition*, Vol. 32, pp. 1751-1766, 1999.

[2] B. K. P. Horn, B. G. Schunck, "Determining optical flow", *Artificial Intel*. Vol. 17, pp. 185-203, 1981.

[3] W. M. Theimer, H. A. Mallot, "Phase-based Bino-cular Vergence Control and Depth Reconstruction Using Active Vision", *CVGIP: Image Understanding*, Vol. 60, No. 3, November, pp. 343-358, 1994.

[4] D. J. Fleet, A. D. Jepson, "Computation of Component Image Velocity from Local Phase Infor-mation", *Inter. J. of Computer Vision*, Vol. 5, No. 1, pp. 77-104, 1990.

[5] E. Bruno, D. Pellerin, "Robust Motion Estimation using Spatial Gabor-like Filters", *Signal Processing*, Vol. 82, No. 2, February 2002, pp. 297-309.

[6] L. Wiskott, J.M. Fellous, N. Kruger, C.von der Malsburg, "Face Recognition by Elastic Bunch Graph Matching", *IEEE Trans.on Pattern Anal. Mach. Intel*. Vol. 19, No. 7, pp. 775-779, 1997.

[7] J. G. Daugman, "Complete Discrete 2-D Gabor Transformas by Neural Networks for Image Analysis and compression", *IEEE Trans. on Acoustics, Speech, and Signal Processing*, Vol. 36, No. 7, pp. 1169-1179, July, 1988.

[8] T. P. Weldon, W. E. Hinggins, D. F. Dunn, "Efficient Gabor Filter Design For Texture Segmentation", *Pattern Recognition*, Vol. 29, No. 12, pp. 2005-2015, Dec., 1996.

[9] D. Dunn, W. Higgins, J. Wakeley, "Texture Segmen-tation using 2-D Gabor Elementary Functions", *IEEE Trans. on Pattern Anal. Machine Intell*. Vol. 16, No. 2, pp. 130- 149, February, 1994.

[10] Xilinx Inc., "Xilinx System Generator Vol. 3.1 for The MathWorks Simulink: Quick Start Guide", 2003.

[11] J. Badenas, J. M. Sanchiz, F. Pla, "Motion-based Segmentation and Region Tracking in Image Sequences", *Pattern Recognition*, Vol. 34, pp. 661-670, 2001.

[12] F. Yang and M. Paindavoine, "Fast Motion Estimation Based on Spatio-temporal Gabor Fitlers: Parallel Implementation on multi-DSP, Advanced Signal Processing Algorithms, Architecture, and Implementations", *Proceedings of SPIE*, Vol. 4116, pp. 454-462, 2000.