International Journal of Soft Computing and Engineering (IJSCE)
ISSN: 2231-2307, Volume-2, Issue-1, March 2012

FPGA Based High Performance Optical Flow Computation Using Parallel Architecture

N.Devi, V.Nagarajan

Abstract-The proposed work describes a highly parallel architecture for high performance optical flow computation. This system implements the efficient Lucas and Kanade algorithm with multi-scale extension for the computation of large motion estimations. This work deals with the architecture, evaluation of the accuracy and system performance. It also has extension to the original L&K algorithm. So the capable of working is larger than the standard mono scale approaches. In this proposed system, Matlab and Modelsim simulation are selected for local optical flow algorithms due to their potential for a high-performance massive parallelization. The results are obtained with a throughput of one pixel per clock cycle along the whole processing scheme by using the fine-pipeline based architecture.

Key words: FPGA, Lukas kanade algorithm, Pipelining

I. INTRODUCTION

In this technology world digital image processing is used for various applications such as video surveillance, target recognition [1], image enhancement and optical flow estimation with the help of image processing algorithm. This scheme is used in medicine, space exploration, surveillance, authentication, automated industry inspection and robot navigation fields.

To speed up the processing scheme, the digital image processing techniques goes for the hardware implementation. Three types of hardware implementation are possible. They are ASIC, DSP and FPGA. FPGA is the reconfigurable device. It only allows the pipeline architecture and parallelism (DSP won't allow this type of architecture). It has low computational time and low cost. This type of chips simplifies the debugging and verification of complex algorithm.

Motion is an important source of information about the world. It used in various field such as segmentation, surface structure from parallax, self-motion, recognition, understanding behavior, understanding scene dynamics.

Optical flow has two properties. One is determining what image property to track and the second is how to track it. In research field brightness is often assumed as constant but it is often violated by Mother Nature. So the resulting optical flow in 2D motion field sometimes shows a very poor approximation. For example, Fig.1. shows a rotating Lambertian sphere with a static light source produces a static image but a stationary sphere with a moving light source produces drifting intensities.

Manuscript Received January 09, 2012.

N.Devi, Department of Electronics and Communication Engineering, Anna university of Technology. Adhiparasakthi Engineering College, Melmaruvathur, Chennai, India (**tvndevi@gmail.com**)

*__Dr.V.Nagarajan__, Department of Electronics and Communication Engineering, Anna university of Technology. Adhiparasakthi Engineering College, Melmaruvathur, Chennai, India **(nagarajanece31@rediffmail.com)**

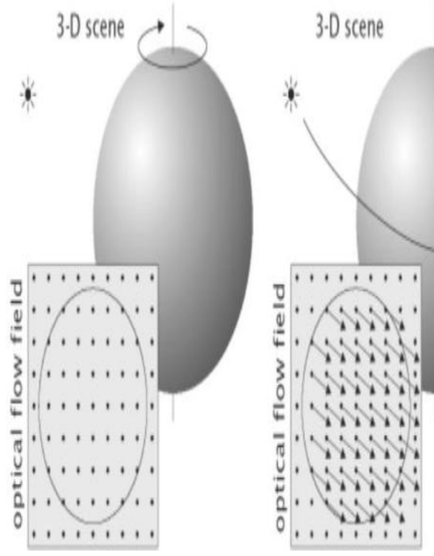

Fig.1. Optical Motion for Static and Rotating Lambertian Sphere

This project describes the high performance Optical flow computation technique which converts the video input into sequence of frames. These frames are again converted into text using Matlab. Here the simulation is done with the help of Modelsim. Optical flow estimation is done with the help of Lucas-Kanade algorithm and parallel architecture.

Some of the drawbacks in existing system are limited by speed or hardware resources, need of larger area, the results are not so accurate when using hardware, external memory also a limiting factor in hardware. To overcome these drawbacks the new approach with L&K algorithm is described here. It is used to find the high performance optical flow computation with high speed, high resource utilization and accurate result.

The objective of this proposed system is to achieve a throughput of one pixel per clock cycle and implementation of high-performance parallelization system using FPGA with high resources utilization.

II. OPTICAL FLOW ESTIMATION

Optical flow or optic flow is the pattern of apparent motion of objects, surfaces, and edges in a visual scene. It is caused by the relative motion between an observer (an eye or a camera) and the scene. Fig.2. have the 3D and 2D view of optical flow.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

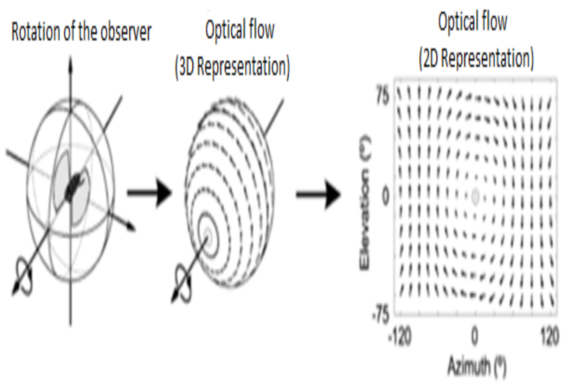

Fig.2. Optical Flow

The motion estimation of sequences of ordered images allow as either instantaneous image velocities or discrete image displacements. The optical flow methods try to calculate the motion between two image frames [2]-[3] which are taken at times t and $t + \delta t$ at every voxel position.

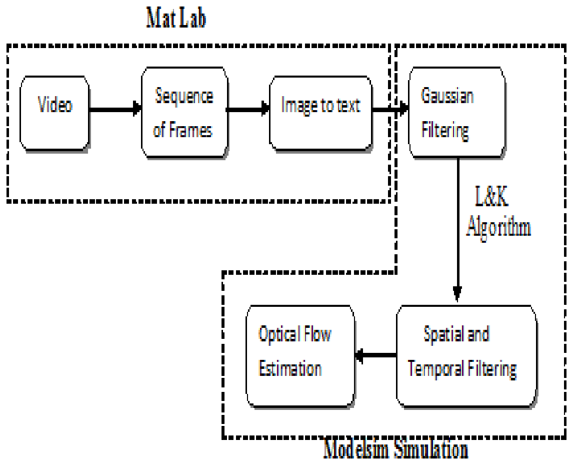

Fig.3. Processing Structure

Fig.3. describes the whole processing structure of this project. Different methods [4]-[10] are used to find the optical flow estimation but L&K algorithm is mainly used to give the partial derivatives of the pixels. Computation of the optical flow estimation is assuming the brightness constancy and find velocity vector (u,v). One of the most accurate, computationally efficient, and widely used methods for the optical flow computation is L&K approach. It is a local method that belongs to the gradient based calculations.

The accuracy for estimation of large displacement is the main disadvantage of L&K algorithm. So the high performance approach is used. It includes multi-scale implementation with this algorithm that computes the optical flow velocity components for each different resolution.

This project consists of coarse to fine multi-scale implementation.This architecture has several steps. They are

Stage 1: Depends on the size of the image frames pyramid computation of the input frames takes place.

Stage 2: Motion estimation is done with the help of L&K algorithm.

Stage 3: Velocity estimation is done with the help of velocity vectors.

Stage 4: Warping process is performed. The image frame to frame warping is done by using 2-by-2 bilinear interpolation &produce smoothed images.The term "image warping" means warp the input frame to already estimated position. Local motion should be done in filter tuning range.

Stage 5: Merging is done to find the optical flow. Warped frames are used for optical flow estimation & find new velocity (u,v).

Stage 6: 3-by-3 median filtering is done. It removes the non-confident values and act as bi-dimensional filter.

A. Optical Flow

Based on the Taylor series approximation of the image signal and using partial derivatives with respect to the spatial and temporal coordinates optical flow is estimated. For a 2D+t dimensional case (3D or n-D cases are similar) a voxel at location (x,y,t) with intensity $I(x,y,t)$ will have moved by δx, δy and δt between the two image frames, [3] and the following image constraint equation can be given as,

$$I(x,y,t) = I(x + \delta x, y + \delta y, t + \delta t) \qquad (1)$$

Assuming the movement to be small, the image constraint at $I(x,y,t)$ with Taylor series can be developed to get:

$$\frac{\partial I}{\partial x} v_x + \frac{\partial I}{\partial y} v_y + \frac{\partial I}{\partial t} = 0 \qquad (2)$$

Where V_x, V_y are the velocity of x and y components or optical flow of $I(x,y,t)$ and $\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}$ and $\frac{\partial I}{\partial t}$ are the derivatives of the image frames at (x,y,t) in the corresponding directions.

$$I_x V_x + I_y V_y = -I_t \text{ or } \nabla I^T . \vec{V} = -I_t \qquad (3)$$

This is an equation of two unknowns and cannot be solved as such. This problem is said as aperture problem of the optical flow algorithms. To find the accurate optical flow another set of equations is needed, given by some additional constraint. For estimating the actual flow all optical flow methods introduce additional conditions.

B. Gaussian Pyramid Generation

The first task is to find a representation which in effect decorrelates the image pixels to design an efficient compression code and represent the image directly in terms of the pixel values. An image pyramid is the representation of an image at different resolutions. The image pyramids are mainly used to generate a number of homogeneous parameters that represent the response of a bank of filters at different scales and possibly different orientations.

C. Lucas-Kanade Algorithm

The main concern of optical flow estimation is the pixel displacements between successive frames [11]. Here each pixel have the velocity vector (u,v) and assumed that the intensity of the pixel in the(x,y) position in the image at time t , and the pixel(x+u,y+v) in the image

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

of time(t+1) does not change. This only have good accuracy for slow-moving objects between frames (it depends on the distance from the object to the camera, on the 3-D object velocity, and on the camera frame-rate)

By assumption in Lucas-Kanade method the displacement of the image contents between two nearby instants (frames) is small and approximately constant within a neighborhood of the point *p* under consideration. Thus the optical flow equation can be assumed to hold for all pixels within a window centered at *p*. Namely, the local image flow (velocity) vector (V_x,V_y)must satisfy

$$I_x(q_1)V_x + I_y(q_1)V_y = -I_t(q_1) \quad (4)$$
$$I_x(q_2)V_x + I_y(q_2)V_y = -I_t(q_2) \quad (5)$$
$$\vdots$$
$$I_x(q_n)V_x + I_y(q_n)V_y = -I_t(q_n) \quad (6)$$

Where$q_1, q_{2,\ldots,} q_n$ are the pixels inside the window, and $I_x(q_i)$,$I_y(q_i)$,$I_t(q_i)$ are the partial derivatives of the image *I* with respect to position *x*, *y* and time *t*, evaluated at the point q_i and at the current time.

These equations can be written in matrix form $Av = b$, where

$$A = \begin{bmatrix} I_x(q_1) & I_y(q_1) \\ I_x(q_2) & I_y(q_2) \\ \vdots & \vdots \\ I_x(q_n) & I_y(q_n) \end{bmatrix}, \quad v = \begin{bmatrix} V_x \\ V_y \end{bmatrix}, \quad \text{and}$$

$$b = \begin{bmatrix} -I_t(q_1) \\ -I_t(q_2) \\ \vdots \\ -I_t(q_n) \end{bmatrix} \quad (7)$$

This system [12] has more equations than unknowns and thus it is usually over-determined. The Lucas-Kanade method using the least squares principle and obtains a compromise solution. Namely, it solves the 2×2 system

$$A^T Av = A^T b \text{ or } v = (A^T A)^{-1} A^T b \quad (8)$$

Where A^T is the transpose of matrix A. That is, it computes with the sums running from *i*=1 to *n*.

The matrix A^TA is often called the structure tensor of the image at the point *p*.

The same importance is given to all *n* pixels q_i in the window as the plain least squares solution. In practice it is usually better to give more weight to the pixels that are closer to the central pixel *p*. For that, one uses the weighted version of the least squares equation,

$$A^T WAv = A^T Wb$$
Or
$$v=(A^T WA)^{-1} A^T Wb \quad (9)$$

Where *W* is an n×n diagonal matrix containing the weights $W_{ii} = w_i$ to be assigned to the equation of pixel q_i. The weight w_i is usually set to a Gaussian function of the distance between q_i and *p*.

Due to the disadvantage of L&K algorithm in the accuracy of the estimations of large displacements, high performance approach is consider with the multi-scale implementation that computes the optical flow velocity components for each different resolution with good accuracy.

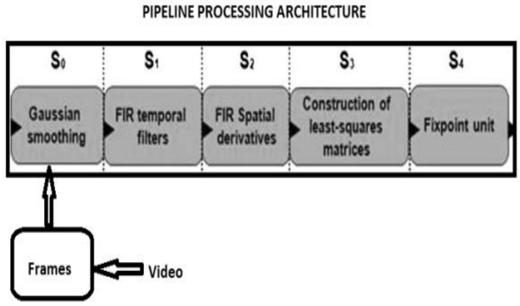

Fig.4. Pipeline Processing Architecture

L&K algorithm uses pipeline processing type of architecture for faster computations. This process is shown in the Fig.4. These different stages are used to find the optical flow estimations. The core computation is divided in to five stages.

D. Multi Scale Estimation

The L&K algorithm is solved with a coarse-to-fine multi-scale implementation [12] based on the high performance flow computation because of the limited motion estimation.

Multi-scale estimation is shown in Fig.5. This architecture required for the pyramid building. Here motions between frames are estimated by using this several steps.

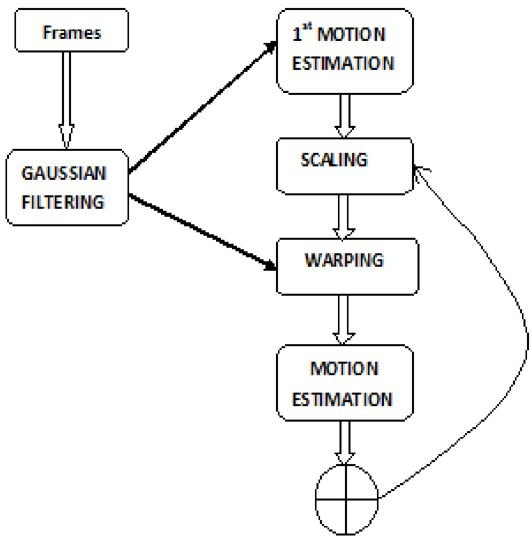

Fig.5. The Flow Diagram Of Multi Scale Optical Flow

III. SIMULATION RESULTS

Optical flow algorithm is simulated by using video files. This video file is given as input in Matlab R2007b and the output waveform is seen in the ModelSim-Altera 6.3g_p1 tool.

A. Matlab Simulation

Matlab simulation has two steps. In first step video file

*Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication*

is given as input and the second step is image file processing. As shown in Fig.6. the video file (.avi) is converted into sequence of frame1, frame2, frame3, frame4, frame5, frame6.

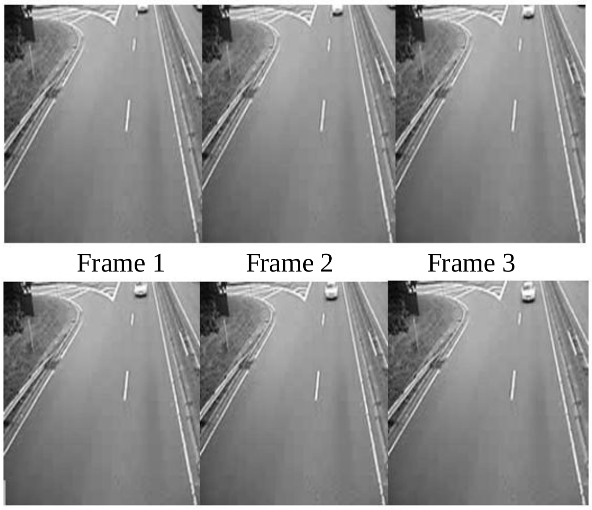

Fig.6. Video to Frame Sequence

The colour image file is converted into gray scale image and also corresponding text file which contains its pixel value.

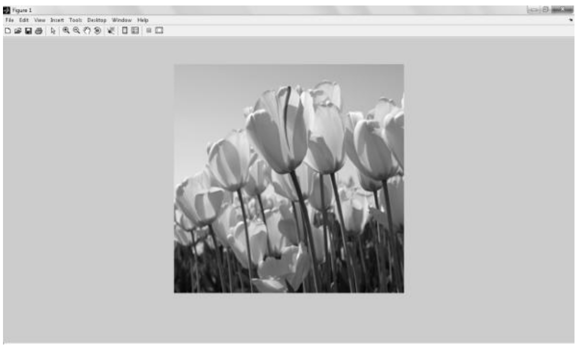

Fig.7. Matlab Simulation

The simulation result is shown in the Fig.7.This process is done with the help of matlab and the image is resized into 256×256.

B. Modelsim Simulation Output

Optical flow is estimated by using the verilog code in modelsim simulation tool.It has several steps for simulation result.

The pixel values are given to find the Gaussian filtering. In Gaussian filtering the pixel values are compressed. The four pixel values are compressed into one pixel and stores in same memory location.

The compressed pixel values are again given for spatial, temporal filtering and L&K optical flow core estimation is done in modelsim tool. In L&K optical core the pixel values are multiplied into some convolution function.

Fig.8. contains the filter output which gives the optical flow estimation. The output is true only when the difference in pixel presents. The convoluted pixel values are given as input and the corresponding difference is obtained. If the difference is present then the value is one otherwise the value is equal to zero.

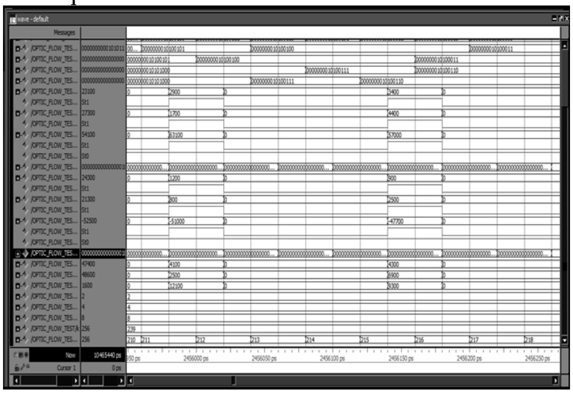

Fig.8. Optical Flow Estimation Output

The optical flow estimation is done with the help of matlab and modelsim tools effectively. The final output is get from the video input by applying several filtering stages [13]. The performance of the system is evaluated using the difference in pixel input.

IV. CONCLUSION

In this proposed system a multi-scale implementation of the well-known L&K method is selected due to its good tradeoff between accuracy and efficiency. This work has been implemented with the help of Matlab and Modelsim software.High system performances and processing speed is done with the help of fine pipelined architecture. High performance Optical flow is estimated in this proposed system.

V. FUTURE EXPANSION

- To achieve high performance optical flow, this system have to be implement in FPGA devices like Altera DE2-70 because VGA resolution taking full advantage of massive parallelism of the FPGA.
- To obtain good accuracy rates the system is going to implement in FPGA device.
- To obtain the target application requirements with hardware resource utilization the system is going to design.

REFERENCES

1. Andreas Gustafsson "Interactive Image Warping" Helsinki University Of Technology May 1993.
2. David J. Fleet, Yair Weiss "Optical Flow Estimation" pp.1-24,2005.
3. D.J. Fleet and A.D. Jepson "Optical Flow Estimation" vol. no 2503 pp.1-22, 2005.
4. J. Diaz, E. Ros, F. Pelayo, E. M. Ortigosa, and S. Mota, "FPGA-based real-time optical-flow system," IEEE Trans. Circuits Syst. for Video Technol., vol. 16, no. 2, pp. 274–279, Feb. 2006.
5. Zhaoyi Wei, Dah-Jye Lee and Brent E. Nelson Department of Electrical and Computer Engineering, Brigham Young University, Provo, Utah, 84602 USA "FPGA-based Real-time Optical Flow Algorithm Design and Implementation" Journal Of Multimedia, Vol. 2, No. 5, September 2007.
6. K. Pauwels and M. M. V. Hulle, "Realtime phase-based optical flow on the GPU," in Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recog. Workshops (CVPRW), pp.1-8, 2008.

7. Michael Chisholm, Dr. Eric McCreath "Calculating Optical Flow using FPGAs" 13th June 2008.
8. F. Barranco, J. Diaz, E. Ros, and B. del Pino, "Visual system based on artificial retina for motion detection," IEEE Trans. Syst., Man,Cybern., vol. 39, no. 3, pp.752–762, Jun. 2009.
9. M. M. Abutaleb, A. Hamdy, M. E. Abuelwafa, and E. M. Saad "A Reliable FPGA-based Real-time Optical-flow Estimation" International Journal of Electrical and Electronics Engineering 2010.
10. Guillermo Botella, Antonio Garcia, Manuel Rodriguez-Alvarez, Eduardo Ros, Uwe Meyer-Baese María C. Molina "Robust Bioinspired Architecture for Optical-Flow Computation" IEEE Transactions On Very Large Scale Integration Systems, Vol. 18, No. 4, April 2010.
11. Francisco Barranco, Matteo Tomasi, Javier Diaz, Mauricio Vanegas, and Eduardo Ros "Parallel Architecture for Hierarchical Optical Flow Estimation Based on FPGA" IEEE journal 2011.
12. Kui Liu, Qian Du, He Yang, and Ben Ma "Optical Flow and Principal Component Analysis-Based Motion Detection in Outdoor Videos", Mississippi State University, MS 39762, USA January 2010.
13. James R. Bergen, P. Anandan, Keith J. Hanna, and Rajesh Hingorani "Hierarchical Model-Based Motion Estimation" David Sarnoff Research Center, Princeton NJ 08544,USA.