Major Project

On

Analysis of Stereo Imaging Algorithm and Parameterization for Implementation on FPGA

Submitted in partial fulfillment of the requirements

For the degree of

Master of Technology in Computer Science & Engineering

By

THAKKAR GAURANG S.
(06MCE021)

Under Guidance of

Dr. S. N. PRADHAN

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
INSTITUTE OF TECHNOLOGY
NIRMA UNIVERSITY OF SCIENCE & TECHNOLOGY
AHMEDABAD 382481
MAY 2008

This is to certify that Dissertation entitled

Analysis of Stereo Imaging Algorithm and Parameterization for Implementation on FPGA

Submitted by

THAKKAR GAURANG S.

has been accepted towards fulfillment of the requirement
for the degree of
Master of Technology in Computer Science & Engineering

Dr. S. N. Pradhan
P. G. Coordinator

Prof. D. J. Patel
Head of
Department

Prof. A. B. Patel
Director, Institute of Technology

CERTIFICATE

This is to certify that the Major Project entitled **"Analysis of Stereo Imaging Algorithm and Parameterization for Implementation on FPGA"** submitted by **Mr. Thakkar Gaurang (06MCE021)**, towards the partial fulfillment of the requirements for the degree of Master of Technology in Computer Science & Engineering, Nirma University of Science and Technology, Ahmedabad is the record of work carried out by him under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this major project, to the best of my knowledge, haven't been submitted to any other university or institution for award of any Master degree.

Project Guide:-
Dr. S.N. Pradhan
P.G. Coordinator,
Department of Computer Science & Engineering,
Nirma University,
Ahmedabad.

Date:-

ACKNOWLEDGEMENT

It gives me immense pleasure in expressing thanks and profound gratitude to **Dr**. **S. N. Pradhan**, P.G. Coordinator, Computer Science & Engineering Department, Nirma University, Ahmedabad for his valuable guidance and continual encouragement throughout my Major project. I also express my sincere thanks to **Mrs. Swati Jain,** Lecturer, Computer Science & Engineering Department, Nirma University, Ahmedabad for her support and continuous guidance. I am heartily thankful to them for his precious time, suggestions and sorting out the difficulties of my topic that helped me a lot during this study.

I would like to give my special thanks to **Prof. D. J. Patel**, Head, Computer Science & Engineering Department, Nirma University, Ahmedabad for his encouragement and motivation throughout the Major Project. I am also thankful to **Prof. A. B. Patel**, Director, Institute of Technology, Nirma University, Ahmedabad for his kind support in all respect during my study.

I am thankful to all my friends and faculty members of Computer Science & Engineering Department, Nirma University, Ahmedabad for their special attention and suggestions towards the project work.

Thakkar Gaurang S.
Roll No. 06MCE021

ABSTRACT

In this era of information technology the image processing is applied in various applications such as geo-informatics for e-governance and others. As we know typical satellite images are in very large number and even a single image is of such a high resolution. Hence efficient implementation is required to meet the real time constraints. These image processing algorithms are usually implemented in software but may also be implemented in special purpose hardware to meet timing constrains. Therefore to achieve satisfactory speed up for execution of image processing algorithms special purpose processors like DSPs or DSP-FPGA hybrid architecture can be used.

The aim of this dissertation is to study some compute intensive algorithms of Stereo Image Matching and parameterize it to optimize the implementation on FPGA. The hierarchical image matching algorithm has been studied, implemented and two variations have been proposed: one is to vary the correlation window size that means to use smaller correlation window to reduce the computations involved. The other is to apply new correlation based formula as a cost function of similarity in conjunction with hierarchical image matching algorithm. The results are obtained by applying these two variations and compared with the results obtained keeping correlation window size reasonably large as well as the original normalized cross correlation formula.

These results are analyzed and a novel solution to this problem is proposed which could result in exact matches and good efficiency of the algorithm at the same time without sacrificing required accuracy.

CONTENTS

LIST OF FIGURES

1. INTRODUCTION

1.1 General

Digital image processing is characterized by very high computational demands. Although it can be handled by "standard" computer hardware, such solution is not viable for an embedded system, where dimensions of the computer system, power consumption or data throughput are of concern. For these reasons, specialized hardware solutions based on a digital signal processor (DSP) or a Field Programmable Gate Array (FPGA) are usually used in embedded systems. As increasingly complex algorithms are implemented using digital signal processing, the performance demands of these algorithms rise exponentially. For cost-sensitive, high-volume applications like stereo image processing, online video processing, development of extremely specialized ASP (Application specific processors) are driven. However, for many other applications, the only options for implementing high-performance digital signal processing have been general-purpose DSPs and, more recently, FPGAs. Available Processor types range from general purpose processors that handle a wide variety of applications, to application-specific processors like DSPs, which are specific to a particular application class such as signal processing, to single purpose processors, which are customized to a very specific function.

The heart of any digital signal processing architecture is the Multiply-and-Accumulate (MAC) unit. Most signal processing applications utilize a great deal of multiplication: The MAC unit of a DSP accelerates this type of calculation by performing the multiplication of two numbers and then adding the result to all of the previous multiplications in what is called an "accumulator". Another key enabling technology of DSPs is the ability to process several operations at the same time. One way that DSPs can execute four operations at the same time is to use what is known as Very Long Instruction Word (VLIW) architecture. A VLIW is a single instruction that actually represent several operations. DSPs have typically been used to implement many of these applications. Although DSPs are

programmable through software, the DSPs' hardware architecture is not flexible. Therefore, DSPs are limited by fixed hardware architecture such as bus performance bottlenecks, a fixed number of MAC blocks, fixed memory, fixed hardware accelerator blocks and fixed data widths. The DSPs' fixed hardware architecture is not suitable for algorithms that want to exploit parallelism in either data or instruction.

The architecture of FPGA, on the other hand, is designed with fine-grain parallelism, which makes it well suited for massively parallel algorithms. The basic characteristics of FPGA are relatively small capacity of the on-chip memory and relatively narrow throughput of memory interfaces, lack of wide-word processing units, and high cost of performing complex numerical operations, such as division, square root, logarithmic, exponential, and trigonometry functions (in smaller devices, these operations cannot be implemented at all). FPGAs provide a reconfigurable solution for implementing traditional applications and offer higher throughput along with DSP. Systems implemented in FPGA and DSPs can have customized architecture, customized bus structure, and customized memory, customized hardware accelerator blocks and a variable number of MAC blocks. A major advantage of FPGAs for many system architectures is that FPGA can work as accelerators for certain computations along with DSP, to increase the throughput of the over all system. System architects use this capability to create products with various price points and performance capabilities without significantly affecting development costs or inventory. FPGA devices provide a reconfigurability which can be useful in changing design which is already ported into FPGA. FPGA devices incorporate a variety of embedded features such as embedded processors, memory blocks, etc. Using FPGAs, DSP designers can customize their design to partition between DSPs and FPGAs for optimal implementation of their applications. Thus DSP designer can get the performance, utilization of resources and faster execution of applications.

Stereo Image Processing is a special class of image processing area where high amount of computation power is required. Stereo Image

Processing implements functions like image correction, image rectification, image matching and disparity calculations like tasks. In this tasks image matching is a very basic but most expensive task. Stereo image matching requires two preprocessed images and matching of these images done by suitable algorithm. Results of algorithms for image matching may vary because different algorithms have different computation costs, running time, accuracy as well as artificial intelligence may also be involved. In this dissertation Hierarchical matching algorithm is used for the level by level comparisons and generating accurate results. This dissertation covers design, implementation and analysis of the algorithm on FPGA.

1.2 Motivation

Image processing is a one of the fast developing research area of computer vision. Faster image processing is very essential in current scenario for work automation .This work is useful in developing the vision using the computerized analysis , object detection and classification of the images captured by the sensors for better interpretation and analysis. Generally satellite images are of high resolution 4Kx4K, 16Kx16K and higher resolutions. To support the processing of these images we require special purpose processors like DSPs or ASPs specifically designed for these types of applications.

Primary use of these high resolution images is in the generating computerized earth surface model by using DEM. This DEM can be use by government/military for administration, security, surveillance. For DEM generation stereo image processing is performed on the images. Stereo Image Processing require image equalization, image rectification, image matching, Disparity map/Depth map generation and finally DEM generation from the Depth map. The term image matching means automatically correspondence establishment, between primitives extracted from two or more (digital) images depicting at least partly the same physical objects in space. Thus the 3D information of the objects can be computed. It is also a core step for 2D or 3D object localization and tracking, and very often a prerequisite for object detection, classification and identification.

Image matching itself is a very difficult problem. A fully automatic, precise and reliable image matching method, to adapt to different images and scene contents, does not exist yet. The limitations arise mainly from an insufficient understanding and modeling of the underlying processes (human stereo vision) and the lack of appropriate theoretical measures for self-tuning and quality control. Stereo image matching is used to reconstruct the 3D information from 2D images. During the image acquisition, the 3D world is projected onto the image and this projection causes information loss (this is most evident in the case of occlusions. Image matching is ill-posed problem, because for a given point in one image, its correspondences on other image may not exist due to occlusion, because there may be more than one possible match due to repetitive texture patterns or a semi-transparent object surface, and because the solution may be unstable with respect to image noise or poor textures. For reducing computations, the search space for these parameters must be constrained and reduced.

1.3 Scope of Work

As the title suggests the goal of this dissertation focuses on the providing solution of high computation need for stereo image processing. To achieve this goal, work carried out in this research is useful for the organizations which require high computation power for processing of high resolution satellite images. Stereo imaging is one of the application which demands more computation power for generating disparity map and depth calculation of objects on the earth. By testing basic image processing algorithms on different DSPs as well as on FPGAs, suitable processor can be decided for performing image matching for the calculation of disparity between the stereo image pair. This includes selection of right DSP processors and development of algorithms by partitioning the parallel computation tasks between the FPGA (Spartan-3) and DSP Processor for increased throughput. Wherever intensive parallelism is available in algorithm that module is performed in FPGA. Therefore, before implementation of stereo imaging algorithm on FPGA, the algorithm needs to be prototyped and tested for correct result and efficient

implementation. Procedure of developing prototype and parameterization of image processing algorithm is carried out for various steps of Stereo Reconstruction Pipeline for FPGA. FPGA Development is to be done in Handel-C language and verification of results is to be done through the Handel-C simulator using Xilinx Spartan-3(1500L) FPGA. The proposed architecture will produce outputs in terms of disparity map and depth map for the stereo image pair and analysis of parameters involved in algorithm. These images and parameters evolved through procedure can be used for efficient stereo reconstruction (Generating 3d space from 2d images) algorithm that can produce results efficiently while implemented on FPGA.

1.4 Outline of thesis

The thesis is organized as follows:

- Chapter 2 provides brief introduction about stereo imaging. The basic principle used in stereo matching algorithm and how disparity could be measured from stereo image pair. Camera models and camera parameters are discussed. Epipolar geometry, image rectification and introduction to FPGA has been discussed.
- Chapter 3 discusses various image matching algorithm. Image matching algorithm has been classified and then algorithm for spatial and frequency domain are discussed.
- Chapter 4 provides detailed description of the hierarchical image matching algorithm and various steps involved in matching process.
- Chapter 5 includes the implementation of the algorithm. It also depicts results for various input images. Then parameters affecting the outcome have been discussed and the effects of varying these parameters are obtained. The data obtained are analyzed.
- Chapter 6 concludes the dissertation work undertaken.

2. LITERATURE SURVEY

2.1 Introduction to stereo imaging

Stereo Imaging is the process of constructing the 3-Dimentional model using the 2-Dimentional Images for better human understanding. The task of building a general purpose computational-vision system is a grand challenge due to the compute-intensive nature of many vision algorithms. However, researchers have been successful in designing algorithms and building systems that deal with some specific tasks of the human vision system. One important feature of the human vision system is its ability to perceive depth of a viewed scene. This ability to perceive depth, known as stereo vision, or stereo imaging is made possible by the difference in viewpoints of the scene when sensed by our left and right eyes. The information about depth in a scene is of great importance because it helps us navigate in a three-dimensional environment and aids us in recognizing objects of interest, among other tasks. In computer based stereo-vision systems, a stereo-rig is a pair of cameras placed side-by-side, much like our eyes, to capture the left and right images. The processing required extract depth information from the image pair when performed by the human brain due to its immense and complex computational capabilities. In a stereo-vision system, this processing is carried out using a computing platform that can be based on software, hardware, or a mixture of the two. The depth information is encoded in the disparity, defined as the difference in pixel locations of corresponding points in the image pair. The disparity is inversely proportional to the distance of an object from the cameras, so the disparity increases as objects get closer to the cameras. The estimation of this disparity then becomes the primary task of a stereo-vision system.

In the simplest setup of a stereo-rig, where the optical axes of the two cameras are parallel and the vertical axes are aligned, corresponding pixels lie at the same vertical coordinate in the image pair [3]. The search for the corresponding pixel is therefore limited to the same scanline in the image pair, which allows processing of each scanline as they arrive. In the

more general case where the cameras are not aligned as described above, the search for corresponding pixel may span across numerous scanlines and this increases the computational load of the system. When the cameras are not in the ideal setup, Image rectification of input images can be performed. Rectification is the process by which the input image pair is warped to resemble the output from an aligned stereo-rig.

Often, when viewing a scene from different viewpoints as in a stereo setup, objects visible in one image may not be visible in the other image. A foreground object hides, or occludes, different parts of the background in the left and right views, a phenomenon known as occlusion. In addition, the information present at the left edge of the image captured by the left camera is not available in the right image and vice-versa as this part of the scene falls outside the viewing area of the other camera. This further complicates the task of accurate disparity estimation because pixels visible in one image may not have a corresponding match in the other image of the pair. Related areas of Stereo Imaging:

* Aerial Stereo Photogrammetry
* Robotic Vision/Machine Vision
* 3D Computer Graphics
* Computer Vision Geometry

2.2 Binocular disparity

From the pair of 2-D images formed on the retinas, the brain is capable of synthesizing a rich 3-D representation of our visual surroundings [6]. The horizontal separation of the two eyes gives rise to small positional differences, called binocular disparities, between corresponding features in the two retinal images. These disparities provide a powerful source of information about 3-D scene structure, and alone are sufficient for depth perception.

Animals, including humans, with overlapping visual fields have stereoscopic information available to them from a comparison of the images obtained at the two eyes. Each eye sees a slightly different view of the world due to the horizontal separation of the two eyes.

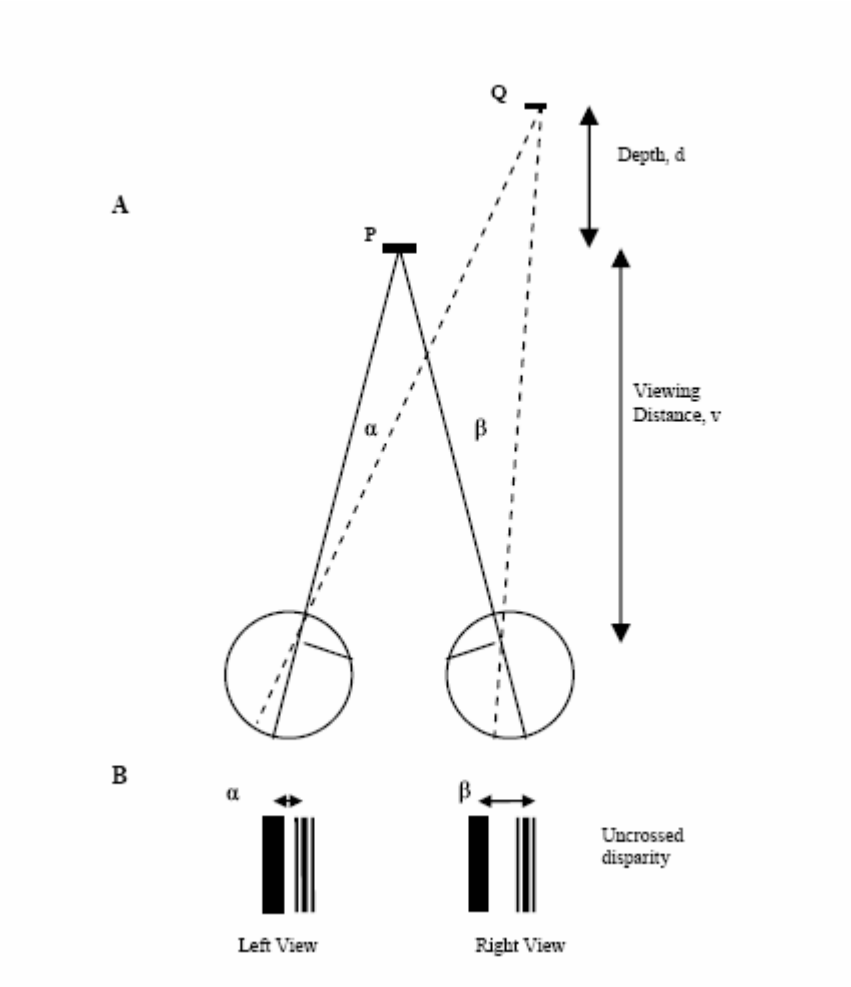


Figure: 2.1 Binocular Disparity

Figure 2.1 shows how the geometry of binocular vision gives rise to slightly different images in the two eyes. If the two eyes are fixating on a point P, then the images cast by P fall at the centre of the fovea in each eye. Now consider a second point Q. If the images of Q fell (say) 5 degrees away from the fovea in both eyes we should say that Q stimulated corresponding points in the two eyes, and that Q had zero disparity. If instead the image was located 6 degrees away from the fovea in one eye but 5 degrees away in the other, we should say that Q stimulated disparate or non-corresponding points and that Q produced a disparity of 1 degree. In general, if Q's image falls α degrees from the fovea in the left eye and β degrees from the fovea in the right eye than the binocular disparity is (β - α), measured in degrees of visual angle. The amount of disparity depends on the physical depth (d) of Q relative to the fixation point P. In fact, disparity is approximately proportional to this

depth difference divided by the square of the viewing distance (v). Thus disparity increases with the amount of depth, but decreases rapidly with increasing viewing distance.

2.3 Camera model specification

A camera is a mapping between the 3D world (object space) and a 2D image. The camera used is central projection camera. All cameras modeling central projection are specializations of the general projective camera. The anatomy of this most general camera model is examined using the tools of projective geometry. The geometric entities of the camera, such as the projection centre and image plane, can be computed quite simply from its matrix representation. Specializations of the general projective camera inherit its properties, for example their geometry is computed using the same algebraic expressions[1].

The specialized models fall into two major classes - those that model cameras with a finite centre, and those that model cameras with centre "at infinity". Of the cameras at infinity the affine camera is of particular importance because it is the natural generalization of parallel projection.

2.3.1 Different camera models

- The Pin Hole Camera Model

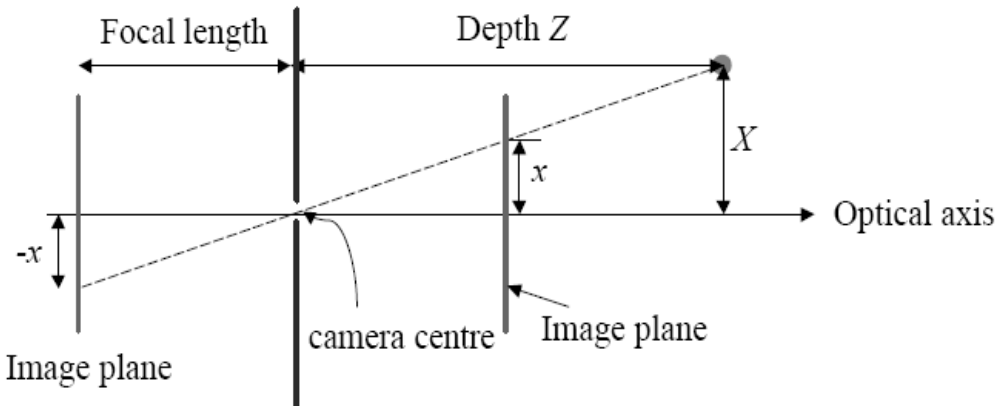


Figure: 2.2 Basic Pin Hole Camera 2D view

f=focal length of camera

X=World coordinate of object in 3D space

x= Image coordinate of object in 2D space of Image plane.

Principal Point = point on image plane which resides on line joining the camera center C and perpendicular to Image plane.

Where relation between x and X is given by

$$x = f\frac{X}{Z} \quad \text{.....(1)}$$

where f is a focal length of the camera lens.

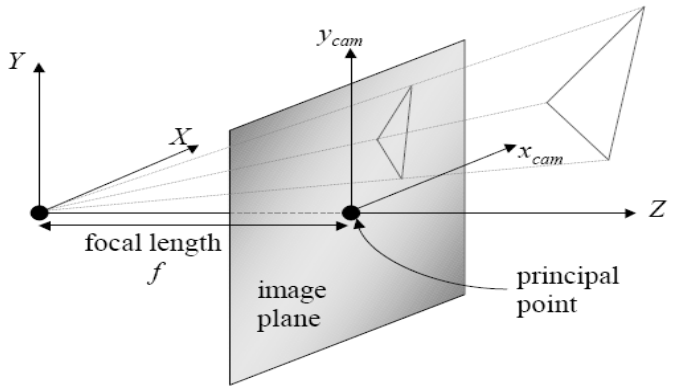


Figure: 2.3 Basic pin hole camera 3D view

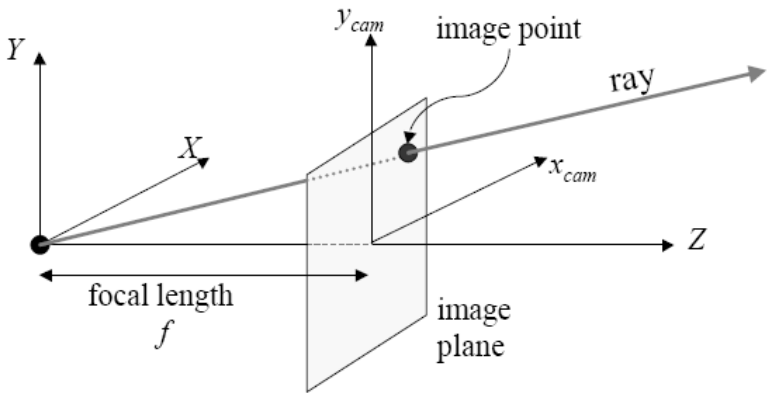


Figure: 2.4 Ray projection of point on image through camera center

- CCD Camera Model

  The pinhole camera model just derived assumes that the image coordinates are Euclidean coordinates having equal scales in both

axial directions. In the case of CCD cameras, there is the additional possibility of having non-square pixels. If image coordinates are measured in pixels, then this has the extra effect of introducing unequal scale factors in each direction. In particular if the number of pixels per unit distance in image coordinates are m_x and m_y in the x and y directions, then the transformation from world coordinates to pixel coordinates is obtained by on the left by an extra factor diag(rn_x,m_y,1).Thus, general form of calibration matrix of CCD is matrix of a CCD camera is :

$$K = \begin{bmatrix} \alpha_x & 0 & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 0 \end{bmatrix} \quad \text{.....(2)}$$

Where, $\alpha_x = fm_x$ and $\alpha_y = fm_y$, represent the focal length of the camera in terms of pixel dimensions in the x and y direction respectively. Similarly, $x_0 = (x_0, y_0)$ is the principal point in terms of pixel dimensions, with coordinates $x_0 = m_x P_x$ and $y_0 = m_y P_y$.

2.3.2 Camera calibration

- Camera Parameters.

1) Internal Parameters.

A single camera is characterized by 4 internal camera parameters.

α_x scaling factor in x-direction.

α_y scaling factor in y-direction.

(x_0, y_0) centre of projection in image plane.

Camera calibration matrix K contains the 4 internal camera parameters:

$$K = \begin{bmatrix} \alpha_x & 0 & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 0 \end{bmatrix}$$

2) External Parameters.

A camera also has an additional 6 external camera parameters (3 translation and 3 rotation) describing the location and orientation of the camera with respect to the world coordinate frame.

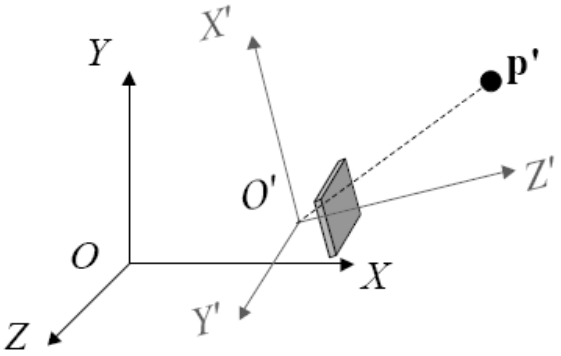


Figure: 2.5 Location and orientation of camera with respect to world frame

2.4 Epipolar Geometry

The key problem in stereo computation is to find corresponding points in the stereo images. Corresponding points are the projections of a single point in the three-dimensional scene. The difference in the positions of two corresponding points in their respective images is called "parallax" or "disparity." Disparity is a function of both the position of the point in the scene, and of the position, orientation, and physical characteristics of the stereo cameras. When these camera attributes are known, corresponding image points can be mapped into three-dimensional scene locations. A camera model is a representation of the important geometrical and physical attributes of the stereo cameras. It may have a relative component, which relates the coordinate system of one camera to the other, and is independent of the scene, and it may also have an absolute component, which relates one of the camera coordinate systems to the fixed coordinate system of the scene.

In addition to providing the function that maps pairs of corresponding image points onto scene points, a camera model can be used to constrain the search for matching pairs of corresponding image points to one dimension (Figure 2.6) [5]. Any point in the three-dimensional world space, together with the centers of projection of two

camera systems, defines a plane (called an "epipolar" plane). The intersection of an epipolar plane with an image plane is called an epipolar line. Every point on a given epipolar line in one image must correspond to a point on the corresponding epipolar line in the other image. The search for a match of a point in the first image may therefore be limited to a one-dimensional neighborhood in the second image plane, as opposed to a two-dimensional neighborhood, with an enormous reduction in computational complexity.

When the stereo cameras are located and oriented such that there is only a horizontal displacement between them, then disparity can only occur in the horizontal direction, and the stereo images are said to be "in correspondence" [4]. When a stereo pair is in correspondence, the epipolar lines are coincident with the horizontal scan lines of the digitized pictures--enabling matching to be accomplished in a relatively simple and efficient
manner. Stereo systems that have been primarily concerned with modeling human visual ability have employed this constraint. In practical applications, however, the stereo pair rarely is in correspondence. In aerial stereo photogrammetry (the process of making measurements from aerial stereo images), for example, the camera may typically be tilted as much as 2 to 3 degrees from vertical. With any tilt, points on a scan line in one image will not fall on a single scan line in the second image of the stereo pair and thus the computational cost to employ the epipolar constraint will be significantly increased. It is possible, however, to re-project the stereo images onto a common plane parallel to the stereo baseline such that they are in correspondence.

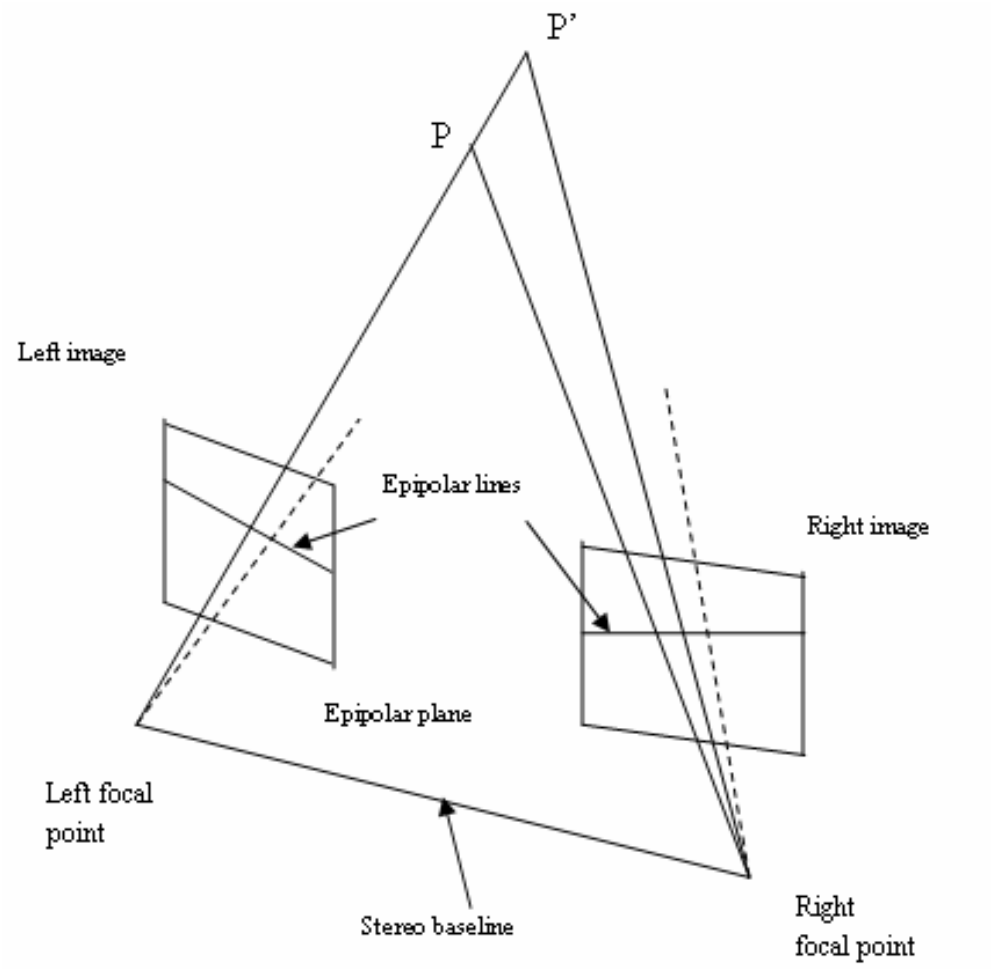


Figure: 2.6 Epipolar geometry [4]

The left and right camera systems are shown in above figure:2.6. The line connecting the focal points of the camera systems is called the stereo baseline. Any plane containing the stereo baseline is called an epipolar plane. Suppose that a point P in the scene is projected onto the left image. Then the line connecting P and the left focal point, together with the stereo baseline, determines a unique epipolar plane. The projection of P in the right image must therefore lie along the line that is the intersection of this epipolar plane with the right image plane. (The intersection of an epipolar plane with an image plane is called an epipolar line.) If the geometrical relationship between the two camera systems is known, we need only search for a match along the epipolar line in the right image.

2.4.1 Various terminology of Epipolar Geometry

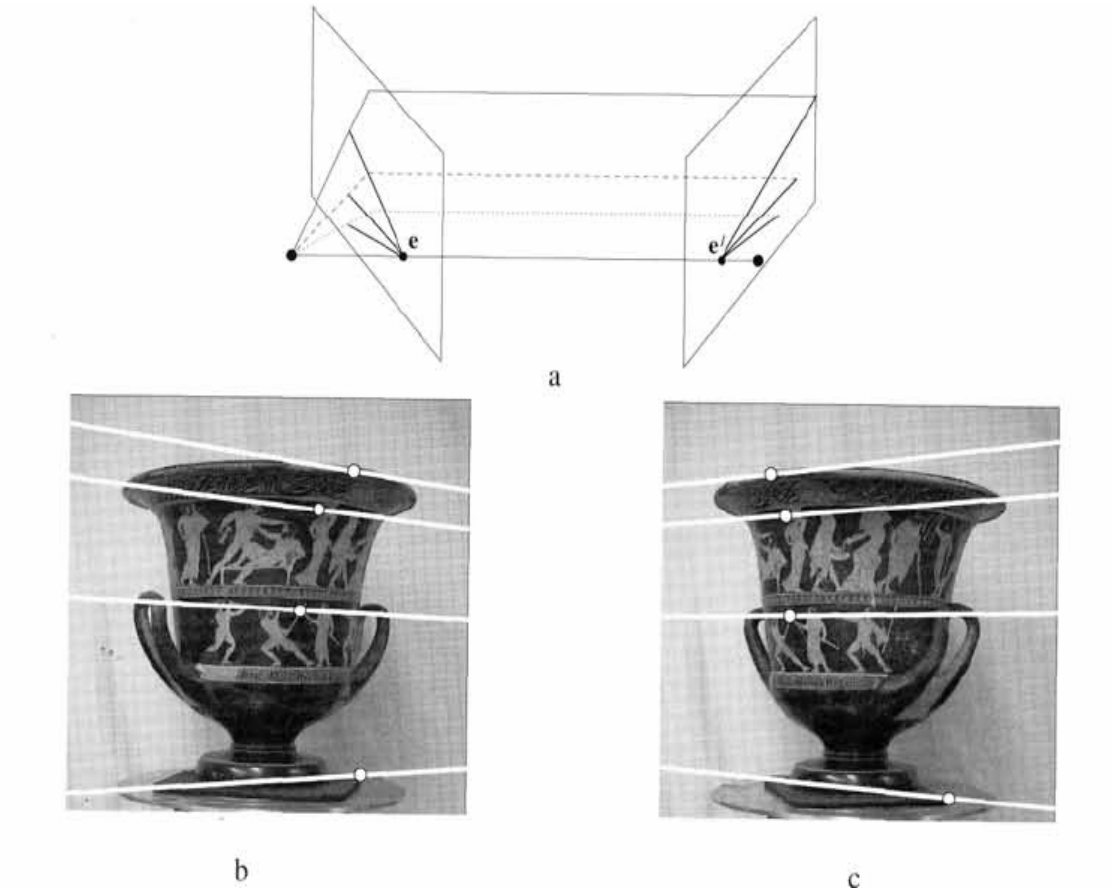


Figure: 2.7 Converging cameras

The epipole is the point of intersection of the line joining the camera centers (the baseline) with the image plane. Equivalently, the epipole is the image in one view of the camera centre of the other view. It is also the vanishing point of the baseline (translation) direction.

An epipolar plane is a plane containing the baseline. There is a one-parameter family (a pencil) of epipolar planes.

An epipolar line is the intersection of an epipolar plane with the image plane. All epipolar lines intersect at the epipole. An epipolar plane intersects the left and right image planes in epipolar lines, and defines the correspondence between the lines.

Terminology shown above are explained well in figure2.7 (a) shows

Epipolar geometry for converging cameras, (b) and (c) show a pair of images with superimposed corresponding points and their epipolar lines (in white). The motion between the views is a translation and rotation. In each image, the direction of the other camera may be inferred from the intersection of the pencil of epipolar lines. In this case, both epipoles lie outside of the visible image.

2.5 Image Rectification

Rectification of an image requires the synthesis of a new image through warping of the original image. The problem of finding correspondence in an image pair taken from cameras in a general position can be simplified by rectifying the image pair before proceeding to find the matches [13]. Rectification is the process by which the two images taken from cameras in a general position are reprojected onto a common image plane that is parallel to the baseline of the stereo-rig. This is illustrated in Figure 2.8 The rectified images can be thought of as acquired by a new stereo-rig, obtained by rotating the original cameras around their optical centers.

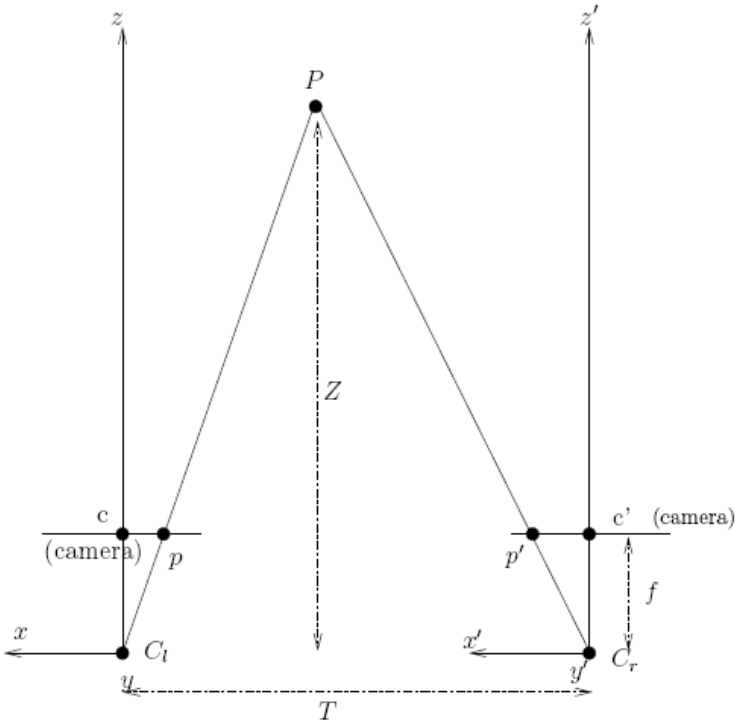


Figure: 2.8 Image rectification

Mathematically, the reprojection can be described by a 3 x 3 projection or homography matrix H. The matrix H represents the transformation of coordinates from the original image to the re projected image as follows:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \mathbf{H} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

The stereo pair can then be rectified by applying two appropriate homographies H_l and H_r to the two images. H_l and H_r are computed from the position and orientation of the two cameras given that the stereo-rig has been calibrated and its intrinsic and extrinsic parameters are known.

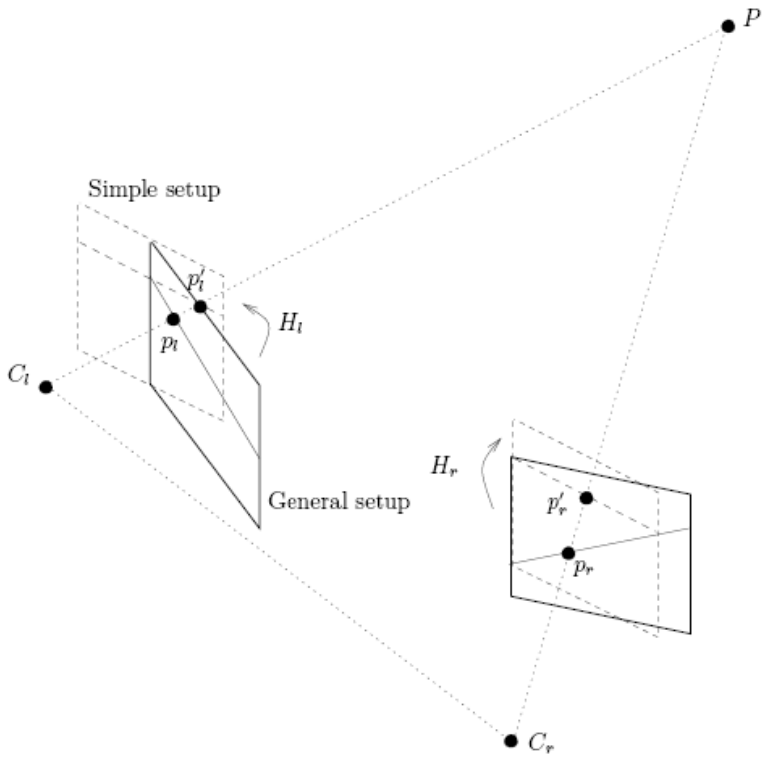


Figure: 2.9 Simple stereo geometry

The actual rectification is performed using backward mapping by re-sampling the original images. For each pixel (x_0, y_0) in the rectified image, the corresponding pixel (x, y) in the original image is computed using Hi1. This backward mapping produces real-valued coordinates in the original

image so the intensity value of each pixel in the rectified image must be interpolated from pixels in this neighborhood. One method of obtaining the intensity values for the rectified image is through bilinear interpolation, which computes the intensity value from a neighborhood of four pixels.

2.6 Introduction to FPGA

Programmable logic is loosely defined as a device with configurable logic and Flip-flops linked together with programmable interconnect. Memory cells control and define the function that the logic performs and how the various logic functions are interconnected. Though various devices use different architectures, all are based on this fundamental idea [10].

There are few major programmable logic architecture available today. Each architecture typically has vendor-specific sub-variants within each type. The major types include:

- Simple Programmable Logic Devices (SPLDs),
- Complex Programmable Logic Devices (CPLDs), and
- Field Programmable Gate Arrays (FPGAs)
- Field Programmable Inter-Connect (FPICs)

2.6.1 FPGA - Field Programmable Gate Array

An FPGA consists of a matrix of logic blocks that are connected by a switching network. Both the logic blocks and the switching network are reprogrammable allowing application specific hardware to be constructed, while at the same time maintaining the ability to change the functionality of the system with ease. As such, an FPGA offers a compromise between the flexibility of general purpose processors and the hardware-based speed of ASICs. Performance gains are obtained by bypassing the fetch-decode-execute overhead of general purpose processors and by exploiting the inherent parallelism of digital hardware.

FPGA is a silicon chip with unconnected logic gates. It is an integrated circuit that contains many (64 to over 10,000) identical logic cells that can be viewed as standard components. The individual cells are

interconnected by a matrix of wires and programmable switches. Field Programmable means that the FPGA's function is defined by a user's program rather than by the manufacturer of the device. Depending on the particular device, the program is either 'burned' in permanently or semi-permanently as part of a board assembly process, or is loaded from an external memory each time the device is powered up.

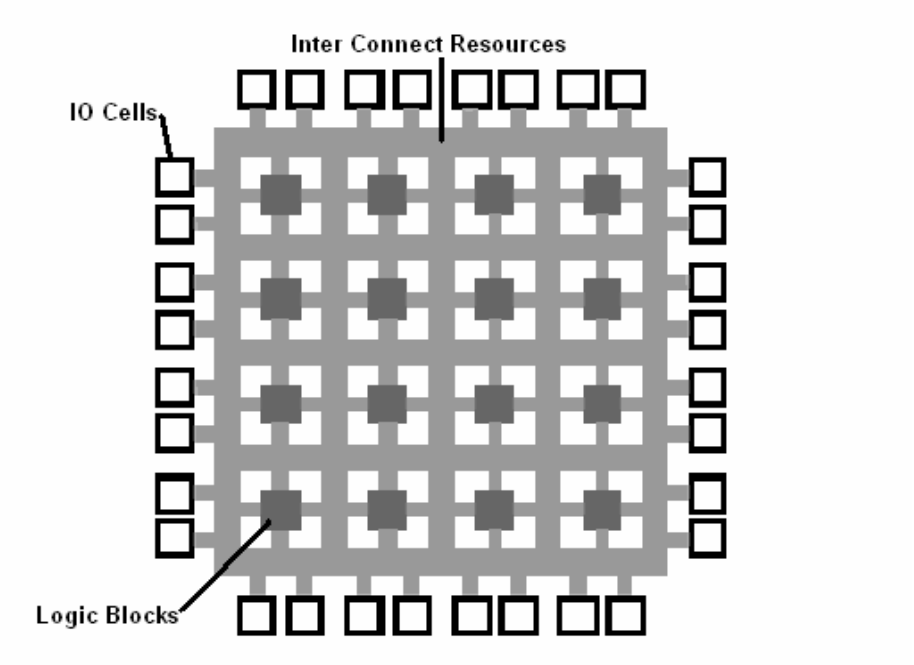


Figure: 2.10 Basic FPGA architecture

The FPGA has three major configurable elements: configurable logic blocks(CLBs), input/output blocks, and interconnects. The CLBs provide the functional elements for constructing user's logic. The IOBs provide the interface between the package pins and internal signal lines. The programmable interconnect resources provide routing paths to connect the inputs and outputs of the CLBs and IOBs onto the appropriate networks.

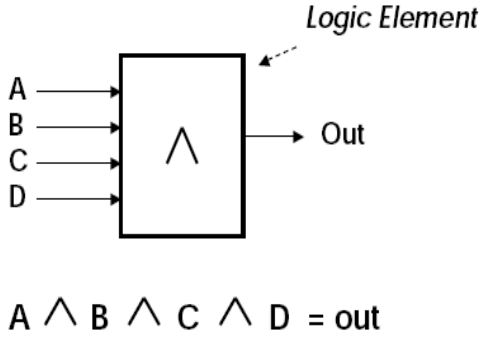


Figure: 2.11 Basic CLB architecture

As shown in Figure 2.11, each CLB contains a logic element which is implemented as a lookup table. This logic element operates on four one-bit inputs and outputs single data bit. Using CLB any Boolean function of four inputs can be performed. These includes 64K functions available functions.

The Field-Programmable Gate Arrays (FPGAs) provide the benefits of custom CMOS VLSI, while avoiding the initial cost, time delay, and inherent risk of a conventional masked gate array. The FPGAs are customized by loading configuration data into the internal memory cells. FPGAs are becoming a critical part of every system design. There are many different FPGAs with different architectures / processes but all of them have the same common feature: that the layout of unit is repeated in matrix form. In this case, the unit is consisting of PLDs, logic gates, RAM, and many other type of components.

There are four main classes of FPGAs currently commercially available: symmetrical array, row-based, hierarchical PLD, and collection of gates. Many emerging applications in communication, computing and consumer electronics industries demand that their functionality stays flexible after the system has been manufactured. Such flexibility is required in order to cope with changing user requirements, improvements in system features, changing protocol and data-coding standards, demands to support variety of different user applications, etc.

An FPGA has a large number of these cells available to use as building blocks in complex digital circuits. Custom hardware has never been so easy to develop. Like microprocessors, RAM based FPGAs can be infinitely reprogrammed in-circuit in only a fraction of a second. Design revisions, even for a fielded product, can be implemented quickly and painlessly. Taking advantage of reconfiguration can also reduce hardware.

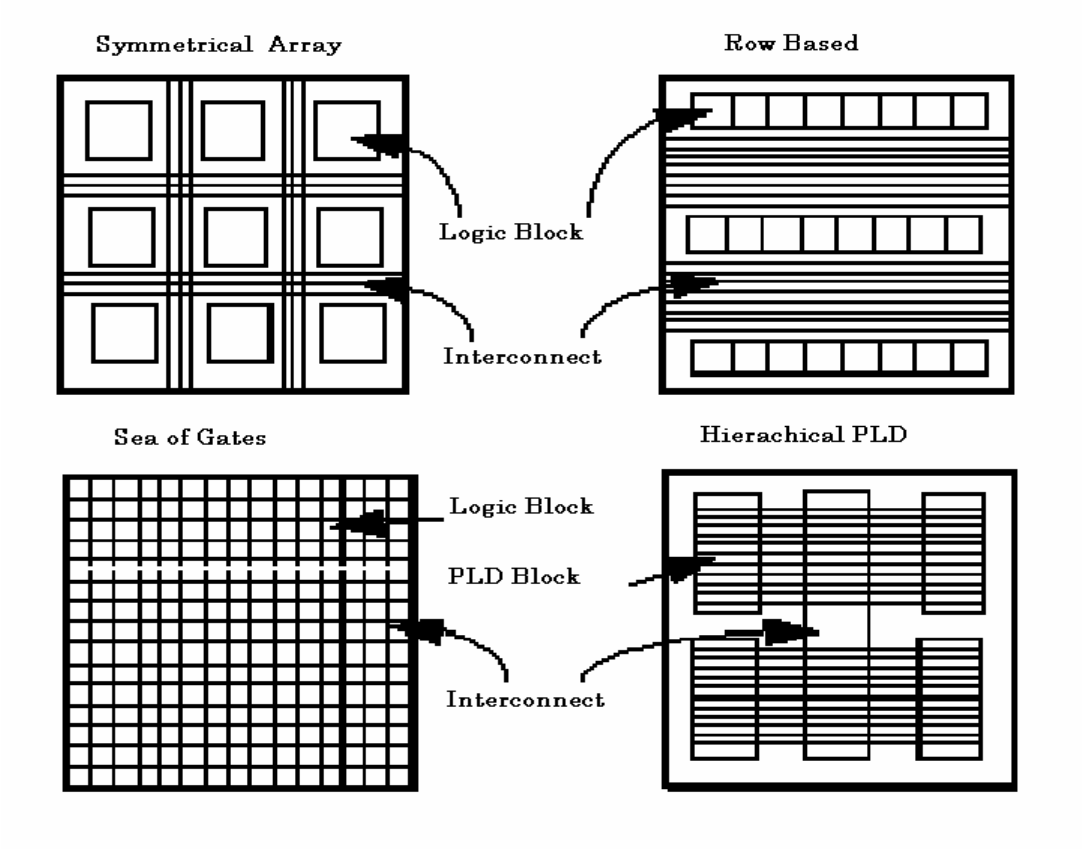


Figure: 2.12 Classes of FPGA architecture

FPGAs are extremely cost-effective at surprisingly high production volumes. FPGA makes simple FIFO - FPGA-based, synchronous FIFO that uses the same clock for read and write operations. Logic networks realized in FPGA are slower by two or three orders of magnitude than those realized in full custom design, but are much faster by several orders than simulation of logic functions by software. Even application program can be run on FPGAs and perform much faster than on general purpose computers in many cases. With FPGAs, debugging or prototyping of new design can be done as easily and quickly as software. As the price of FPGAs goes down with higher speed, FPGAs are replacing other semi-custom design approaches in many applications. FPGA can be pack only about one-tenth of the number of logic gates in ASIC, CMOS VLSI because devices for user programmability such as SRAMs (Static Random Access Memory), non-volatile memory, and anti-fuses, take-up large areas. Thus, for debugging or verifying logic design that needs to be done quickly,

FPGAs are used, and then ASIC,CMOS VLSI are used for large volume production after completing debugging or verification.

Availability of reprogrammable technologies has enabled the configuration of flexible system allowing runtime configuration of system hardware and software. The design methodology combines a C-based software design targeting FPGAs as a device and rapid FPGA hardware design flow based on Handel-C, a C-like programmable language. In order to create an FPGA design, a designer has several options for algorithm implementation. While gate-level design can result in optimized designs, the learning curve is considered prohibitory for most engineers, and the knowledge is not portable across FPGA architectures. Several high-level hardware design languages (HDLs) in which FPGA algorithms may be designed are stated here:

- Verilog HDL
- AHDL-a Hardware Design Language
- VHSIC Hardware Design Language
- Handel-C
- Catapult-C
- SystemC™

3. IMAGE MATCHING ALGORITHMS

3.1 Overview

Stereo correspondence has traditionally been, and continues to be, one of the most heavily investigated topics in computer vision. Many algorithms have been implemented by various researchers. To find correspondence as shown in previous chapters we can use some constraints for the image matching such as epipolar geometry. Among them some very basic and elementary algorithms are discussed in brief.

Now to find a match in right image to left image, there are many ways like:

- Objects?

  Here in this case both of the images under goes through the preprocessing process and objects are found out from both images and then they are matched.

- Edges?

  Here in this case both of the images under goes through the preprocessing process and edges are extracted from both images and then they are matched.

- Pixels?

  Here the gray value of pixel is used for the purpose of matching, and often more than one matches are found and the problem becomes complex to find which among them is the right match. So this does not provide good matching reslults.

- Collections of pixels?

  Here also gray value of pixels is used, but here collection of pixels are used means window surrounding the pixel is used. Most of the matching algorithms developed use this method.

Above methods can be classified in other terms: there are total two classes of algorithm [12].

- Correlation based algorithm:

  In this method all the pixels are matched with the right image and

matches are found. Because of it is done for all pixels it produces dense set of correspondences.

- Feature based algorithm:

Here features of both the images are extracted and then all the features of one image is matched with the extracted features of the other image. Now here it is possible that some of the features of one image may not match with the features in other so that it produces sparce set of correspondence.

Formally it could be defined as a mapping between two images both spatially and with respect to intensity. If we define these images as two 2D arrays of a given size denoted by I_1 and $I_2$where I_1 (x, y) and I_2(x, y) each map to their respective intensity (or other measurement) values, then the mapping between images can be expressed as:

I_2,(x, y) =g(I_1 (f(x, y)))

Where, f is a 2D spatial-coordinate transformation, i.e., f is a transformation which maps two spatial coordinates, x and y, to new spatial coordinates x‘ and Y’

(X’, Y’) =f(x, y)

and g is a 1D intensity or radiometric transformation.

Suppose the epipolar geometry has been applied then for searching the match for one pixel in left image algorithm is only supposed to search for the pixel in right image on epipolar line. Now because of the search constraint the search space has now been reduced to only one dimension from the two dimensional space. But this could be applied only some cases only when the images are rectified. For example if we are searching match for pixel (55,44) and images are rectified and only horizontal disparity is there in images then it should be searched on line y=55. This all depends on type of the image and the application.

As shown in the figure:3.1 below, to get match in pixel(yellow colored) the window surrounding that pixels is considered and it is to be matched with the window surrounding pixels on the epipolar line in the right image.

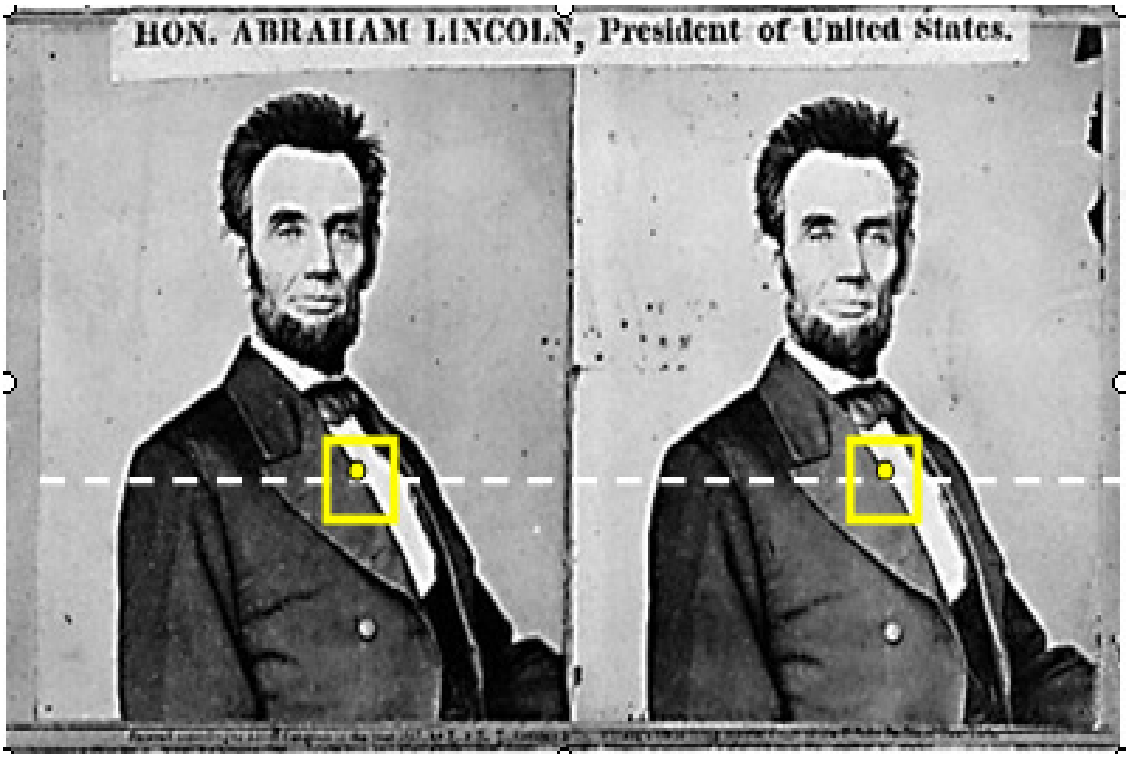


Figure: 3.1 Use of correlation window

3.2 Different methods

- **Cross correlation**

Cross-correlation is the basic statistical approach to registration. It is often used for template matching or pattern recognition in which the location and orientation of a template or pattern is found in a picture. By itself, cross-correlation is not a registration method[7]. It is a similarity measure or match metric, i.e., it gives a measure of the degree of similarity between an image and a template. However, there are several methods for which it is the primary tool, and it is these methods and the closely related sequential methods. These methods are generally useful for images which are misaligned by small rigid or afflne transformations. For a template T and image I, where T is small compared to I, the two-dimensional normalized cross-correlation function measures the similarity for each translation

$$C(u,v) = \frac{\Sigma_x \Sigma_y T(x,y) * I(x-u, y-v)}{\sqrt{\Sigma_x \Sigma_y I^2 (x-u, y-v)}}$$

If the template matches the image exactly, except for an intensity scale factor, at a translation of (i, j), the cross-correlation will have its peak at C(i, j). Thus, by computing C over all possible translations, it is possible to find the degree of similarity for any template-sized window in the image. Notice the cross-correlation must be normalized since local image intensity would otherwise influence the measure. The cross-correlation measure is directly related to the more intuitive measure which computes the sum of the differences squared between the template and the picture at each location of the template:

$$D(u,v) = \Sigma_x \Sigma_y (T(x,y) - I(x-u, y-v))^2$$

This measure decreases with the degree of similarity since, when the template is placed over the picture at the location (*u,v*) for which the template is most similar, the differences between the corresponding intensities will be smallest. The template energy defined as $\Sigma_x\Sigma_y$ T^2 *(x, y)* is constant for each position (*u, v*) that measured. Therefore, it should be normalized using the local image energy $\Sigma_x\Sigma_y$ I^2 (*x-u, y-v*). Notice that if you expand this intuitive measure D*(u, v)* into its quadratic terms, there are three terms: a template energy term, a product term of template and image, and an image energy term. It is the product term or correlation $\Sigma_x \Sigma_y$ *T(x, y) I(x-u, y-v)* which when normalized, determines the outcome of this measure.

A relative measure which is advantageous when an absolute measure is needed is the correlation coefficient

$$\frac{\mathrm{cov}\,ariance(I,T)}{\sigma_I \sigma_T} = \frac{\Sigma_x \Sigma_y (T(x,y) - \mu_T)(I(x-u, y-v) - \mu_I)}{\sqrt{\Sigma_x \Sigma_y (I(x-u, y-v) - \mu_T)^2 \Sigma_x \Sigma_y (T(x,y) - \mu_T)^2}}$$

Where, μ_T and σ_T are mean and standard derivation of the template and μ_I and σ_I are mean and standard deviation of the image. This statistical measure has the property that it measures correlation on an absolute scale ranging from [−1, 1]. Under certain statistical assumptions, the value measured by the correlation coefficient gives a linear indication

of the similarity between images. This is useful in order to quantitatively measure confidence or reliability in a match and to reduce the number of measurements needed when a pre specified confidence is sufficient. Consider a simple example of a binary image and binary template, i.e., all the pixels are either black or white, for which it is possible to predict with some probability whether or not a pixel in the image will have the same binary value as a pixel in the template. Using the correlation coefficient, it is possible to compute the probability or confidence that the image is an instance of the template. We assume the template is an ideal representation of the pattern. The image may or may not be an instance of this pattern. However, if we can statistically characterize the noise that has corrupted the image, then the correlation coefficient can be used to quantitatively measure how likely it is that the image is an instance of the template.

- **SSDA (Sequential similarity detection algorithm)**

A far more efficient class of algorithms than traditional cross-correlation, called the sequential similarity detection algorithms (SSDAS) [7]. Two major improvements are offered.

1. First, they suggest a similarity measure *E*(u, v), which is computationally much simpler, based on the absolute differences between the pixels in the two images.

$$E(u,v) = \Sigma_x \Sigma_y \left| T(x,y) - I(x-u, y-v) \right|$$

The normalized measure is defined as

$$E(u,v) = \Sigma_x \Sigma_y \left| T(x,y) - \hat{T} - I(x-u, y-v) + \hat{I}(u,v) \right|$$

Where T and I are the average intensities of the template and local image window respectively. This is significantly more efficient than correlation. Correlation requires both normalization and the added expense of multiplications. Even if this measure is non normalized a minimum is guaranteed for a perfect match. Normalization is useful, however, to get an absolute measure of how the two images differ, regardless of their intensity scales.

2. The second improvement introduced is a sequential search strategy. In the simplest case of translation registration this strategy might be a sequential threshold. For each window of the image (determined by the translation to be tested and the template size), one of the similarity measures defined above is accumulated until the threshold is exceeded. For each window the number of points that were examined before the threshold was exceeded is recorded. The window which examined the most points is assumed to have the lowest measure and is therefore the best registration. The sequential technique can significantly reduce the computational complexity with minimal performance degradation. There are also many variations that can be implemented in order to adapt the method to a particular set of images to be registered. For example, an ordering algorithm can be used to order the windows tested which may depend on intermediate results, such as a coarse-to-fine search or a gradient technique. The ordering of the points examined during each test can also vary depending on critical features to be tested in the template. The similarity measure and the sequential decision algorithm might vary depending on the required accuracy, acceptable speed, and complexity of the data. Although the sequential methods improve the efficiency of the similarity measure and search, they still have increasing complexity as the degree of freedom of the transformation is increased. As the transformation becomes more general the size of the search grows. On the one hand, sequential search becomes more important in order to maintain reasonable time complexity; on the other hand it becomes more difficult not to miss good matches.

A limitation of both of these methods is their inability to deal with dissimilar images. The similarity measures described so far, the correlation coefficient, and the sum of absolute differences are maximized and minimized, respectively for identical matches. In the next section the Fourier method will be described.

For small translations, rotations, or scale changes. The correlation methods can be used sometimes for more general rigid transformations but become inefficient as the degrees of freedom of the transformation

grows. The Fourier methods can only be used where the Fourier transform of an image which has undergone the transformation is related in a nice mathematical way to the original image. The methods to be described in the next section are applicable for images which have been translated or rotated or both. They are specifically well suited for images with low frequency or frequency- dependent noise; lighting and atmospheric variations often cause low frequency distortions. They are not appropriate for images with frequency-independent noise (white noise) or for more general transformations.

- **Fourier method**

Another useful property of correlation is given by the Correlation theorem. The Correlation theorem states that the Fourier transform of the correlation of two images is the product of the Fourier transform of one image and the complex conjugate of the Fourier transform of the other [2]. This theorem gives an alternate way to compute the correlation between images. The Fourier transform is simply another way to represent the image function. Instead of representing the image in the spatial domain, the Fourier transform represents the same information in the frequency domain. Given the information in one domain, we can easily convert to the other domain. The Fourier transform is widely used in many disciplines; It can be computed efficiently for images using the Fast Fourier Transform or FFT. Hence, an important reason why the correlation metric is chosen in many registration problems is because the Correlation theorem enables it to be computed efficiently, with existing, well-tested programs using the FFT (and occasionally in hardware using specialized optics)[7]. The use of the FFT becomes most beneficial for cases where the image and template to be tested are large. However there are two major caveats. Only the cross-correlation before normalization may be treated by FFT. Second, although the FFT is faster it also requires a memory capacity that grows with the log of the image area. Last, both direct correlation and correlation using FFT have costs which grow at least linearly with the image area.

The method to be described in this section registers images by

exploiting properties of the Fourier Transform. The transform can be efficiently implemented in either hardware or using the Fast Fourier Transform. These methods differ because they search for the optimal match according to information in the frequency domain.

By using the frequency domain, the Fourier method achieves excellent robustness against correlated and frequency- dependent noise. They are applicable, however, only for images which have been at most rigidly misaligned. The most basic method is described. It is called phase correlation and can be used to register images which have been shifted relative to each other.

Phase correlation can be used to align two images which are shifted relative to one another. In order to describe their method, we will define a few of the terms used in Fourier analysis which we will need. The Fourier transform of an image $F(x, y)$ is a complex function; each function value has a real part $R(\omega_x, \omega_y)$ and an imaginary part $I(\omega_x, \omega_y)$ at each frequency (ω_x, ω_y) of the frequency spectrum:

$$F(\omega_x, \omega_y) = R(\omega_x, \omega_y) + iI(\omega_x, \omega_y)$$

This can be expressed alternatively using the exponential form as:

$$F(\omega_x, \omega_y) = \left|F(\omega_x, \omega_y)\right| e^{I\phi(\omega_x, \omega_y)}$$

where $|F(\omega_x, \omega_y)|$ is the magnitude or amplitude of the Fourier transform and where $\Phi(\omega_x, \omega_y)$ is the phase angle. The square of the magnitude is equal to the amount of energy or power at each frequency of the image and is defined as:

$$\left|F(\omega_x, \omega_y)\right|^2 = R^2(\omega_x, \omega_y) + I^2(\omega_x, \omega_y)$$

The phase angle describes the amount of phase shift at each

frequency and is defined as:

$$\phi(\omega_x, \omega_y) = \tan^{-1}\left[I(\omega_x, \omega_y) / R(\omega_x, \omega_y)\right]$$

Phase correlation relies on the translation property of the Fourier transform, sometimes referred to as the Shift Theorem. Given two images F_1 and F_2 which differ only by a displacement (d_x, d_y), i.e.,

$$F_2(x, y) = F_1(x - d_{x,}, y - d_y)$$

Their corresponding Fourier transforms F_1 and F_2 will be related by

$$F_2(\omega_x, \omega_y) = e^{-j(\omega_x dx + \omega_y dy)} F_1(\omega_x, \omega_y)$$

In other words, the two images have the same Fourier magnitude but a phase difference directly related to their displacement. This phase difference is given by $e^{j(\Phi 1 - \Phi 2)}$, It turns out that if we compute the cross-power spectrum of the two images defined as:

$$\frac{F_1(\omega_x, \omega_y) F_2^*(\omega_x, \omega_y)}{\left|F_1(\omega_x, \omega_y) F_2^*(\omega_x, \omega_y)\right|} = e^{(\omega_x dx + \omega_y dy)}$$

Where, F^* is the complex conjugate of F, the Shift Theorem guarantees that the phase of the cross-power spectrum is equivalent to the phase difference between the images. Furthermore, if we represent the phase of the cross-power spectrum in its spatial form, i.e., by taking the inverse Fourier transform of the representation in the frequency domain, then we will have a function which is an impulse, that is, it is approximately zero everywhere except at the displacement which is needed to optimally register the two images.

The Fourier registration method for images which have been displaced with respect to each other therefore entails determining the

location of the peak of the inverse Fourier transform of the cross-power spectrum phase, Since the phase difference for every frequency contributes equally, the location of the peak will not change if there is noise which is limited to a narrow bandwidth, i.e., a small range of frequencies. Thus this technique is particularly well suited to images with this type of noise. Consequently, it is an effective technique for images obtained under differing conditions of illumination since illumination changes are usually slow varying and therefore concentrated at low-spatial frequencies. Similarly, the technique is relatively scene independent and useful for images acquired from different sensors since it is insensitive to changes in spectral energy. This property of using only the phase information for correlation is sometimes referred to as a whitening of each image. Among other things, whitening is invariant to linear changes in brightness and makes the phase correlation measure relatively scene independent.

3.3 Search Strategies Used In Image Registration

Certain search strategies used in image registration are shown below:

- Decision Sequencing:

Improves efficiency for similarity optimization for rigid Transformations.

- Relaxation:

Practical approach to find global transformations when local distortions are present, exploits spatial relations between features.

- Dynamic programming:

Good efficiency for finding local transformations when an intrinsic ordering for matching is present.

- Generalized Hough transformation:

For shape matching of rigidly displaced contours by mapping edge space into "dual-parameter" space.

- Linear programming:

For solving system of linear inequality constraints, used for finding rigid transformation for point matching with polygon-shaped error bounds at each point.

- Hierarchical technique:

Applicable to improve and speed up many different approaches by guiding search through progressively finer resolution. This technique has been explored and discussed in next chapter in detail.

- Tree and graph matching:

Uses tree/graph properties to minimize search, good for inexact and matching of higher-level structures.

4. HIERARCHICAL IMAGE MATCHING

4.1 Introduction

The final accuracy of the final DEM is dependent on conjugate point identification in the stereo images. Hence image matching is the most important task in DEM generation in digital mode. In general the major difficulties in automatic image matching are due to temporal changes of the data sets, different view angles, scale changes between the images and sensor differences etc.

The hierarchical image matching is the most used methodology among available all strategies stated as in previous chapter. This method has advantage of improving of speed up many different approaches by guiding search through progressively finer resolutions [11]. The details of an approach are shown in subsequent topics.

4.2 hierarchical matching technique

Objects represented in the image space may vary enormously in the size and extent. In order to identify and qualitatively describe events in the object space, it is necessary to evaluate and combine the image at different scales, a procedure known as the multi-space technique. Smoothing the original image with a low pass filter of varying sizes results in images at various scales (levels of hierarchy). At each scale, the corresponding images called image pyramids. Selection of optimal number of pyramids depends on the viewing angle of stereo pair used, terrain undulations and the seed point selection in the matching process. After forming the image pyramids hierarchical matching uses four basic steps at each level to get the final match points at the lowest level.

1. Interest point identification.
2. Local mapping between stereo images.
3. Digital correlation up to subpixel accuracy.
4. Blunder detection.

The hierarchical procedure is shown in detail in figure 4.1 and 4.2.

At the highest level of pyramid the seed point are identified manually or through an interest operator on reference image and blind correlation of these points in the other image. The match points of particular level will be used to establish the local correspondence between the stereo pair images at next level.

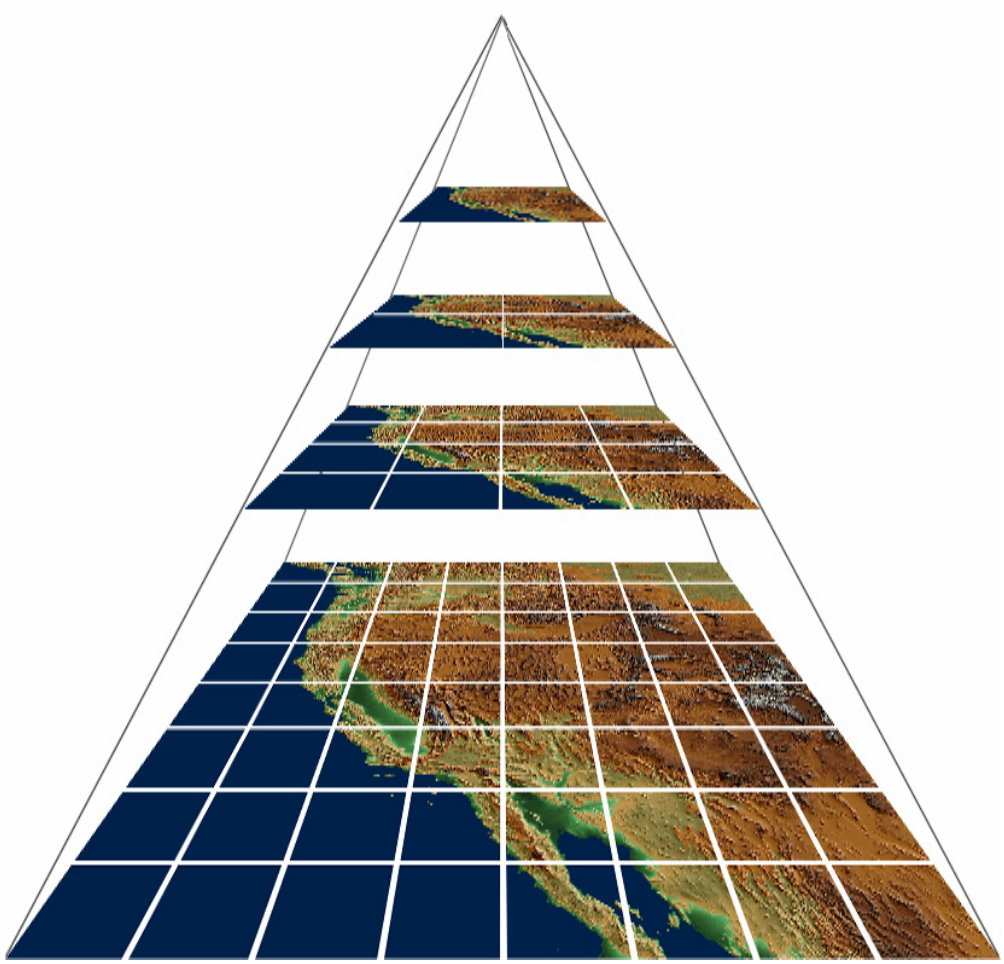


Figure: 4.1 Image pyramid

After this at each level first interest operator is obtained to get some candidate points for match. Local mapping using the previous level's match point establishes a correspondence with the second image for all the interest points. Then digital correlation finds the exact location of the interest points in the second image. Blunder detection eliminates mismatches at each level, if any. This procedure is repeated up to last level that is full resolution. The match points obtained in the last level are the conjugate points for the DEM.

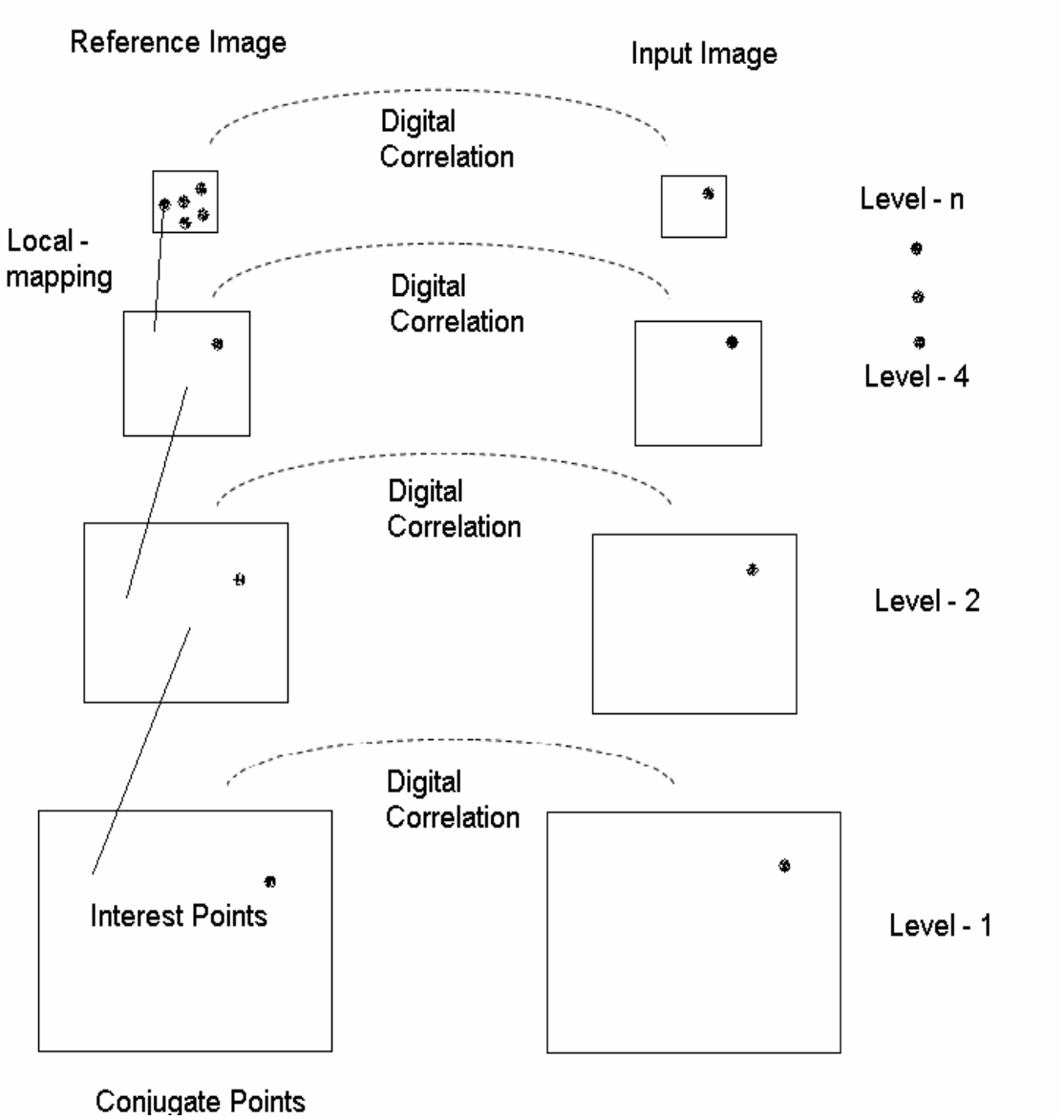


Figure: 4.2 Hierarchical matching method

Hierarchical matching can be performed on raw data directly or the data can be put in epipolar before the image pyramid generation. This has the advantage of using one sided mask. But this has disadvantage of twice the model accuracy in calculating DEM, as we have to go back from epipolar to raw geometry once again after matching, to compute DEM.

4.3 Candidate Feature Selection (Interest Operator)

Good feature extraction is a reliable pre-processing step for good image matching. Therefore selecting reliable and accurate approximate values fro succeeding fine correlation attracts ever –increasing interest. In the field of computer vision and pattern recognition many different operators have been developed for feature extraction. Most widely used operators are Moravec operator and Forstner. In this section and improved Forstner operator, in terms of speed and simplicity in threshold.

The interest operator has two steps.

1. The Ground Operator

At each point, the four gradients to the neighbouring pixels are calculated (As Figure 4.3.).A point is kept only for the second step, if at least two of the gradients are larger than a threshold Dg. As for the determination of Dg it can be manually set to achieve visually acceptable results, before one have found a method, in which the optimal threshold Dg for different images could be automatically determined.

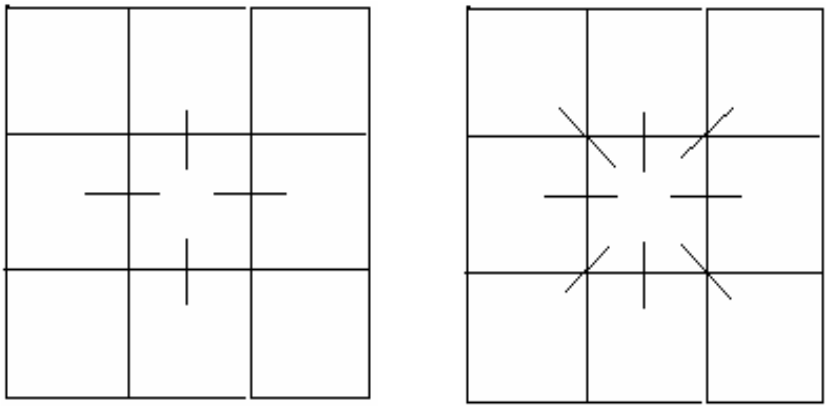


Figure: 4.3 Search space for ground point

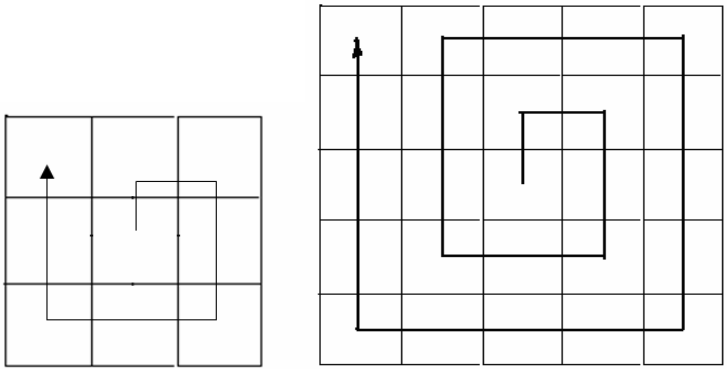


Figure: 4.4 Window for suppression of local non-maxima

Selection of interest points: for the second step two versions are possible:

Version I:

Interest values (IV) for points selected by the ground operator.

$$IV = \sum_{i=1}^{8} abs(Dg_i)$$

Where Dg_i is the difference of grey levels between two adjacent pixels.

Suppression of local non-maxima, in which compared window is progressively enlarged. Once IV within the compared window (the light hatched parts of the window (Figure 4.4), which indicates suppression windows are 3x3 and 5x5) is greater than the center's value of the suppression window, the comparison stops, and the center's IV is set to zero. The size of the suppression window can be selected accordingly to the density of interest points expected for the results.

Improvement of the Forstner operator with respect to speed and accuracy:

The interest operator at each pyramid level gives number of interest points, almost four times of that of number at previous level.

2. Local Mapping:

This is basically to establish a local transformation between reference image and the second image of the stereo pair for each interest point located in the reference image .For any interest point ,nearest ten neighbours in the previous level 's match points are selected and a first order polynomial between match points is used for mapping.

$X_r=a_0+a_1X_1+ a_2Y_1$

$Y_r=b_0+b_1X_1+ b_2Y_1$

(X_r,Y_r) and (X_l, Y_l) are the scan line ,pixels positions of the match points in left (reference image) and right (second image of stereo pair) images.

The polynomial coefficients (a_0, a_1, a_2) and (b_0, b_1, b_2) are obtained

through least square solutions on ten points.

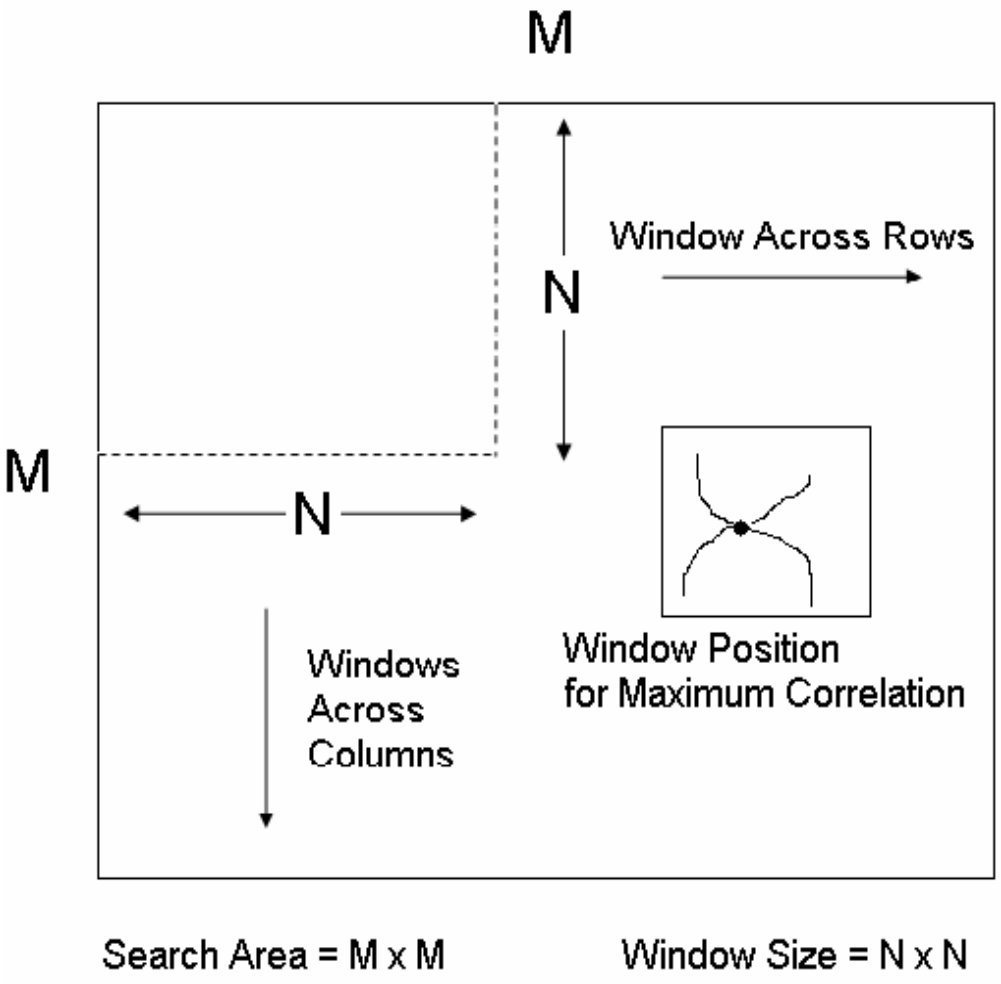


Figure: 4.5 Search space for correlation

4.4 Decision Making and Blunder detection

The following decision making and blunder detection criteria are commonly used in the matching process:

1 Correlation Coefficient.

A decision can be taken, whether a point is matched or not by the correlation coefficient magnitude. As explained earlier the correlation coefficient 1 indicates a perfect match and the window & search areas are extracted from same images. However this will not happen anytime ,since the image chips used are from different images taken at different times .Hence always the correlation coefficient is less than 1. But a threshold can be fixed on correlation coefficient (>0.5) to identify the probable match point . This is further confirmed by two-way correlation ,in

the sense ,a reverse correlation taking reference chip from input image and search chip from reference image. A point is said to be a match point when the correlation coefficients from the both forward and reverse correlations are within a particular limit(<0.01).

2 Epipolar Geometry.

This is another criteria used for strengthening imager matching process by going through object space. The match points in the reference image are transformed to ground by using precise relationship obtained after space resection using GCPs .Then these co-ordinates are transformed to second input (image) using the second image's ground to image relationship .Ideally the obtained scan line pixel co-ordinates should match with earlier computed co-ordinate of the same point through conjugate point matching algorithm. If the matching at that point is correct .However the pixel value will not match, since the height is not used in the computation. At least the scan line difference will show the indication of correct match. A point is considered as match point, if the scan line difference lies within the model accuracy.

3 Height Thresholding.

The minimum and maximum elevations of the area for which DEM is to be obtained ,are known apriori, from the maps .These values can also be used for blunder detection /decision making in deciding match points .If the DEM computed for a point is below the minimum elevation of it is above the maximum elevation then the points are rejected.

4 Interactive Editing.

Though this is time consuming process ,interactive DEM editing is the powerful tool for blunder detection and correlation of the conjugate points identified are digital correlation .in this the conjugate points are viewed through a stereo display mode. Cross cursors are put at the conjugate points locations in both left and right images. The cursors corresponding to, all the points ,which are correctly matched ,normally appear on the surface of the terrain in the stereo mode .The mismatches can easily identifiable in stereo mode, as they appear out of the model

surface or below the model surface . All three points can be identified interactively, and can be deleted or recomputed .This eliminates the blunders, but this process is tedious, if any, mismatches are clustered in a small area.

5. IMPLEMENTATION AND RESULTS

As the dissertation name suggest here the aim is to analyze the stereo imaging algorithm and prepare the prototype of the algorithm and parameterize it for implementation on FPGA. Therefore, the hierarchical image matching algorithm has been identified. The reasons for choosing the hierarchical algorithm has been discussed in previous chapters: the main reason is the speed of the algorithm. The speed of the algorithm is mainly because correlation is not applied for all the pixels present in the image. If the correlation is applied on all the pixels of the image and the satellite images are of such a great resolution, therefore it would take tremendous amount of time. Therefore in Hierarchical algorithm case those points are found out where change in height of the object is possible and these points are called the interest points. Now for these points only the correlation is found out.

5.1 Implementation

As part of the implementation the hierarchical algorithm has been implemented for 5 levels of calculations. The whole implementation has been done in MATLAB (version2007-b). The algorithm has been applied on high resolution satellite images. The algorithm works as follows:

1. From the input image (both left and right) highest level of image pyramid is calculated.
2. From this image interest points are identified.
3. For all interest points correlation is found in second(right image).
4. Now all the interest points and its matches are mapped to image of its lower level.
5. For this lower level image again matching is found and again mapped to its lower level.
6. This process is done till u get to highest resolution original input image.
7. At this level matches gathered are the final matches they are used for disparity map generation.

5.2 Output results

The input stereo image pair [15] & its results are shown as below: The images are actually 974 X 974 resolution. Here shown images are shrinked version of the original images to accommodate in page. After that at highest resolution algorithm with subpixel has been implemented [8][9].

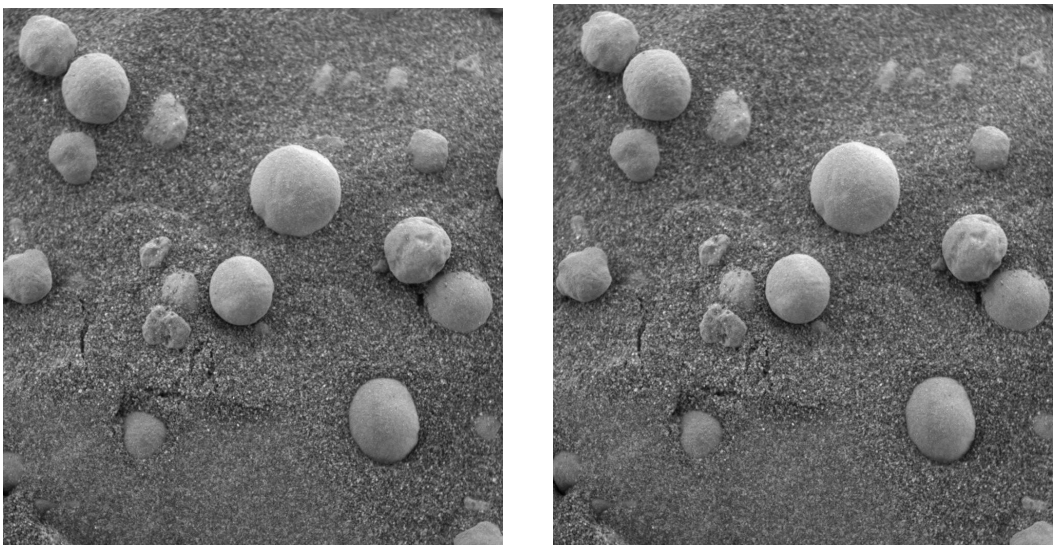
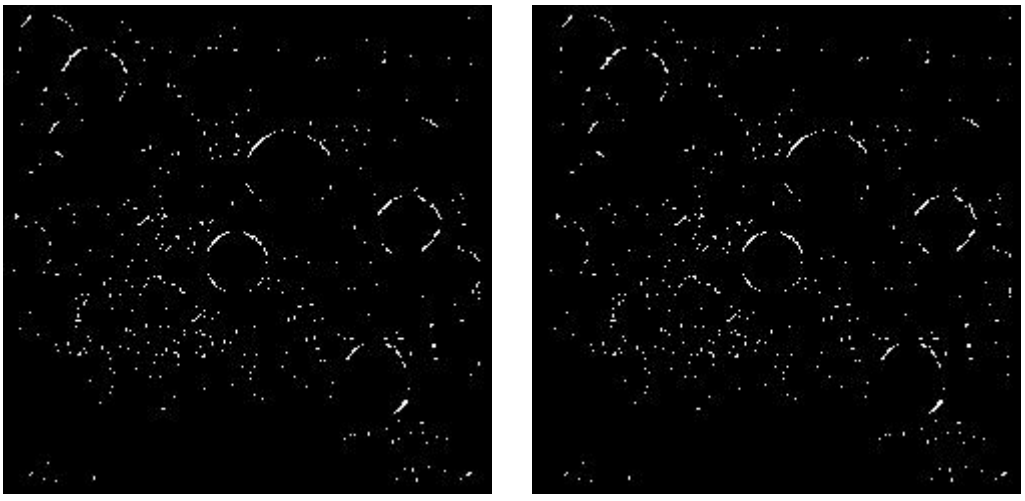


Figure: 5.1 Input Stereo image pair[15]

Here input images are not of such a great resolution. Therefore only it has been scaled down to 3 levels and the matches found at various levels are shown in following figures.



A B

Figure: 5.2 'A' interest points extracted at level4. 'B' matches found of 'A'

There are certain parameters like correlation window sizes, threshold set for finding ground points in calculation of ground operator

'Dg' that can affect the outcome of the algorithm. Here in this case the Dg is set to 20 and window size is 32.

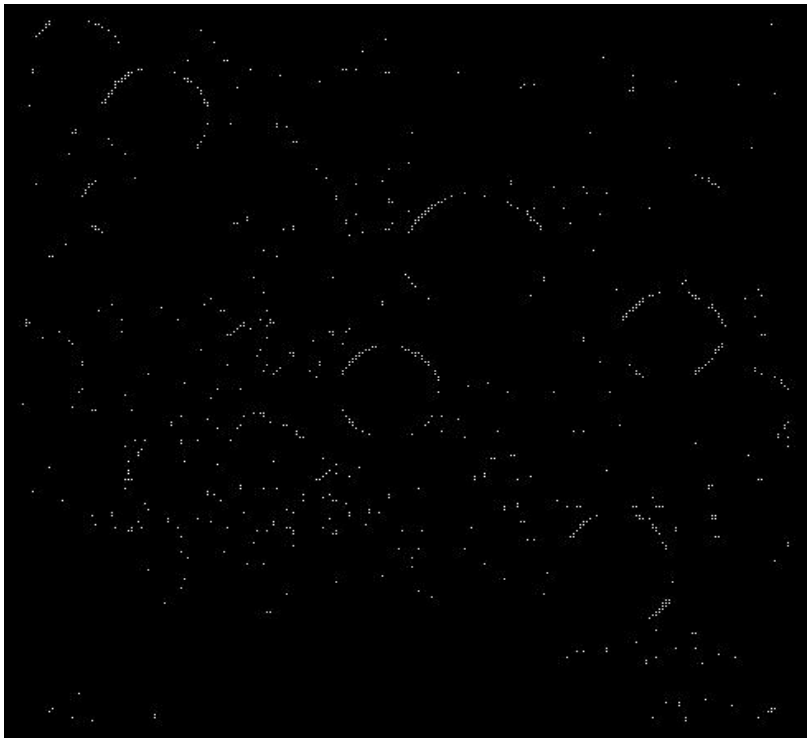


Figure: 5.3 Interest Points at level 2 mapped from level4

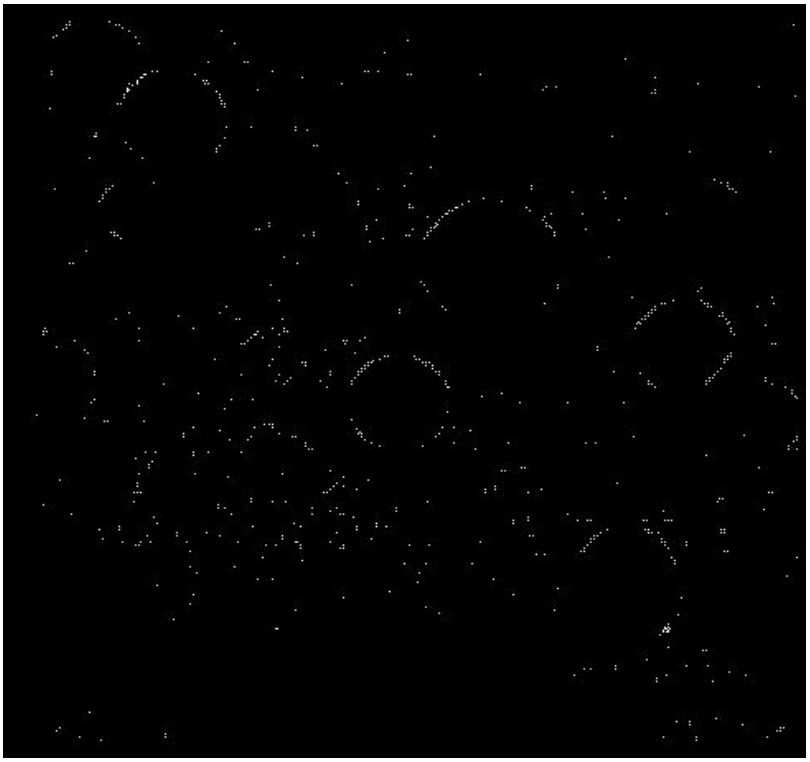


Figure: 5.4 matches found at level 2

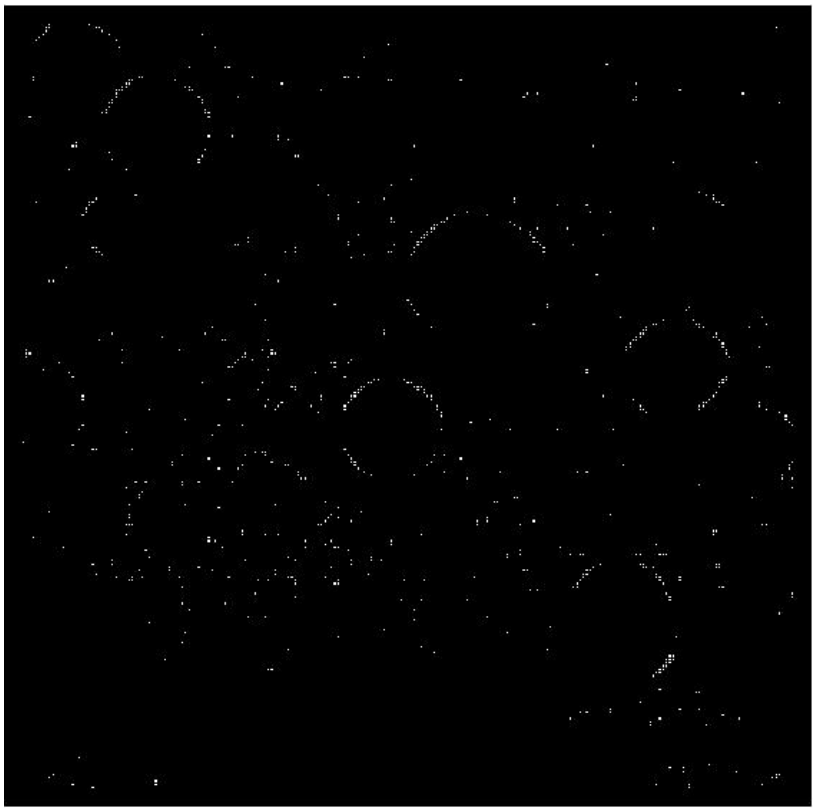


Figure: 5.5 Interest points at level 1

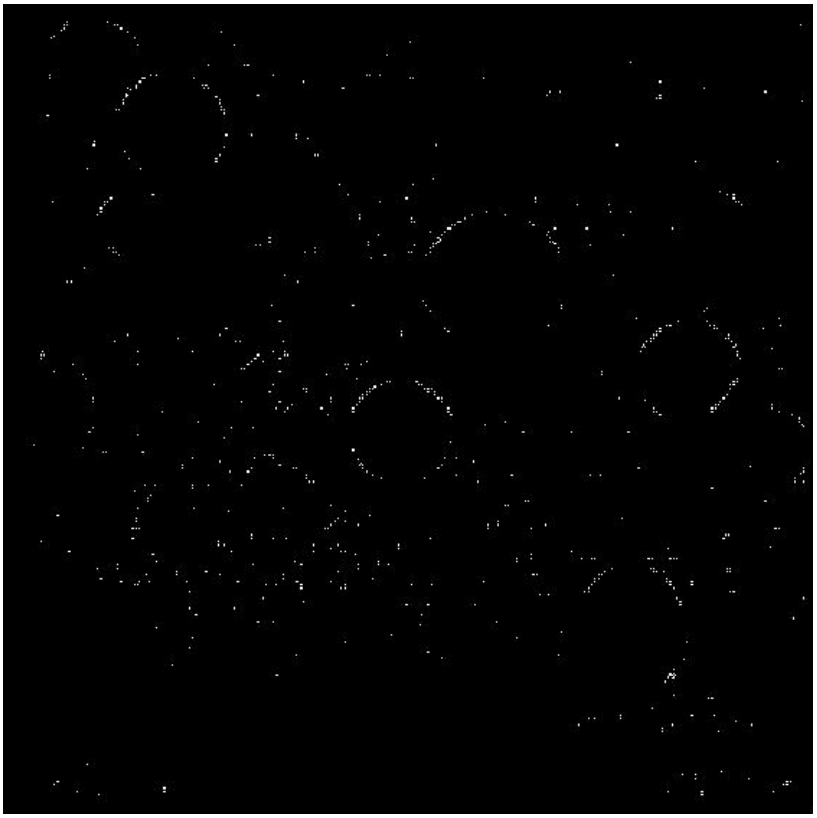


Figure: 5.6 Match points found at level 1

This implementation of the algorithm has been applied to get the disparity map. The input image and matches found are shown in preceding figures 5.7 and 5.8 and 5.9. To calculate disparity map one customized algorithm has been implemented and disparity map calculated from those interest points are used as input and disparity map is calculated. This is shown in figure 5.10 and ground truth disparity map is shown in figure 5.11. As we can see in the figure 5.9 the disparity map shows most of the objects well distinguished. The correlation window size is 32 and Dg is 10.

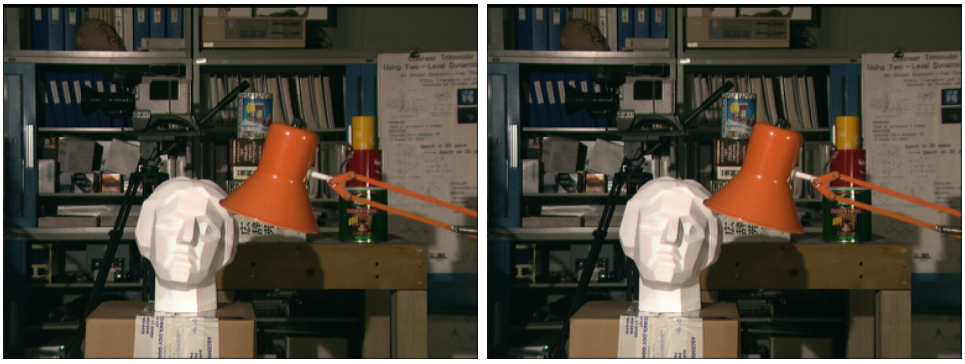
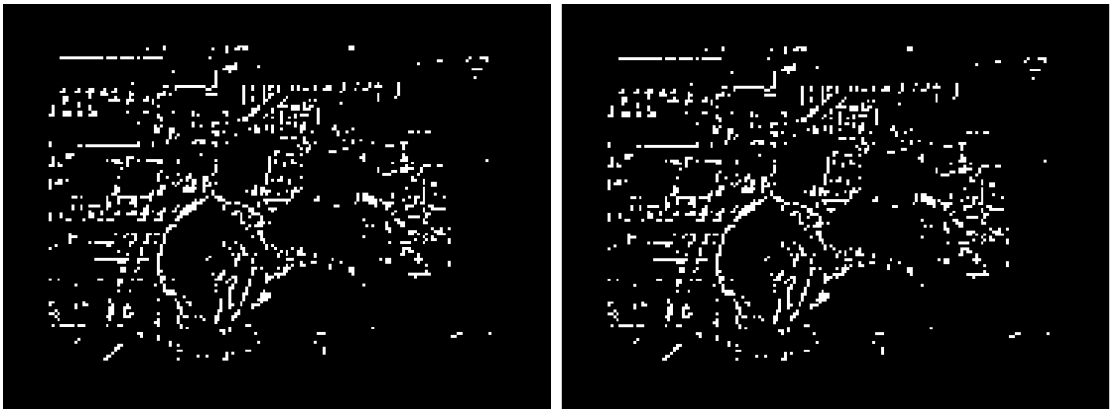


Figure: 5.7 Input stereo image pair[14]



'A' 'B'

Figure: 5.8 –'A' Interest points –'B' matches found window-size:32

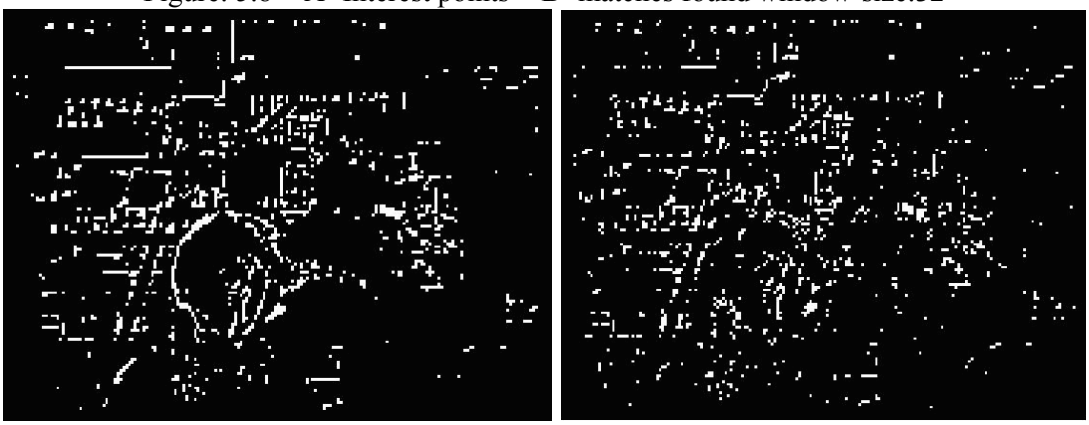


Figure: 5.9 match for window-size:8

Figure: 5.10 disparity map from customized algo.

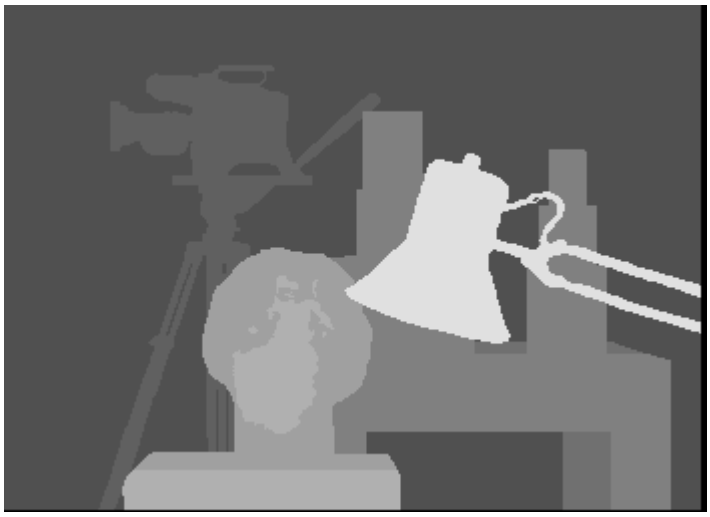


Figure: 5.11 ground truth disparity[14]

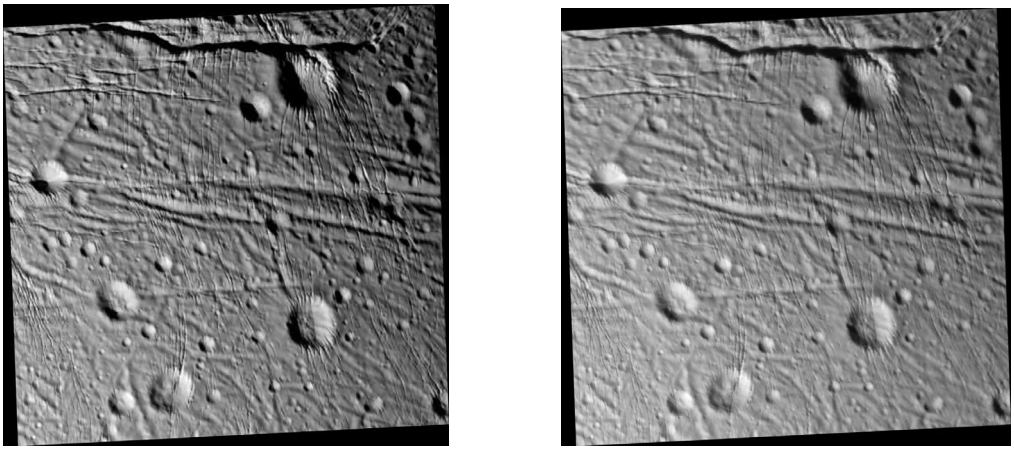


Figure: 5.12 Input stereo image pair

5.3 Analysis

In above section, the prototype has been implemented and their output is shown. Now it is required to implement this stereo matching

algorithm on FPGA. The major reason to use FPGA is to achieve the speed in execution of the algorithm. Here some parameters such as the correlation window size and ground operator threshold affect the output of the algorithm. For that reason to find the optimal correlation window size that requires less amount of calculations and feasible matching results, for various images and for various window sizes the output images are obtained and they are compared.

The next thing considered here is the correlation formula. As we know the correlation formula has calculation in denominator that involves the square root calculation. The implementation of the square root function on FAPA is not simple as we can do on using any language. The calculation of the square root function on the FPGA takes total of 102 cycles. So the idea is to remove square and square root from the correlation function. So the equation of the correlation now becomes as:

$$r = \frac{\sum_m \sum_n (A_{mn} - \overline{A})(B_{mn} - \overline{B})}{\left(\sum_m \sum_n (A_{mn} - \overline{A})\right)\left(\sum_m \sum_n (B_{mn} - \overline{B})\right)}$$

For various images this formula has been applied as correlation cost function and its results are compared with the original correlation formula. The results are incredible the match points obtained with use of the both of the formulas differs by only few numbers.

The results are shown in tables below. The second row shows the number of pixels differing while applying the small correlation window and comparing with the results obtained with window size of the 32. The third row shows the same quantity in percentages. While the forth row shown in table indicates no. of pixels differ while applying the new correlation formula and comparing it with the original normalized cross correlation formula. Here in this case the window sizes are same and only difference is that the formula differs. Fifth row indicates same quantity in percentages. Now the sixth row shows no. of pixels differ while applying both the variations the small correlation window and new correlation formula. The last row shows the same quantity in percentage.

The table 5.1 shows the results for the variation shown above

applied to the same image as shown in figure: 5.1. The table 5.2 shows the results for the variation shown above applied to the same image as shown in figure: 5.7. The table 5.3 shows the results for the variation shown above applied to the same image as shown in figure: 5.12.

Window Size→	25	21	17	13	9	7
#Pixels differ compared with W size 32 (a).	204	267	355	439	539	570
Pixels differ for variation (a) in %	4.64	6.07	8.07	9.98	12.28	12.96
#Pixels differ for new corr. Formula (b).	136	64	59	72	130	224
Pixels differ for variation (b) in %	2.6	1.24	1.13	1.36	2.4	4.2
Variation (a) & (b).	300	407	545	726	911	967
Variation (a) & (b) %	6.8	9.25	12.3	16.5	20.72	21.99

Table: 5.1

Window Size→	25	21	17	13	9	7
#Pixels differ compared with W size 32 (a).	27	45	61	70	91	107
Pixels differ for variation (a) in %	4.8	8.0	10	12.5	16	19.14
#Pixels differ for new corr. Formula(b).	38	41	34	46	113	157
Pixels differ for variation (b) in %	6.51	7.0	5.83	7.7	18.9	28.0
Variation (a) & (b).	49	68	75	87	143	197
Variation (a) & (b) in %	8.7	12	13.4	5.56	25	32.54

Table: 5.2

Window Size→	25	21	17	13	9	7
#Pixels differ compared with W size 32 (a).	304	369	412	524	791	1207
Pixels differ for variation (a) in %	4.16	4.84	5.41	6.8	10.3	15.8
#Pixels differ for new corr. Formula(b).	389	545	684	1057	2067	3576
Pixels differ for variation (b) in %	4.95	6.8	8.5	13.1	25.7	44
Variation (a) & (b).	292	290	354	443	540	633
Variation (a) & (b) in %	5.91	5.87	7.17	8.97	10.94	12.8

Table: 5.3

Now it could be realized form the data from the tables above that if the variation (a) is applied then the change in pixels match is quite less up

to the window size of 21. Therefore it is feasible to go for the smaller window size.

Now if the variation (b) means new correlation formula, is applied then the matches obtained that differ are quite less up to the window size 17. Therefore it is feasible to go for new correlation formula for this window size and results are acceptable. If both the variations are applied then the amount of mismatch obtained are quite less up to some window size. Now here every thing depends upon the type of the application and the acceptable tolerance limit. If the tolerance limit is up to 4%. Then certainly the correlation window size of 25 and new correlation formula is feasible. Only window size changed from the 33 to 25 reduce half of the calculation for the matching. If this algorithm is implemented on FPGA then if we can remove the square root function and apply new correlation formula then it can save many clock cycles of the calculations. Hence good speed up could be achieved.

One important thing in this result is that when we see the difference of the pixels matched is less than 3 pixels for most of the differing pixels. Therefore when match point at the higher level is mapped to the lower level and then at lower level it searches again within the search space surrounding that pixel. Now if the pixel difference is less than 3 then at the lower level the search space remains the same and it will not affect the final outcome at the lowest level result.

Hence, A new approach could be applied here and that is for all the higher levels, one could apply the new correlation based formula and smaller window. In the lowest level with highest resolution one should apply the reasonably large correlation window and original normalized cross correlation formula. This should outcome in good efficiency (less computation) without sacrificing accuracy.

Above mentioned procedure has been applied for different window sizes 33, 25, 21 for two images above used and another one image shown in figure: 5.13. Here results are obtained for only one variation of new correlation formula. From total of 3 levels for higher two levels new

formula applied and for the last level original correlation formula has been applied ('n'). The matches obtained are compared with the matches obtained by applying original formula for all the levels ('o'). The results are shown in table: 5.4. These results show that it fulfils all requirements with good accuracy and fewer computations.

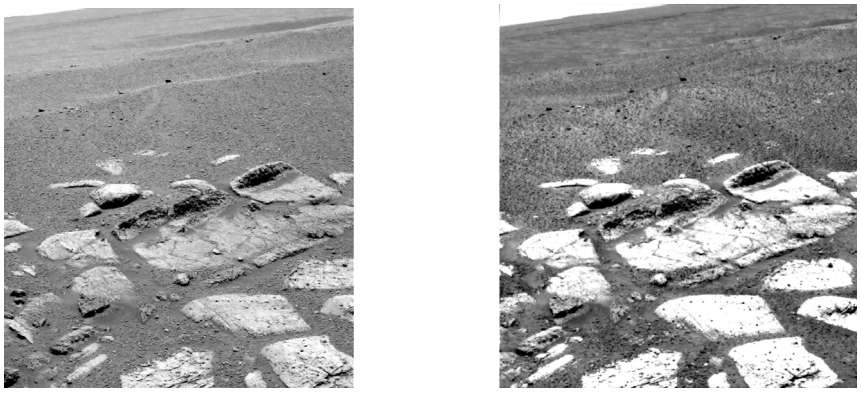


Figure: 5.13 Input stereo Image pair

Window Size->	33	25	21
#Pixels differ in ('n') compared with ('O') for image fig: 5.1 in %	0.009	0.000	0.0005
#Pixels differ in ('n') compared with ('O') for image fig: 5.12 in %	0.0002	0.0004	0.0005
#Pixels differ in ('n') compared with ('O') for image fig: 5.13 in %	0.025	0.021	0.012

Table: 5.4

One other parameter is the no. of levels applied for hierarchy. This certainly depends upon the resolution of the input image pair. More the resolution of the image more the no of levels preferred.

Again the threshold used in finding the ground operator points depends upon the input image characteristics and the application.

6. CONCLUSION

As the aim of the dissertation is analysis of the stereo image matching algorithm and parameterization for implementation on FPGA, hierarchical image matching algorithm has been studied and implemented. For implementation on FPGA various improvements are suggested. By applying improvements (small window size & new correlation formula) the effect on the final outcome is obtained and analyzed.

Putting together the experimental results from the investigation and the discussion, it could be realized that as the window size decreases, the accuracy of the match also decreases. The affect of the new correlation formula also increases with smaller windows. After some limit accuracy decreases tremendously (here it is 8). Hence, up to some limit the window should be decreased and that depends on the resolution of input image and tolerance of the inaccuracy for particular application.

One good solution to this problem is that go for the optimal (small window size and new correlation formula) implementation at the higher levels of the image pyramid and apply the calculation much faster and only at the last level go for the reasonable large correlation window size and original normalized cross correlation formula. It will result in exact matches and good efficiency (fewer calculations) at the same time without compromising the accuracy.

REFERENCES

Books:

[1] Richard Hartley. "Multiple View Geometry in Computer Vision". University of Oxford, UK.

[2] R. Gonzales, P. Wintz, Woods, "Digital Image Processing", Addison-Wesley, 1987.

[3] Ian H. Witten, Eibe Frank, "The Geometry from Multiple Images", MIT Press, 2000.

Research Papers:

[4] Stephen t. Barnard and martin a. Fischler. "Computational Stereo". SRI International, Menlo Park, California 94025, ACM Computing Surveys (CSUR), Volume 14.

[5] Naveed Bin Rais. Hammad A. Khan, Dr. Farrukh Kamran. Dr. Habibullah Jamal. ".A new algorithm of Stereo matching using Epipolar
Geometry". International Journal of Computer Vision.

[6] C. H. Lo, A. Chalmers "Stereo Vision for Computer Graphics: The effect that Stereo Vision has on Human Judgments of Visual Realism" ,Proceedings of the 19th spring conference on Computer graphics SCCG '03 ,ACM Press.

[7] Lisa Brown," A survey of Image Registration Techniques", ACM Computing Surveys, Vol. 24.No. 4, December 1992.

[8] Qi Tian and M N Huns," Algorithm for Subpixel Registration", Computer Vision, Graphics and Image Processing, 35,220-233(1985).

[9] Emmanouil Z. Psarakis and Georgios D. Evangelidis" An Enhanced Correlation-Based Method for Stereo Correspondence with Sub-Pixel Accuracy" Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1.

[10] R. W. Hartenstein and M. Z. Servit," Field-Programmable Logic Architectures, Synthesis, and Applications", IEEE Symposium on Field-Programmable Custom Computing Machines, pages 156– 167. Springer-Verlag, Berlin, 1994.

[11] B. GopalaKrishna," Conjugate Point Identification and Image Matching Methods". Lecture No. 9, Tutorial on Satellite Photogrammetry and Surveying, SAC ISRO, Ahmedabad During 11-14 December 2000.

[12] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two frame stereo correspondence algorithms", International Journal of Computer Vision, vol. 47, no 1, pp-7-42, April 2002.

[13] C. Loop and Z. Zhang," Computing Rectifying Homographies for Stereo Vision." IEEE Conf. Computer Vision and Pattern Recognition, 1999.

Websites:

[14] www.middleburry.edu

[15] www.samadams.at.northwestern.edu/stereoimage.