3D RECONSTRUCTION FROM 2D DATA

By
NC Saba Ahmed
NC Muneeba Raja
NC Fizza Shams
NC Omar Ikram

Project Supervisor
Col. Naveed Sarfraz Khattak

Submitted to the Faculty of Computer Science
National University of Sciences and Technology, Rawalpindi in partial fulfillment
for the requirements of a B.E Degree In Computer Software Engineering
AUGUST 2009

CERTIFICATE

Certified that the contents and form of project report entitled **"3D Reconstruction from 2D UAV Data"** submitted by 1) NC Saba Ahmed 2) NC Muneeba Raja 3) NC Fizza Shams, 4) Omar Ikram have been found satisfactory for the requirement of the degree.

Supervisor: _____
Col. Naveed Sarfaraz Khattak

ABSTRACT

3D RECONSTRUCTION FROM 2D UAV DATA

Stereo vision refers to the ability to infer information on the 3-D structure of a scene from two or more images taken from different viewpoints. Depth information is lost in the initial phase when image is taken. Stereo pairs stand as an imperative source for depth extraction. Processing a pair involves certain steps and techniques exist for each one. But there is not a completely defined approach, which encompasses these steps directing to depth extraction from a stereo pair. This report describes a system, which automatically recovers the depth information from images frames. After testing a number of stereo pairs, the experimental results demonstrate that our proposed approach leads to a system encompassing the state-of-art algorithms, which extracts the relative depth information from a stereo pair. Our system finds a number of applications in 3D vision, Robots systems, photogrammetry, traffic analysis and various other applications.

COPYRIGHT

DEDICATION

Dedicated to our parents who have been constant sources of encouragement for us and to our teachers who had full confidence in us and provided help during the project.

ACKNOWLEDGEMENTS

Humblest gratitude to Allah Almighty – the All-Knowing and the All-Powerful, the Creator, the Most Beneficent, Most Merciful. He is the Omni-Present and the Omni-Potent. Indeed, the working of the universe is nothing but a manifestation of the Great Powers He Possesses. Without His consent, not even a single breath could enter or leave our bodies, let alone the undertaking of work on this project. May He bestow us with His Guidance and make things clear for us when they get vague and confusing (Ameen).

We would like to thank our parents, who understood our concerns and spared us household chores, which would have definitely hindered the regular attention required for this work.

Our Supervisor, **Col. Naveed Sarfaraz Khattak**, is one of the person who requires special mention. The interest and eagerness exhibited by him to assist work in the development of complete system has been phenomenal, which has been complemented by his excellent management and organizational skills. Also his patient, consistent and professional guidance helped us in achieving our project goals. We indeed are thankful to our Co-Supervisor, **Maj. Dr. Naveed Iqbal Rao,** for showing his confidence in us helping us during the project work. Finally, thanks to Military College of Signals, (NUST) for providing us the opportunity to enhance our technical and practical skills.

TABLE OF CONTENTS

CHAPTER 1
OVERVIEW

LIST OF FIGURE

Future Work and Enhancement

The work on this project has been done up till the part of constructing a 3D model from 2D images. Our images are taken from a UAV. Further work can be done in enhancing the 3D structure by extracting exact depth of the objects. The depth information is lost when we take images of the 3D world. Therefore, our 3D model can be converted into DEMs (Digital Elevation Model) by calculating accurate or with minimum amount of error in the height. This procedure can be achieved by slight addition to the ortho-rectification. By calculating the exact height, we are not only able to visualize the objects in the images in 3D space but also look at its height which will give a better perspective of understanding how the objects stand with one another in the real world. Such work will have a great importance in the military where understanding of the topology of earth surface is very important in planning tactics and strategy.

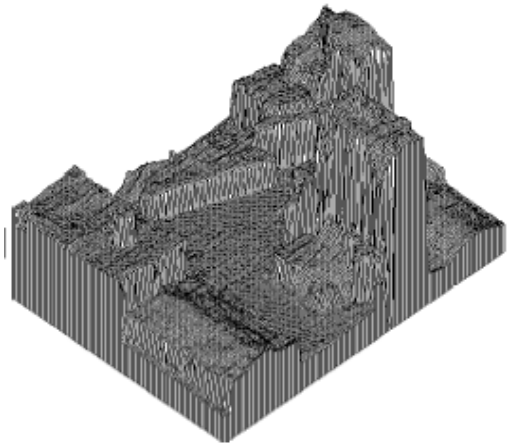
CHAPTER 1

OVERVIEW

1.1 3D Reconstruction

The area of Robotics has found an intensive research effort over the last decade. The endeavor by the researchers to make robots intelligent machines has started giving dividends and a need of giving vision to these machines stands as a prerequisite. Instead of making a robot move on a specified path and perform predefined actions, can there be a possibility that it detects the obstacles and explores its own path during maneuver. It led to the evolution of a field called as Computer or Machine Vision.

Computer Vision is not only restricted to robots and machines but finds immense applications in other areas as modeling scenes for virtual reality applications, either in the areas of business (real estate, architecture,



information-dispensing) , education (electronic museums and multimedia books), or entertainment (interactive 3-D games, movies) and above all the military applications (not only in extracting the true information of enemy deployment but also useful in target recognition ,detection and tracking). The option of creating virtual environments by capturing real scenes through video cameras is getting particular attention, given the labor-intensive and thus expensive nature of creating models by hand using a 3-D geometric modeler. Any given view of the camera or a depth imaging such as a light-stripe rangefinder is insufficient for creating models of a large scene or an entire object; thus merging of multiple views taken at different locations is usually necessary. Integrating the different views to result in a seamless 3-D model then follows this. Depth information, which is extracted from stereo imagery using different techniques of Computer Vision, not only enables to get topographic maps of large areas but also is significant in getting intelligence of hostile terrain and enemy deployment. The process used is called as epipolar geometry alternatively known as stereovision. Stereovision does not remain restricted to 3D reconstruction but its roots extend much beyond this. Figure 1.1 shows 3D model of scene.

1.2 What is Machine Vision?

The goal of a machine vision system is to create a model of the real world from images. A machine vision system recovers useful information about a scene from its two-dimensional projections. Since images are two-dimensional projections of three-dimensional world the information is not directly available and must be recovered. Recovery requires the inversion of a many-to-one mapping. To recover the information, knowledge about the objects in the scene and the projection geometry is required.

The information recovered by a vision system is different in different cases. If diagnosis of a disease using computed tomography images is desired, then some techniques of machine vision can be applied. Quantitative measurements on regions of interest can also be made easily available. Machine vision systems help a physician to recover information by enhancing the images. Such systems have been used for quality control of products ranging from pizza to turbine blades, from submicron structures on wafers to auto-body panels, and from apples to oranges. The information obtained from two pairs of images known as stereo images acquired by a mobile robot or a flying unmanned aircraft or a satellite is combined to get a robust map of environment at a resolution which is sufficient for the task. Such information is useful in autonomous navigation of automobiles airplanes, tanks, and robots. Machine vision systems are playing an increasingly important role in analysis and information management of the exceedingly large volume of data collected by satellites.

1.2.1 Relationship to Other Fields

1.2.1.1 Machine Vision and Image Processing. Image processing is a well-developed field. Image processing techniques usually transform images into other images; the task of information recovery is left to a human user. This field includes topics such as Image enhancement, image compression and correcting blurred and out of focus images. Machine vision algorithms take images as inputs but produce other types of outputs, such as representations for the object contours in an image. Emphasis in machine vision is on recovering information automatically, with minimal interaction with a human. Image processing algorithms are useful in early stages of a machine vision system. They are usually used to enhance particular information and suppress noise.

1.2.1.2 Computer Graphics and Machine Vision. Computer graphics generates images from geometric primitives such as lines, circles, and free form surfaces. Computer graphics techniques play a significant role in visualization and virtual reality. Machine vision is the inverse problem: estimating the geometric primitives and other features from the image. Thus computer graphics is the synthesis of images and machine vision is the analysis of images. These two fields have been growing closer. Machine vision is using curve and surface representations and several other techniques from computer graphics and computer graphics is using many techniques from machine vision to enter

models into the computer for creating realistic images. Visualization and virtual reality are bringing these two fields closer.

Vision = Geometry + Measurements + Interpretation

1.2.1.3 Artificial Intelligence and Machine Vision. Artificial intelligence is concerned with designing systems that are intelligent. Artificial intelligence is used to analyze scenes by computing a symbolic representation of the scene contents after the images have been processed to obtain features. Artificial intelligence may be viewed as having three stages: Perception cognition and action. Perception translates signals from the world into symbols, cognition manipulates symbols and action translates symbols into signals that effect changes in the world. Computer vision is often considered as sub field of artificial intelligence. Neural networks are being increasingly applied to solve some machine vision problems.

1.2.1.4 Human Vision and Machine Vision. Many techniques in machine vision are related to what is known about human vision Many researchers in computer vision are more interested in preparing computational models of human vision than in designing machine vision systems [1]. Unlike the cameras rigidly attached to a passive stereo rig, the two eyes of a person can rotate in their socket. At each instant, they fixate on a particular point in space. Figure 1.2 explains a simplified two-dimensional situation. If l and r denote the angle between the vertical planes of symmetry of two eyes and two rays passing through the same scene point, the corresponding disparity is defined as difference of r and l. It is an elementary exercise in trigonometry to show that $d = D - F$ where D denotes the angle between these rays, and f is the angle between the two rays passing through the fixated point. Points with zero disparity lie on the Vieth- Muller circle that passes through the fixated point and the interior nodal points in the eye. Points inside the circle have positive disparity and points outside it have negative disparity. The three dimensional case is of course more complicated, the locus of zero disparity points becoming a surface , the horopter, but the general conclusion is same and absolute positioning requires the vergence angles.

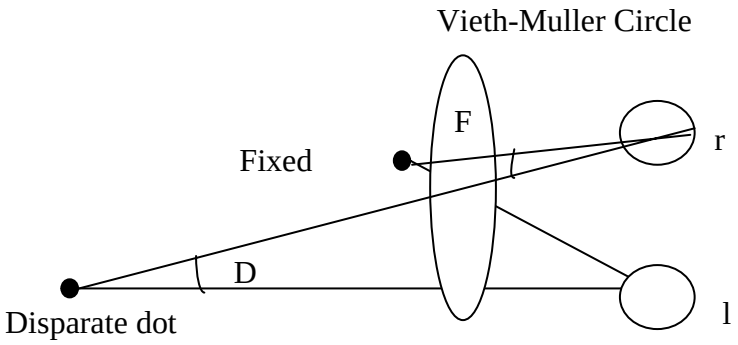
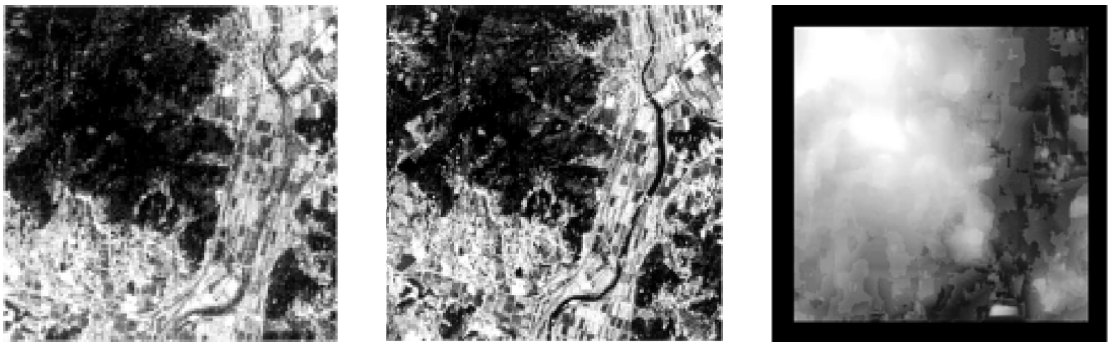


Figure 1.2 Human Vision System

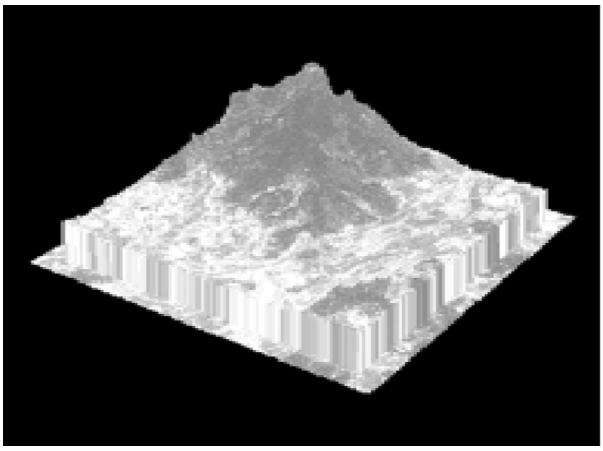
1.3 Applications

1.3.1 Digital Elevation Models (DEMs)

Satellite stereo pairs are processed to extract elevation information of the terrain whose digital representation is called as DEM. Figure 1.3 shows representation.



Left Image **Right Image** **DEM**



3D Model of DEM

Figure 1.3 Creation of DEM from Satellite Imageries

1.3.2 Robot Vision

Again the stereo pairs taken by 2 cameras focused at the same location give the depth information thus giving vision to the robots (Figure 1.4).

Figure 1.4 Robot vision

1.3.3 Traffic Management

Tasks like Traffic scene, Number of vehicles, Type of vehicles, Location of closest obstacle and Assessment of congestion can be performed using stereovision (Figure 1.5)

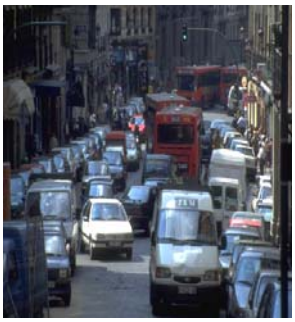


Figure 1.5 Traffic management

1.3.4 Pilotless Vehicles

To give vision to the vehicles and make them drive without drivers. The same concept is being used in military research for Remotely Piloted Vehicles (Figure 1.6).

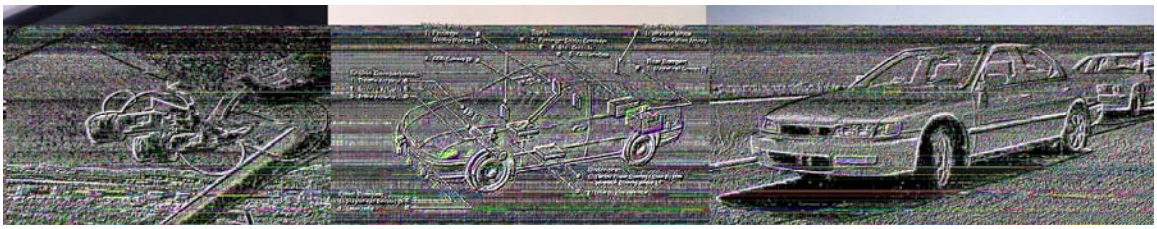


Figure 1.6 Pilotless Vehicles. A project being undertaken in Germany

1.3.5 Visually Guided Mobile Robots

Though through use of laser and infrared rays, guided robots exist but the technique involves emission of radiation, which may lead to their detection in case of military applications and also does not give better results. Also such robots were limited to detect the obstacle in front but were not able to get the picture of scene in front. The modern robots equipped with stereovision mechanism are able to get the depiction of scene in front and are guided more accurately.

1.3.6 Multi Camera Multi Person Tracking

A stereo system can also be used for tracking and so can also extend to target recognition and detection (Figure 1.7).

Figure 1.7 Person tracking Modules

1.4 Stereopsis

The image in the human retina is a projection of the three-dimensional world onto a two-dimensional surface, the information on the third dimension,

depth, is already lost at the very first stage of vision. However, fusing the images perceived by our two eyes and exploiting the difference between them allows us to obtain a sense of depth, which in the human visual system is called stereo vision. Stereo vision refers to the ability to infer information on the 3-D structure and distance of a scene from two or more images taken from different viewpoints. For computers, two views of a scene are analogous to the two eyes in the human visual system. By having two cameras displaced from each other, knowing the camera focal lengths and using epipolar geometry, the depth of objects in an imaged scene can be estimated.

In stereopsis, two images are taken of the same scene from slightly different viewpoints. Those objects in the scene that are far away from the two cameras will appear nearly identical in the two images, whereas those objects that are near the cameras will change significantly between the two images. The key idea underlying stereopsis (or stereo vision) is to make use of the *disparity*, or change in image location, of an object from one view to the next. The closer an object is to the camera, the larger the disparity will be. This allows us to reconstruct three-dimensional shape from disparity. Generally the depth (or distance) information recovered from stereo is quite noisy, and thus a surface interpolation process is often applied to the data. Such interpolation methods have broad applicability beyond computer vision.

A classic stereo pair is a narrow baseline stereo with two cameras shortly displaced from each other while wide baseline stereo involves the cameras largely displaced and resultantly the images have a lot of occluded regions. The depth to a physical point can be computed by triangulation if projections of the point in two images are known (reconstruction problem). The idea underlying stereopsis is to determine a correspondence (or matching) between each location *of* one image and some location of second image. In other words, to find the pairs of points that results from the projection of the same point *M* into the two images. Note that a given point need not have an image in both the images there may be some other point in the scene that hides point from view in the first

or second image, causing there to be no correspondence and such points are said to be occluded.

If an object is infinitely far away, then its projection into the two camera planes will be at the same location, and the disparity will be zero. If an object is close to the cameras then the disparity will be large. In other words, disparity is inversely proportional to the distance between an object and the camera system. Stereopsis is a common technique for recovering shape both in artificial vision systems and in the human visual system. It is not indispensable, however, as a significant percentage of people have little or no stereo vision.

Establishing the matching image coordinates is the fundamental problem in stereo vision. If knowledge about the camera geometry and relative viewpoints is available, a powerful geometric constraint, known as the epipolar constraint, reduces the search space for possible matches from two dimensions (the entire image plane) to one dimension (the epipolar curve). The basis of stereo vision is finding the parallax in the stereo pair. The concept of parallax can be best understood by the following example [2] .Hold hand in front of face and turn head right and left without moving hand. Then observe how the background and hand are shifted relative to each other (Figure 1.8). This is called parallax error

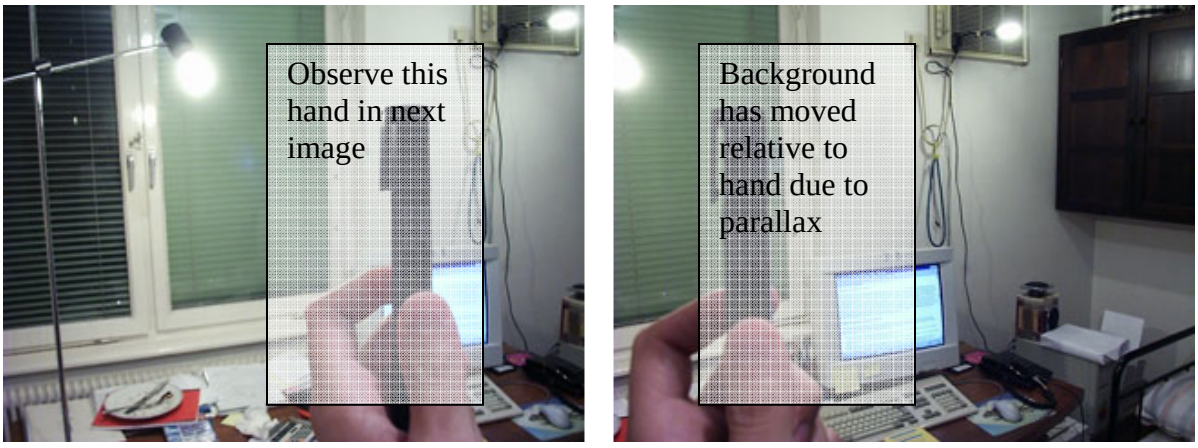


Figure1.8 Head moving towards right showing the Parallax

Now imagine this is recorded by camera. First turn our head to the left and imagine a picture is taken, turn head to the right and imagine taking one more picture. Hand should be in the overlapping area. As anyone will see, the overlapping area has a different content and there is no way to stitch these images together. Things that are nearer have more parallax while things those are far have less parallax. In figure 1.5 observe that the relative movement of window is much more then the relative movement of background window.

1.5 Basic Terminologies

1.5.1 Rank of Matrix

The rank of a matrix is the no of linearly independent rows/columns in the matrix.

1.5.2 Linear Independence.

Let v as vectors and c as scalars so then

$$c_1 v^1 + c_2 v^2 + \ldots\ldots\ldots + c_n v^n = 0$$

It implies that all scalars must be zero only then the above condition can be satisfied.

1.5.3 Eigen Values and Eigen Vectors

Let A=[a_{jk}] be then given matrix. Consider the equation

$$Ax=\lambda x$$

Where λ is the scalar (real or exponential). To be determined and x is the vector (can be a row or column vector and can't be a matrix) to be determined. For every λ one solution is x=0.A scalar λ such that the above equation holds for some vector x ~ $\tilde{\lambda}$ is called an eigen value of A, and this vector is called an eigen vector of A corresponding to this eigenvalue of λ.

$$Ax - \lambda . x = 0$$

$$\begin{pmatrix} a_{11} - \lambda & a_{12} \\ a_{21} & a_{22} - \lambda \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Now if A is 9x9 matrix then x is a 9x1 vector so Ax is 9x1 vector.

1.5.4 Singular Matrix

If A has no inverse then A is called the singular matrix.

1.5.5 Spectrum of A

The set of the eigen values of A is called the spectrum of A.

1.5.6 Eigenspace or Vector Space

The set of all eigenvectors corresponding to an eigenvalue of A together with zero is called the eigenspace of A.

1.5.7 Null Space

A homogeneous linear system

$$\begin{aligned} &a_{11}x_1 + a_{12}x_2 + \cdots\cdots + a_{1n}x_n = 0 \\ &a_{21}x_1 + a_{22}x_2 + \cdots\cdots + a_{2n}x_n = 0 \\ &\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ &a_{m1}x_1 + a_{m2}x_2 + \cdots\cdots + a_{mn}x_n = 0 \end{aligned}$$

always has the trivial solution $x_1=0,\ldots\ldots..,x_n=0$. Nontrivial solutions exist only if rank A<n. If rank A =r<n, these solutions together with x=0 form a vector space of dimension n-r. x solution means that a solution containing all the values of $x_1,x_2,\ldots\ldots x_n$. Now if there are 2 other solutions i.e $x_{(1)}$ and $x_{(2)}$ i.e 2 other sets of $x_1,x_2,\ldots.x_n$ then

$$x=c_1x_{(1)}+c_2x_{(2)}$$

where c_1 and c_2 are scalars.

It is mentioned that vector space of all solutions of above equations is called the Null Space of the coefficient matrix A,

$$A=\begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix}$$

Because if any x in this null space of A is multiplied result is 0 as the solutions that are satisfying the above equations are part of the vector space and these all solutions are based on the answer to be zero.

1.5.8 Symetric and Skew Symmetric Matrix

If $A=A^T$ then the matrix is symmetric. A matrix is skew symmetric when $A^T= -A$

A matrix is orthogonal if $A^T=A^{-1}$.

1.5.9 Length or Norm of A Vector

$$\| a \| = \sqrt{a.a} = \sqrt{| a_1 |^2 + \cdots | a_n |^2}$$

1.5.10 Computational Stereo

It refers to the problem of determining 3-dimensional structure of a scene from two or more images taken from distinct viewpoints. The fundamental basis for stereo is the fact that a single three-dimensional physical location projects to a unique pair of image locations in two observing cameras

1.5.11 Calibration

It is the process of determining camera system external geometry (the relative positions and orientations of each camera) and internal geometry (focal lengths, optical centers and lens distortions). Accurate estimates of this geometry are necessary in order to relate image information (expressed in pixels) to an external world coordinate system.

1.5.12 Occlusion

Disparities can only be computed for features visible in both images; features visible in one image but not the other are said to be occluded

1.5.13 Correspondence problem

The correspondence problem consists of determining the locations in each camera image that are the projection of the same physical point in space.

1.5.14 Reconstruction problem

The reconstruction problem consists of determining 3-dimensional structure from a disparity map, based on known camera geometry.

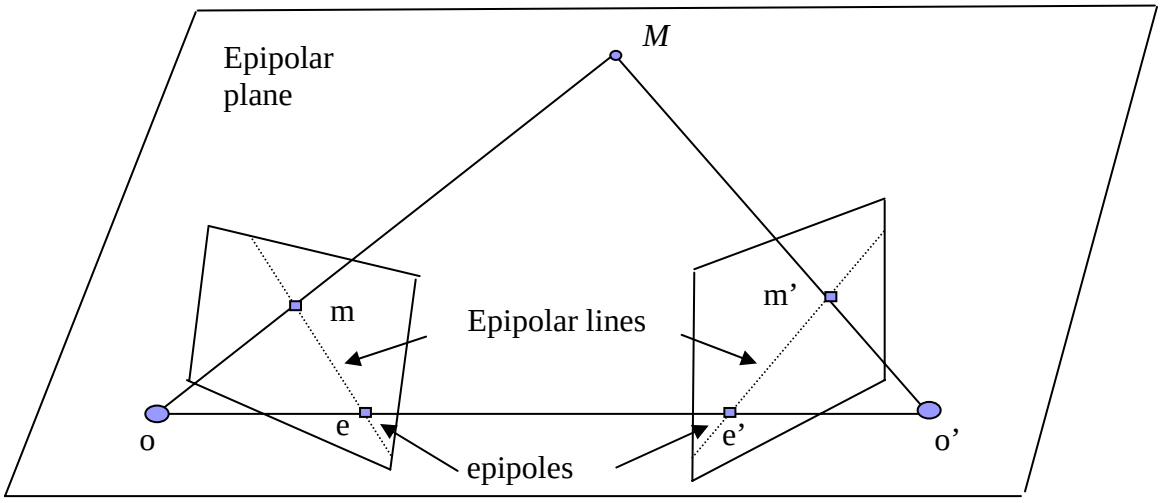


Figure 1.9 Epipolar Geometry

1.5.15 Epipolar Geometry

The epipolar geometry is the intrinsic projective geometry between two views. It is independent of scene structure, and only depends on the cameras' internal parameters and relative pose. The epipolar geometry between two views is essentially the geometry of the intersection of the image planes with the pencil of planes having the baseline as axis (the baseline is the line joining the camera centers, Figure 1.9). This geometry is usually motivated by considering the search for corresponding points in stereo matching [1].

1.6 Techniques Available

One of the goals of visual processing is to extract three-dimensional geometric information from one or more images. Extracting three-dimensional geometry from images is often referred to as *shape-from-x*, because there are a number of different sources of information that can be used to recover the three-dimensional structure of a scene (or shape) from two-dimensional images. For instance, shading in an image reveals information about three-dimensional shapes (e.g., much of the way that the shape of a sphere in a photograph is perceived as being a solid rather than a disk is due to the uniform change in brightness away from the light source). Shape-from-shading is also an active area of research. Some of the techniques available are Relief Displacement, Parallax Theorem, Stereo Imaging, Structure From Motion and X Techniques that includes Photometric stereo, Shape and Shading, Shape from Texture, Shape from Focus and Range Imaging.

1.6.1 Relief Displacement

In this technique the depth information is extracted from one photograph. The basis is the leaning of the elevated objects in the photograph.

1.6.2 Parallax Theorem

It is based on a very common observation that while traveling in the train and looking out of the window, the trees and other objects in the near vicinity pass quickly while mountains at the back pass very slowly. So in a sequence of image especially in a video mosaic this technique can be effectively used. In subsequent frames the objects that are displaced more are nearer to the camera position while the objects that are displaced less are comparatively away from the camera. Basing on this fact and the knowledge of camera's intrinsic and extrinsic parameters the elevation or depth information is extracted from the frames.

1.6.3 Shapes from X

Method described above and numerous other methods known as shape from X techniques have been developed for extracting shape information from intensity image.

Image of the same scene are obtained using light sources from three different directions

1.6.3.1 Shape from Shading

Shapes from shading methods exploit the changes in the image intensity (shadowing) to recover surface shape information. This is done by calculating the orientation of the scene surface corresponding to each point in the image.

1.6.3.2 Shape from Texture

Texture properties such as density, size and orientation are the cues exploited by shape from texture algorithms.

1.6.3.3 Shape from Focus

Due to finite depth of the field of optical systems only objects which are at a proper distance appear focused in the image whereas those at other depths are blurred in proportion to their distances.

1.6.3.4 Shape from Motion

When image of a stationary scene are acquired using a moving camera, the displacement of the image plane coordinate of a scene point from one frame to another depends on the distance of the scene point from the camera.

1.6.4 Stereo Imaging and Human Vision

The image in the human retina is a projection of the three-dimensional world onto a two-dimensional surface, the information on the third dimension, depth, is already lost at the very first stage of vision. However, fusing the images perceived by our two eyes and exploiting the difference between them allows us to obtain a sense of depth, which in the human visual system is called stereo vision. For computers, two views of a scene are analogous to the two eyes in the human visual system. Having two cameras

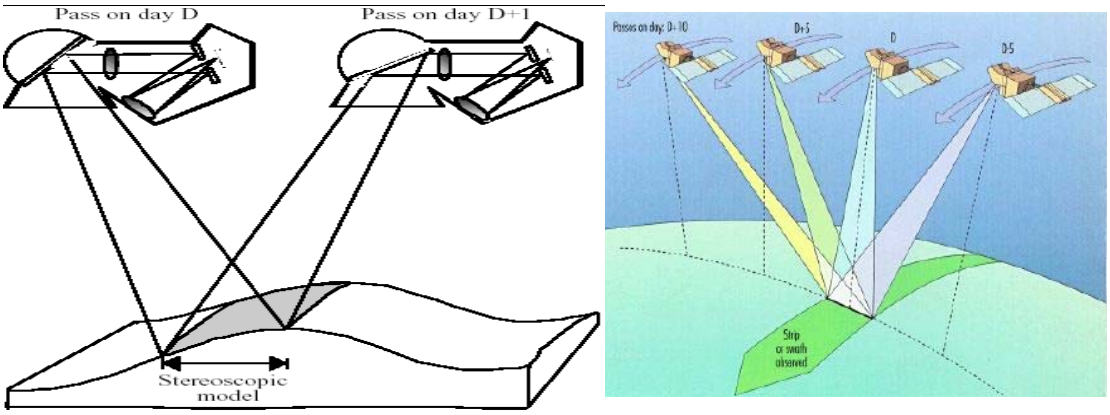


Figure1.10 SPOT Satellite taking Stereo Pairs

displaced from each other, knowing the camera focal lengths and using epipolar geometry can estimate the depth of objects in an imaged scene. The real trick to

autonomously estimate the depth information in a scene is to determine the correspondence between the two images and to then estimate the epipolar geometry, which is described by so-called fundamental matrix. Once epipolar geometry is determined, the 3D structure recovery problem is essentially solved.

In figure 1.10 the desired place is photographed from two different location with the camera positions known and then by epipolar geometry and conjugate pair of images, the depth information is extracted.

1.6.4.1 Stereoscopic Vision Using Spot satellites

The SPOT Satellite Earth Observation System was designed by the CNES (Centre National d'Etudes Spatiales), in France, and developed with the participation of Sweden and Belgium. The system comprises a series of spacecrafts plus ground facilities for satellite control and programming, image production and distribution. The first satellite SPOT 1 was launched on 22 February 1986, and the latest in the series SPOT 5 was launched in May 2002. There are currently three operational satellites. SPOT's unique features - high resolution, stereo imaging and revisit capability - enable it to acquire data from areas of special interest for various applications (cartography, agriculture, environment, land use, landcover, etc...).

1.6.4.2 Epipolar Geometry

The application of projective geometry techniques in computer vision is most notable in the *Stereo Vision*. The epipolar geometry between two views is essentially the geometry of the intersection of the image planes with the pencil of planes having the baseline as axis (the baseline is the line joining the camera centres). Considering the search for corresponding points in stereo matching usually motivates this geometry.

1.6.5 Structure from motion

An important approach is structure from motion. Here the idea is to take a sequence of images, and to use the motion of an object with respect to the camera to reconstruct its three-dimensional shape. There are many methods for solving this problem, but all of them involve first *tracking* the object (finding corresponding points in successive frames), and then applying some sort of technique to recover the three dimensional positions of the points from their two-dimensional motions. The tracking problem is itself quite difficult, particularly when it is necessary to identify corresponding points in successive frames. Many methods require the object to be stationary, and the camera to undergo a restricted type of motion. This helps with both the tracking problem, and the subsequent shape reconstruction. Stereo vision is however more structured of the other methods, because it is assumed that the cameras and the objects are fixed. It is also generally assumed that something about the relation between the two camera frames is known.

CHAPTER 2

STEREOPSIS

2.1 Introduction

In the basic stereovision paradigm, there are two cameras observing a static scene (i.e., where nothing is moving). The relative coordinate systems of the two cameras are known, or are constrained in some fashion. Various modifications include adding a third camera, and adding small motions of the cameras to help resolve possible ambiguities. In the basic two-camera case, the images are generally referred to as *I* and *I'* (resulting from the left and right cameras respectively). The idea underlying stereopsis is to determine a correspondence (or matching) between each location *m* of *I* and some location *m'* of *I'*.

In other words, to find the pairs of points *m* and *m'* that result from the projection of the same point *M* into the two images. Note that a given point *M* need not have an image in both *I* and *I'* — there may be some other point in the scene that hides *M* from view in the left or right image, causing there to be no correspondence.

The *disparity* or difference in image location, of *m* and *m'* then indicates the distance from the cameras to the point *M* in the world. If an object is infinitely far away, then its projection into the two camera planes will be at the same location, and the disparity will be zero. If an object is close to the cameras then the disparity will be large. In other words, disparity is inversely proportional to the distance between an object and the camera system. Stereopsis is a common technique for recovering shape both in artificial vision systems and in the human visual system. It is not indispensable, however, as a significant percentage of people have little or no stereovision.

To make the discussion more precise, the geometry of the camera system is considered. Use a simple pinhole-camera model for this purpose(Figure 2.1), where optical effects due to the lens are ignored completely. A simple camera thus consists of a focal point (or center), *o*, through which all the rays of light pass, and an image plane *I* onto which these rays are projected.

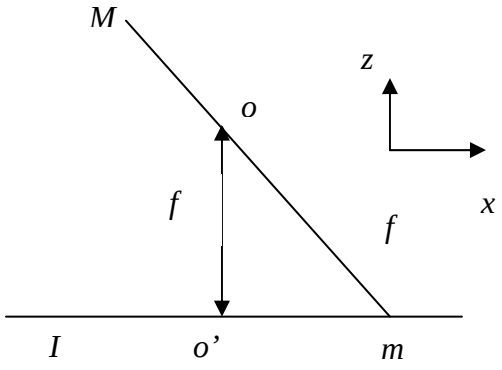


Figure 2.1: A pinhole camera model

The *optical axis* of the camera is the line perpendicular to the image plane, *I*, and through the focal point, *o*. *o' is the* intersection of the optical axis with the image plane. The distance from *o* to *o'* is called the *focal length*, *f*, of the camera. The situation is explained in figure 2.1. Place the origin of the world coordinate system at *o*, and the origin of the image plane at *o'*, and assume that the optical axis points in the ˆ*z* direction, then the following two equations describe the projection of a point at location (*x, y, z*) in the world into location (*x', y'*) in the image plane

$$\frac{x'}{f} = \frac{x}{z} \tag{2.1}$$

$$\frac{y'}{f} = \frac{y}{z} \tag{2.2}$$

These equations are referred to as the *perspective equations*. The projection of the world onto a plane through a central point in this manner is referred to as perspective projection (or central projection). In general it is assumed that the world origin is at *o*, the image origin is at *o'*, and the optical axis is in the ˆ*z* direction, and use the above equations.

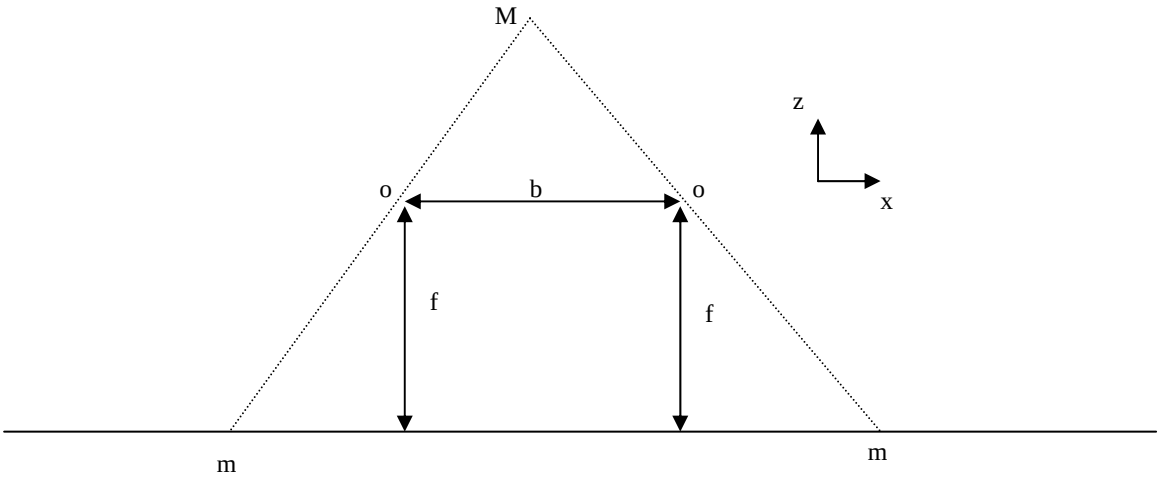


Figure 2.2: A simple stereo camera geometry of the two cameras

2.2 Stereo Derivation

For stereovision there are two cameras at some fixed relative position and orientation with respect to one another. First simple stereo camera geometry is considered in which the optical axes of the two cameras are parallel to one another, and are perpendicular to the *baseline* connecting the two camera centers (which are denoted by *o* and *o'*).

Moreover, assume that the focal length, *f* of the two cameras is the same. This situation is illustrated in Figure 2.2, where the length of the baseline (distance between the camera centers) is denoted by *b*. Let the origin of the coordinate system for the left image plane, *I*, be the projection of its optic axis, *o* (and similarly the origin of the right image plane, *I'*, is at *o'*). Let the origin of the world coordinate frame be along the baseline, at the point equidistant between the two camera centers (at distance *b*/2 from each *o* and *o'*). Note that this camera geometry makes *I* and *I'* the same plane, and with the same coordinate frames except for a translation of the origin in the *x*-direction.

Consider a point M = $(X,Y,Z)^T$ in the world which is imaged into *I* at location m= (x,*y*) and into *R* at location m' = (*x', y'*). By the perspective equations and the geometry

$$\frac{x}{f} = \frac{X + \frac{b}{2}}{Z} \tag{2.3}$$

$$\frac{x'}{f} = \frac{X - \frac{b}{2}}{Z}$$

$$\frac{y}{f} = \frac{y'}{f} = \frac{Y}{Z} \tag{2.4}$$

Where *b* is the baseline width and *f* is the focal length of both cameras. Thus in this simple camera geometry, only the *x* location of a projected point differs

between the left and right images. The *y* location of a given point in space is the same for both images. Recall that the disparity is defined as the distance between (*x, y*) and (*x', y'*), which in this case is just the magnitude of the difference between the *x* coordinates, $x - x'$..

From the above equations, it is observed that

$$\frac{x-x'}{f}=\frac{b}{Z} \tag{2.5}$$

Thus if *b* and *f* are known then the depth *Z* of the point *M* can be computed from the disparity. Note that as *Z* gets infinitely large the disparity goes to zero (things that are very far away have no disparity). If *b* and *f* are unknown then *relative* depths of points cab be computed, but not their absolute distance from the camera (because *b* and *f* although unknown are fixed, and thus there is simply a constant factor difference). In other words, for an uncalibrated camera system (unknown *f* and *b*) it is concluded is disparity is inversely proportional to depth.

The disparity is directly proportional to the focal length, *f*. Thus a larger focal length camera system will produce bigger disparities for the same distance, *z*. Disparity is also directly proportional to the baseline width, *b*. Note that if there is some fixed error in determining disparity, then increasing *b* and *f* will reduce the error in the depth computation (because increasing these quantities increases the amount of disparity for a fixed depth difference). The focal length, *f* is effectively limited for most cameras. The baseline is quite easy to increase, however this results in other problems. As *b* is increased, the two images become less and less similar to one another (in the worst case two finite size images contain nothing in common). Even when the images contain common subparts, the large disparities make it quite difficult to identify corresponding points *m* and *m'* in the two images that both result from the same point *M* in the world (because the points may be very far apart). This is a *fundamental tradeoff* in stereo imaging systems: a wider baseline provides more accurate depth

estimates for fixed errors in disparity; however it also makes the problem of determining a correspondence much more difficult.

A point *M = (X, Y, Z)* in the world and the two camera centers define a plane called the *epipolar plane* (alternatively this plane is defined by *M* and its two images *m = (x, y)* and *m' = (x, y)*. In other words, for each point in space there is a corresponding epipolar plane defined by that point and the two camera centers (or the two images in the stereo camera system). A given epipolar plane intersects with the left camera plane, *I*, defining an epipolar line. Analogously the intersection with *I'* defines an epipolar line. These two lines, one in each image, are referred to as a corresponding pair of epipolar lines. Note that in the simple camera model illustrated in Figure-2.2, the epipolar lines are both parallel to *x*-axis, and have the same *y*-coordinate.

The epipolar lines are important in stereo vision because if the corresponding pairs of epipolar lines in *I* and *I'* are known, then this constrains the possible locations of corresponding pairs of points in the left and right images. If a point *m* lies on a given epipolar line, then the corresponding point *m'* must lie on the corresponding line in the right image (if it occurs at all in the bounded image region of the plane *I'*). For example, in the simple camera geometry *y = y',* and thus the corresponding epipolar lines of the left and right image are those lines with the same *y* coordinates. The central computational problem in stereo vision is to determine for each point *m* in the left image, what matching point *m'* in the right image corresponds to *m* (that is what pairs of points *m* and *m'* are projections of the same point *M*). Thus knowing the corresponding pairs of epipolar lines in the two images constrains the search for corresponding pairs of points to just a line, rather than the entire image plane. In the case of the simple camera geometry a given point in the left image *m = (x, y)* must have a match on the line *m' = (x', y')* (if there is any matching point at all in the right image).

In case of *general* camera geometry, the corresponding epipolar lines in *I'* and *I* form pencils of lines through the images of the other camera centers (a

pencil of lines in the plane is the set of all lines through some given point). That is, all the epipolar lines in *I* go through projection of right camera center into that image (the image of *o'* in *I'*), and analogously all the epipolar lines in *I'* go through the projection of the right camera center into that image. In the case of the simple camera geometry, the right camera center *o'* projects to infinity rather than into *I* (and analogously for *o* and *I'*). Thus the point that all the epipolar lines go through is at infinity — in other words the epipolar lines are parallel. (Note that the image of the left (right) camera center need not actually be visible in the right (left) image, because the image is just a finite portion of I' (I).)

The correspondence of epipolar lines in the left and right images must be discovered through some sort of calibration process that relates the coordinate systems of *I* and *I'*. Accurate calibration is a difficult and tedious process.

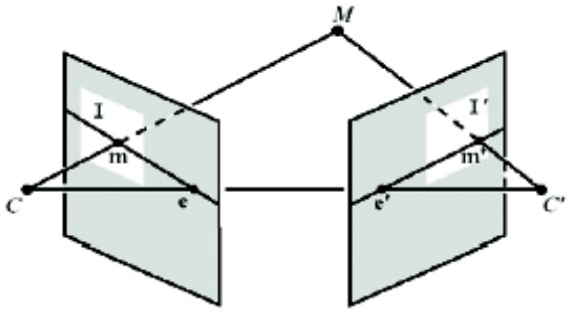


Figure 2.3 Epipolar Geometry

Generally cameras are used that are setup in (approximately) the simple geometry, where corresponding epipolar lines are lines parallel to the *x*-axis with the same *y* coordinate.

2.3 Epipolar Geometry

The only geometrical constraint that exists in a stereo pairs is called epipolar geometry (Figure 2.3). A stereo system can compute a great deal of 3-D information without any prior knowledge of the stereo parameters (uncaliberated stereo). In order to deal properly with reconstruction, it is needed

to deal with the geometry of stereo which is epipolar geometry as shown in following figure. The epipolar geometry is the intrinsic projective geometry between two views. It is independent of scene structure, and only depends on the cameras' internal parameters and relative pose. So two images of a single scene/object are related by the epipolar geometry, which can be described by a 3x3 singular matrix called the essential matrix if image's internal parameters are known, or the fundamental matrix otherwise. It captures all geometric information contained in two images, and its determination is very important in many applications of computer vision.

The fundamental matrix F encapsulates this intrinsic geometry. It is a 3 x 3 matrix of rank 2. Note that in stereovision, projective geometry techniques are notably applied.

Let o and o**'** be a pair of pinhole cameras in 3D space. Let m and m' be the projections through o and o**'** of a 3D point M in images I and I' respectively. The geometry of these definitions is shown in above figure.

The basic line equation states

$$m'^{T} l' = 0 \tag{2.6}$$

The fundamental matrix maps points in I to lines in **l',** and points in l' to lines in I. This is called epipolar constraint i.e.

$$Fm = l' \tag{2.7}$$

Equations 2.6 and 2.7 lead to

$$m^{T} Fm = 0 \tag{2.8}$$

where *F* is called fundamental matrix and this equation defines the epipolar constraint for all pairs of images correspondences *m* and *m'*. The fundamental

matrix *F* is a 3×3 rank-2 matrix that maps points in *I* to lines in *I'*, and points in *I'* to lines in *I*. For a fundamental matrix F there exists a pair of unique points

$$Fe = F^T e' = 0 \qquad (2.9)$$

Where 0 = [0, 0, 0]^T is the zero vector. The points *e* and *e'* are known as the epipoles of image *I* and image *I'* respectively. The epipoles have the property that all epipolar lines in *I* pass through *e*, similarly all epipolar lines in *I'* pass through *e'*. In 3D space, *e* and *e'* are the intersections of the baseline *oo'* with the planes containing image *I* and *I'*. The set of planes containing the line *oo'* are called epipolar planes. Any 3D point *M* not on line *oo'* will define an epipolar plane, the intersection of this epipolar plane with the plane containing *I* or *I'* will result in an epipolar line. The objective of determining epipolar geometry is to reduce the search space for finding the correspondences between the 2 images. Our vivid

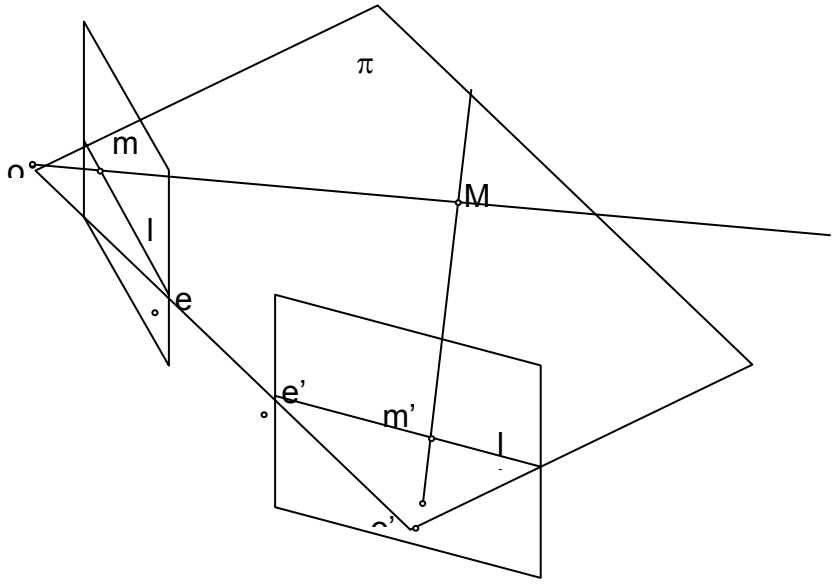


Figure 2.4 Epipolar Planes

3-D perception of the world is due to the interpretation that the brain gives of the computed difference in the retinal position, named disparity between corresponding items. The disparities of all the image points form the so called

disparity map. If the geometry of the viewed scene is known, the disparity map can be converted to a 3-D map of the viewed scene called as the 3-D reconstruction.

◦ The fundamental matrix is the algebraic representation of epipolar geometry. The essential property of the fundamental matrix is that it conveniently encapsulates the epipolar geometry of the uncalibrated imaging configuration. It can be used to reconstruct the scene structure from two uncalibrated views, image rectification, and computation of projective invariants and so on.

A match $m \leftrightarrow m'$ provides a linear constraint on the coefficients of F.

$$m'^{T} F m = 0 \tag{2.10}$$

The fundamental matrix is given as ◦

$$F = \begin{pmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & 1 \end{pmatrix}$$

With the 2 corresponding points $(x, y, 1)^T$ and $(x', y', 1)^T$, the equation $m'^{T} F m = 0$ becomes

$$\begin{pmatrix} x' & y' & 1 \end{pmatrix} \begin{pmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = 0$$

Solving this results into

$$xx' f_{11} + x'y f_{12} + x' f_{13} + xy' f_{21} + yy' f_{22} + y' f_{23} + x f_{31} + y f_{32} + 1 = 0 \tag{2.11}$$

which in the matrix from can be written as

$$\begin{pmatrix} xx' & x'y & x & xy' & yy' & y' & x & y & 1 \end{pmatrix} \begin{pmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ 1 \end{pmatrix} = 0$$

and for 8 corresponding points

$$\begin{pmatrix} x_1x_1' & x'_1y_1 & x_1' & x_1y_1' & y_1y_1' & y_1' & x & y & 1 \\ . & . & . & . & . & . & . & . & 1 \\ . & . & . & . & . & . & . & . & 1 \\ . & . & . & . & . & . & . & . & 1 \\ . & . & . & . & . & . & . & . & 1 \\ . & . & . & . & . & . & . & . & 1 \\ . & . & . & . & . & . & . & . & 1 \\ x_8x_8' & x'_8y_8 & x'_8 & x_8y'_8 & y_8y_8' & y_8' & x_8 & y_8 & 1 \end{pmatrix} \begin{pmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \end{pmatrix} = 0$$

$$Af = 0$$

It was observed that each matching pair of points between the two images provides a single linear constraint on F. This allows F to be estimated linearly (up to the usual arbitrary scale factor) from 8 independent correspondences.

2.4 Geometric Derivation

The mapping from a point in one image to a corresponding epipolar line in the other image may be decomposed into two steps. In the first step, the point m is mapped to some point m**'** in the other image lying on the epipolar line l**'**. This point m**'** is a potential match for the point m. In the second step, the epipolar line **l'** is obtained as the line joining m**'** to the epipole e*'*.

2.4.1 Step 1: Point Transfer Via A Plane

Refer to following figure (Figure 2.5). Consider a plane in space not passing through either of the two camera centers. The ray through the first camera center corresponding to the point m meets the plane π in a point M. This point M is then projected to a point m' in the second image. This procedure is known as transfer via the plane π. Since M lies on the ray corresponding to m, the projected point m' must lie on the epipolar line **l'** corresponding to the image of this ray, as illustrated in figure 1b. The points m and m' are both images of the 3D point M lying on a plane.

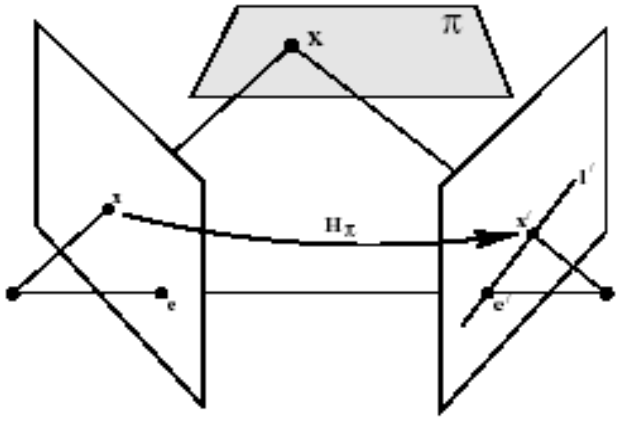


Figure 2.5 Fundamental matrix

The set of all such points m_i in the first image and the corresponding points m'_i in the second image are projectively equivalent, since they are each projectively equivalent to the planar point set M_i. Thus there is a 2D homography H mapping each m_i to m'_i.

2.4.2 Step 2: Constructing The Epipolar Line

Given the point *m'* the epipolar line ***l'*** passing through *m'* and the epipole *e'* can be written as $l' = e' \times m' = [e']_x m'$ (i.e the line can be found by cross product of 2 points) (the notation $[e']_x$ is used for cross product). Since *m'* may be written as *m'* = *Hm*, So

$$l' = [e']_x Hm = Fx \quad (2.12)$$

Where $F = [e']_x H$ is defined as the fundamental matrix. This show following results:

2.4.3 Result 1

The fundamental matrix F may be written as $F = [e']_x H$, where H is the transfer mapping from one image to another via any plane π . Furthermore, since $[e']_x$ has rank 2 and H rank 3, F is a matrix of rank 2.

Geometrically, F represents a mapping from the 2-dimensional projective plane of the first image to the pencil of epipolar lines through the epipole ***e'***. Thus, it represents a mapping from a 2-dimensional onto a 1-dimensional projective space, and hence must have rank 2.

Note, the geometric derivation above involves a scene plane π, but a plane is *not* required in order for F to exist. The plane is simply used here as a means of defining a point map from one image to another. The connection between the fundamental matrix and transfer of points from one image to another via a plane is important.

2.4.4 Correspondence Condition

Up to this point the map m $\longrightarrow$ **l'** defined by F is considered. Now the most basic properties of the fundamental matrix are considered.

2.4.5 Result 2

The fundamental matrix satisfies the condition that for any pair of corresponding points m $\longleftrightarrow$ m' in the two images.

$$m'^T Fm = 0$$

This is true, because if points m and m’ correspond, then m’ lies on the epipolar line l’ = Fm corresponding to the point m. In other words

$$0 = m'^{T} l' = m'^{T} Fm$$

As the vector product of a point and a line is zero if the point lies on the line. Conversely, if image points satisfy the relation $m'^{T} Fm = 0$ then the rays defined by these points are coplanar. This is a necessary condition for points to correspond.

The importance of the relation of result 2 is that it gives a way of characterizing the fundamental matrix without reference to the camera matrices, i.e. only in terms of corresponding image points**.** This enables F to be computed from image correspondences alone. F may also be computed from the two camera matrices, P, P’ and in particular that F is determined uniquely from the cameras, up to an overall scaling.

2.5 Properties of Fundamental Matrix

2.5.1 If F is the fundamental matrix of the pair of cameras (P*;* P’), then F^{T} is the fundamental matrix of the pair in the opposite order: (P’*;* P).

2.5.2 F has seven degrees of freedom: a 3 x 3 homogeneous matrix has eight independent ratios (there are nine elements, and the common scaling is not significant); however, F also satisfies the constraint det F = 0 which removes one degree of freedom.

2.5.3 F is rank 2 homogeneous matrix.

2.5.4 Point correspondence: If m and m’ are corresponding image points, then

$$m'^{T} Fm = 0$$

2.5.5 Epipolar lines:

l’=Fm is the epipolar line corresponding to *m*
l=F^{T}*m’* is the epioplar line corresponding to *m’*

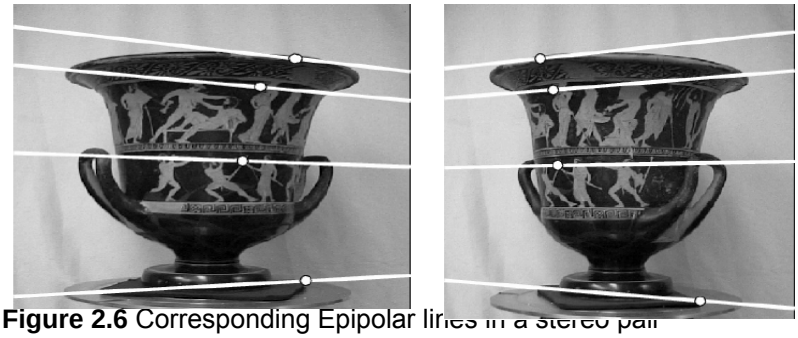


Figure 2.6 Corresponding Epipolar lines in a stereo pair

2.5.6 Epipoles

$Fe=0$ (e is the right null space of F)

$e'^{T}F= 0$ (e' is the left null space of F)

2.5.7 F is a projective map taking a point to a line. In this case a point in the first image m defines a line in the second $l' = Fm$, which is the epipolar line of m. If l and l' are corresponding epipolar lines then any point m on l is mapped to the same line l'. This means there is no inverse mapping, and F is not of full rank. For this reason, F is not a proper correlation (which would be invertible).

2.6 Pure Translation

In considering pure translations of the camera, one may consider the equivalent situation in which the camera is stationary, and the world undergoes a translation In this situation points in 3-space move on straight lines parallel to t, and the imaged intersection of these parallel lines is the vanishing point v in the direction of t. This is illustrated in Figure 2.7. It is evident that e is the epipole for both views, and the imaged parallel lines are the epipolar lines. i.e. has the same coordinates in both images, and points appear to move along lines radiating from the epipole. The epipole in this case is termed the Focus of a point.

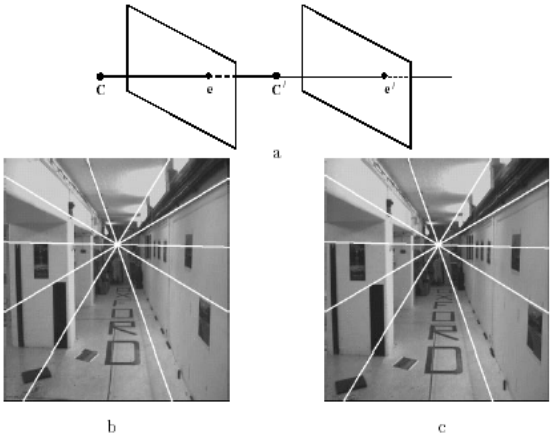


Figure 2.7 Pure Translational Motion.

In Figure 2.7 (a) shows Fixed Point Expansion (FOE) of epipole under the motion (b) and (c) shows the same epipolar lines are overlaid in both cases. Note the motion of the posters on the wall which slide along the epipolar line.

2.7 Geometric Representation Of The Fundamental Matrix

The fundamental matrix can be decomposed into its symmetric and asymmetric parts, and each part is given a geometric representation. The symmetric and asymmetric parts of the fundamental matrix are

$$F_s = (F + F^T)/2 \qquad (2.13)$$

$$F_a = (F – F^T)/2 \qquad (2.14)$$

So that

$$F = F_s + F_a.$$

To motivate the decomposition, consider the points M in 3-space that map to the same point in two images. These image points are fixed under the camera motion so that m =m'. Clearly such points are corresponding and thus satisfy $m^TFm = 0$(though the original condition is $m'^TFm=0$ but as the m and m'

are the same points so this can be used), which is a necessary condition on corresponding points. Now, for any skew-symmetric matrix A the form m^TAm is identically zero. Consequently only the symmetric part of F contributes to $m^TFm = 0$, which then reduces to $m^TF_sm = 0$.

2.8 Difference between Essential matrix and Fundamental matrix

Two perspective images of a single rigid object/ scene are related by the so-called epipolar geometry, which can be described by a 3 × 3 singular matrix. If the internal (intrinsic) parameters of the images (e.g., the focal length, the coordinates of the principal point, ie where camera is placed etc) are known, it is possible to work with the normalized image coordinates, and the matrix is known as the essential matrix; otherwise, work with the pixel image coordinates, and the matrix is known as the fundamental matrix. It contains all geometric information that is necessary for establishing correspondences between two images, from which three-dimensional structure of the perceived scene can be inferred. In a stereovision system where the camera geometry is calibrated, it is possible to calculate such a matrix from the camera perspective projection matrices through calibration. When the intrinsic parameters are known but the extrinsic ones (the rotation and translation between the two images) are not, the problem is known as motion and structure from motion, and has been extensively studied in Computer Vision. The study of uncalibrated images has many important applications. Any geometric information from a projective structure cannot be obtained: measurements of lengths and angles do not make sense. However, a projective structure still contains rich information, such as co planarity, co linearity, and cross ratios (ratio of ratios of distances), which is sometimes sufficient for artificial systems, such as robots, to perform tasks such as navigation and object recognition. In many applications such as the reconstruction of the environment from a sequence of video images where the parameters of the video lens are submitted to continuous modification, camera calibration in the classical sense is not possible.

Any metric information cannot be extracted but a projective structure is still possible if the camera can be considered as a pinhole. Furthermore, if some knowledge of the scene into the projective structure is introduced, a more specific structure of the scene is obtained. For example, by specifying a plane at infinity (in practice, it is needed to specify a plane sufficiently far away), affine structure can be computed, which preserves parallelism and ratios of distances i.e first reconstruct a projective structure, and then use 8 ground reference points to obtain the Euclidean structure and the camera parameters.). The 3D convex hull of an object can be computed from a pair of images whose epipolar geometry is known. If it is assumed that the camera parameters do not change between successive views, the projective invariants can even be used to calibrate the cameras in the classical sense without using any calibration apparatus (known as self-calibration). Even in the case where images are calibrated, more reliable results can be obtained if the constraints arising from uncalibrated images are used as an intermediate step.

CHAPTER 3

SYSTEM DESIGN

3.1 System Modules

Developing a system that computes relative depth requires as its basis the determination of epipolar geometry of the stereo pair, which subsequently needs 8 matching points in a stereo pair. Figure 3.1 illustrates the main blocks of the project. Finding these points lead to the determination of the fundamental matrix

which then simplifies the correspondence problem. With the stereo pair as the input they are processed in these modules and finally the relative depth map is obtained. Below is the high level design of the system being developed. Different modules are given below for easier understanding of the system.

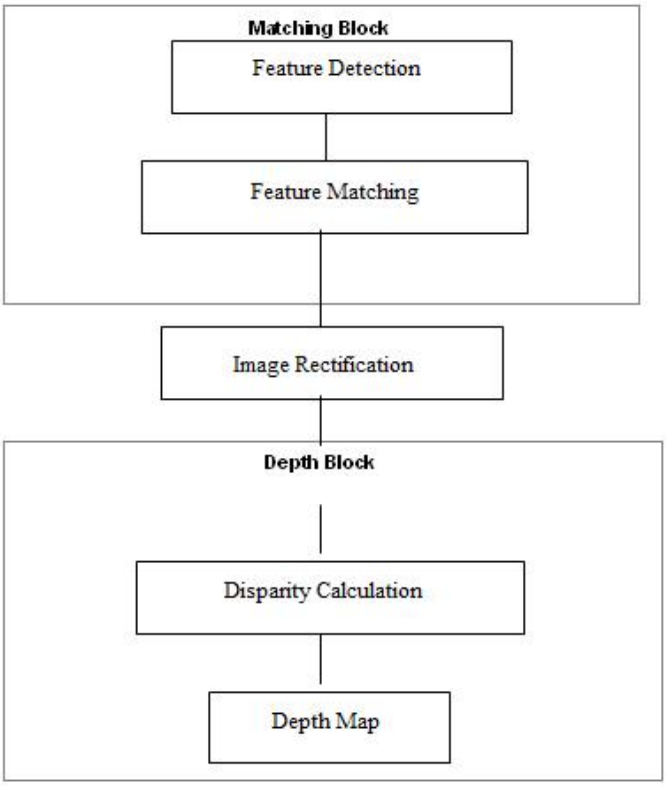


Figure 3.1 System Modular Diagram

3.2 Explanation:

3.2.1 Feature Detection

In feature detection module, basically features would be from the **stereo images.** And after working on different feature detection algorithms it will be examined that which one works better on our images. An algorithm would be better if

1. It detects most of the features in the image.
2. The detected features such as corners are indeed features in the images.

3.2.2 Feature Matching

In this module, matching algorithms is implemented. What this component will do is it will take the features detected previously, as inputs, and find the corresponding pairs of features such as corners. It will compare the features of the left image with the right image and find the best matching pair. After this, fundamental or essential matrix will be calculated depending upon whether calibrated or uncalibrated camera is required. The fundamental or essential matrix will later be used in the epipolar geometry.

3.2.3 Image Rectification

This module move or align the images onto a common image plane using linear transformations. If there are any distortions, they are removed. The distortions can too much brightness or blurriness. The images are aligned in such a way that the epipolar lines become horizontal to the x-axis. The disparities can be found then instead of searching along the skewed scan lines, which is computationally intensive along the image rows.

3.2.4 Disparity Calculation

Disparity is usually computed as a shift to the left of an image feature when viewed in the right image. In this module, different algorithms or techniques will be implemented to calculate disparity such as correlation, SSD, SAD. Its output would be a disparity map and disparity matrix. Different techniques will yield different result.

3.2.5 Depth Map:

This is the last module through which the information extracted from our stereo images pass through. Depth is calculated using the disparity calculated in the previous module and displays it in the form of depth map.

CHAPTER 4

FEATURE EXTRACTION

4.1 Overview

In pattern recognition and in image processing, Feature extraction is a special form of dimensionality reduction. When the input data to an algorithm is too large to be processed and it is suspected to be notoriously redundant (much data, but not much information) then the input data will be transformed into a reduced representation set of features (also named features vector). Transforming the input data into the set of features is called features extraction. If the features extracted are carefully chosen it is expected that the features set will extract the relevant information from the input data in order to perform the desired task using this reduced representation instead of the full size input.

Feature extraction involves simplifying the amount of resources required to describe a large set of data accurately. When performing analysis of complex data one of the major problems stems from the number of variables involved. Analysis with a large number of variables generally requires a large amount of memory and computation power or a classification algorithm which over fits the training sample and generalizes poorly to new samples. Feature extraction is a general term for methods of constructing combinations of the variables to get around these problems while still describing the data with sufficient accuracy.

The concept of feature detection refers to methods that aim at computing abstractions of image information and making local decisions at every image point whether there is an image feature of a given type at that point or not. The resulting features will be subsets of the image domain, often in the form of isolated points, continuous curves or connected regions.

4.2 Feature

There is no universal or exact definition of what constitutes a feature, and the exact definition often depends on the problem or the type of application. Given that, a feature is defined as an "interesting" part of an image, and features are used as a starting point for many computer vision algorithms. Since features are used as the starting point and main primitives for subsequent algorithms, the overall algorithm will often only be as good as its feature detector. Consequently, the desirable property for a feature detector is repeatability: whether or not the same feature will be detected in two or more different images of the same scene.

4.2.1 Low level Image processing

Feature detection is a low-level image processing operation. That is, it is usually performed as the first operation on an image, and examines every pixel to see if there is a feature present at that pixel. If this is part of a larger algorithm, then the algorithm will typically only examine the image in the region of the features. As a built-in pre-requisite to feature detection, the input image is usually smoothed by a Gaussian kernel in a scale-space representation and one or several feature images are computed, often expressed in terms of local derivative operations.

Occasionally, when feature detection is computationally expensive and there are time constraints, a higher level algorithm may be used to guide the feature detection stage, so that only certain parts of the image are searched for features.

Where many computer vision algorithms use feature detection as the initial step, so as a result, a very large number of feature detectors have been developed. These vary widely in the kinds of feature detected, the computational complexity and the repeatability. At an overview level, these feature detectors can (with some overlap) be divided into the Edges, Corners, Blobs and Ridges groups.

4.3 Feature extraction in aerial data

Feature detection has been key issue in all computer vision algorithms. Features are basically points that describe the image. And based upon that feature points one can correlate two or more images. All tracking and 3D reconstruction depends on how good the feature points are. A good set of feature points leads to a better result in the end and bad points could mislead the whole process

A lot of work has been done in feature point detection and analysis for tracking purposes. Harris Corner detector proposed in 1988 is still being used for feature detection. But the problem is Harris detector is affected by scale changes in data set. SIFT is invariant to scale changes, rotation and 3D view point. So this can be used for reliable image matching. A good features extraction algorithm should possess properties like detection of all feature points, no false feature point is detected, feature point is well localized, robust to noise algorithm and efficient in terms of time and memory [1].

There are algorithms available after the research of Moravec. Harris corner detector proposed by Harris and Stephens [2] gives good corner points for reliable image matching. It is invariant to affine image changes like rotation and translation but fails when scale changes are very large. At small changes it may give good results. It just uses derivatives of intensities to check for corner points in images.

4.3.1 Harris Corner detector

The algorithm works by taking a local window and shifting it along the image. It then checks for intensity changes in image. The concept of autocorrelation was used in this approach.

The hessian matrix is given by:

$$\mathbf{C} = \begin{bmatrix} \sum I_x^2 & \sum I_{xy} \\ \sum I_{xy} & \sum I_y^2 \end{bmatrix} \tag{4.1}$$

The strength of the corner is determined by how much the second derivative is there. Based upon Eigen values of C (α, β) following inferences can be made:

For a flat region: no change in all directions; small α and β

For an edge: no change along the edge direction; small α and large β or vice versa

For corner: significant change in all directions; large α and β

As the exact Eigen value computation is not possible so Harris and Stephen proposed following function:

$$R = Det(\mathrm{C}) - \kappa Tr^2(\mathrm{C}) \quad (4.2)$$

Where Det(C) =αβ and Tr(C) =α + β

Harris corner detector is not invariant to image scaling. This shortcoming or defect was corrected by SIFT [3,4].

4.3.2 Scale invariant feature transform (SIFT)

Harris corner detector is just invariant to affine image changes. But in case of UAV data a lot of scale and 3D viewpoint changes. SIFT provide reliable and scale invariant features to track through the data stream. It is not only robust to affine image changes but also invariant to scale and 3D orientation changes.

The key points in the working of algorithm are stated as follows:

Scale space construction: Construction of Gaussian and Difference of Gaussian (DoG) pyramids

Key point localization: Key points are selected from the scale space based upon there stability measure and local extrema.

Orientation assignment: Orientations are assigned to each key point based on there histograms.

Key point descriptor: representation in 128-dimensional vector.

This approach has been named as Scale Invariant feature transform as it transforms original data into scale invariant-coordinates relative to local features. Little explanation to above algorithm is as follows:

4.3.2.1 Scale space construction

Interesting image points can be found out using cascading approach that filters out the potential candidates for interest points that can be used for further research. First step is to find image locations that are invariant to scale changes.

This can be accomplished using scale function that searches for scale invariant features. The convolution kernel in this case is Gaussian. Image is convolved with Gaussian kernel.

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{4.3}$$

To detect scale invariant features, difference of Gaussian blurred images at scales σ and kσ is computed given by equation 4.4 and 4.5 to form a pyramid like structure (Fig 4.1).

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \tag{4.4}$$

$$= L(x, y, k\sigma) - L(x, y, \sigma). \tag{4.5}$$

As the DoG function is closes approximation of scale normalized Laplacian of Gaussian. Maxima and minima of this scale normalized LoG gives scale invariant features as compared to others like image gradient, Harris operator or Hessian matrix.

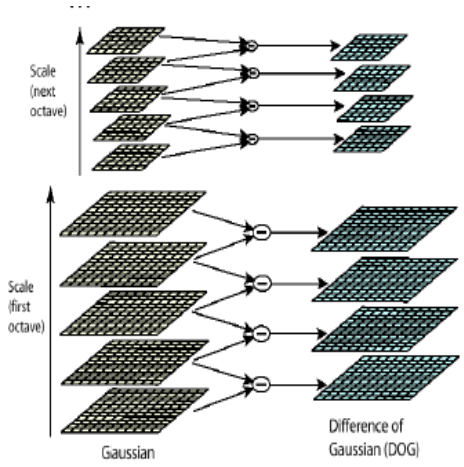


Figure 4.1: Figure showing Gaussian blurred images and there difference forming pyramid of DoG[3]

4.3.2.2 Key point localization

4.3.2.2.1 Local space detection

Local maxima and minima of DoG function are considered as feature points or key point candidates. Further processing will be done on these to filer out the non-scale invariant features. Interest points (called key points in the SIFT

framework) are identified as local maxima or minima of the DoG images across scales. Each pixel in the DoG images is compared to its 8 neighbors at the same scale, plus the 9 corresponding neighbors at neighboring scales. If the pixel is a local maximum or minimum, it is selected as a candidate key point (Figure 4.2).

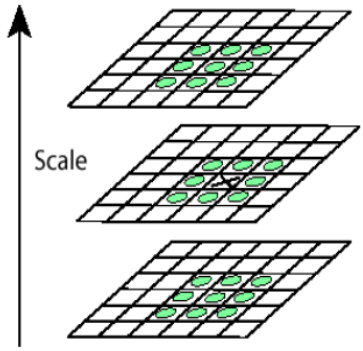


Figure 4.2: Figure showing extrema extraction.

Pixel marked with X is compared with eight neighboring and 9 pixels in adjacent scales of DoG (From Lowe 2004)

4.3.2.2.2 Rejecting low contrast key points

Once candidate feature points are found then they are scanned to get the feature points which are low contrast because they are more prone to noise and are poorly localized along an edge.

4.3.2.2.3 Eliminating Edge responses

Using low contrast filtering will not give that much good result as DoG has strong response at edge as well because of its sensitivity to noise. Edges can easily be detected by using 2 X 2 Hessian matrix. A poorly defined peak will have large principal curvature at edge and small in perpendicular direction. Calculating curvature through Hessian matrix:

$$\mathbf{H} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \tag{4.6}$$

The derivatives are computed as a difference of neighboring points. Eigen values are proportional to principal curvature of D. .

Trace of H is calculated which sum of highest Eigen value α and lowest Eigen value β given below:

$$Tr(\mathbf{H}) = D_{xx} + D_{yy} = \alpha + \beta \tag{4.7}$$

The determinant is given by equation

$$Det(\mathbf{H}) = D_{xx}D_{yy} - (D_{xy})^2 = \alpha\beta \tag{4.8}$$

If the determinant has different signs it means that point can be discarded. If ratio of Eigen values instead of their original values is taken then that would be more robust.

Let α be highest Eigen value and β being lowest such that α=rβ then ratio is given by equation:

$$\frac{Tr(\mathbf{H})^2}{Det(\mathbf{H})} = \frac{(r+1)^2}{r} \tag{4.9}$$

Lowe suggested to take value of r =10 for edge elimination.

4.3.2.3 Orientation Assignment

To determine the key point orientation, a gradient orientation histogram is computed in the neighborhood of the key point (using the Gaussian image at the closest scale to the key point's scale). The contribution of each neighboring pixel is weighted by the gradient magnitude and a Gaussian window with a σ that is 1:5 times the scale of the key point.

For a particular Gaussian image L(x,y) the gradient magnitude and orientation is computed as:

$$m(x,y) = \sqrt{L_x^2 + L_y^2} \tag{4.10}$$

$$\theta(x,y) = \tan^{-1}(L_y/L_x) \tag{4.11}$$

Where m is magnitude and θ is orientation and Lx and Ly are consecutive pixel differences. Peaks in the histogram correspond to dominant orientations. A separate key point is created for the direction corresponding to the histogram maximum and any other direction within 80% of the maximum value. So for multiple points of same magnitude there will be multiple key points created at same location with different orientations.

4.3.2.4 Key point descriptor

Once orientations with respect to scale and location are assigned for each key point then a 2 dimensional coordinate system may be used to enforce the invariance to these features and to describe the local image region. In the next step a local descriptor is calculated that provides in variance to 3D pose and illumination changes. This descriptor is distinct for local image region.

4.4 Comparison of Harris and SIFT for feature detection

Harris corner detector not only detects corners but also highlights the points which have maximum intensity change in particular region. Experiments have been conducted on various frames extracted from UAV data to make comparisons of different techniques. It was found that Harris gives lesser points as compared to SIFT. SIFT features are numerous in number but they are more reliable and lesser false points were detected. The results can be found at the end. Both of features are used in this project. Both have their own advantages and disadvantages in term of computational complexity. Harris gives accurate results in lesser time.

4.5 Experimental Results:

Figure 4.3 Original Stereo Pair

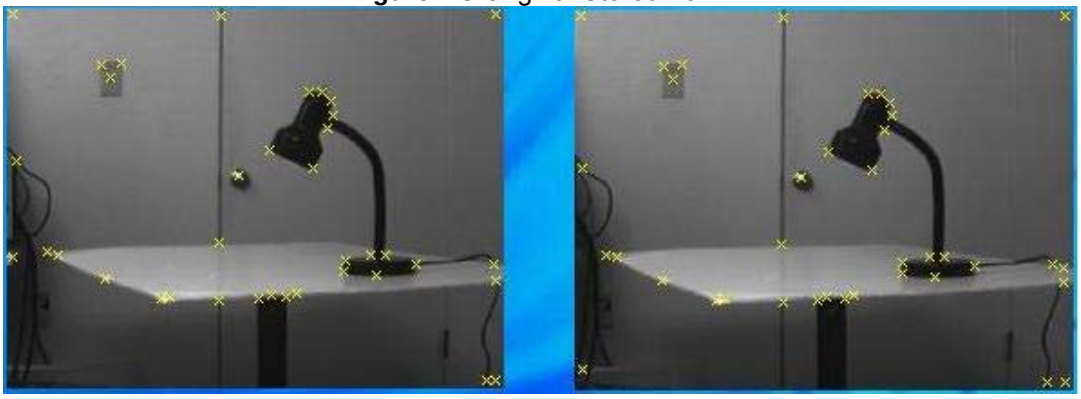
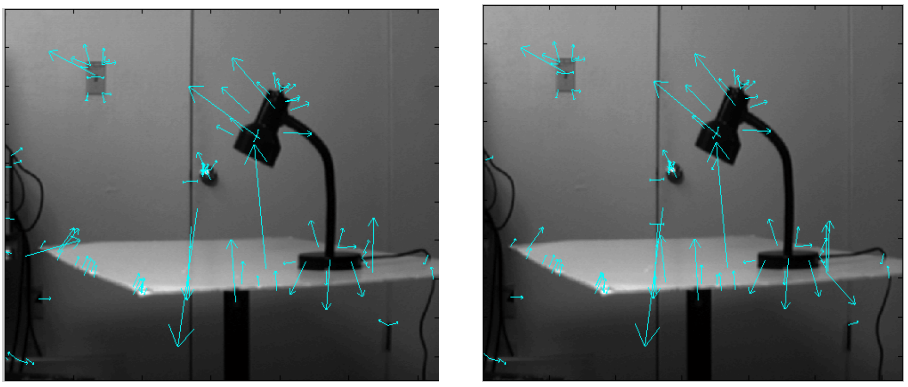


Figure 4.4 Results of Harris Corner Detector, 37 Corners on Left Image and 38 Corners on Right Image



(a) (b)

Figure 4.5 SIFT feature extractor results (a) 100 keypoints (b) 92 keypoints

4.6 Discussion

It has been noticed in the above results that Harris corner detector gave accurate results except few. This is because of the reason that it takes into account extreme intensity changes and declares them as corners (though they are not). If window size for filter is increased then there will be lesser points for that particular image and if window size is decreased then more points will be detected as a result of local interactions. Harris corners can be used as reliable features. As ratio of inaccurately detected points is considerably less.

SIFT on the other hand is computationally intensive. The arrows above show the orientation and magnitude of keypoint. It requires a lot of computations and is not recommended if image contains a lot of information. As only eight points are needed for fundamental matrix estimation so Harris is more desirable which gives speedy and reliable results.

Chapter 5

IMAGE CORRESPONDENCE

5.1 Overview

Image matching has also been another major research area in computer vision algorithms. It is a key problem of computer vision and frequently used in 3D-model reconstruction, object recognition, image alignment, camera self-calibration and so on. Feature point matching is the most common one among all kinds of image matching.

Correspondence between two frames is needed to exploit various differences in them for 3D reconstruction from aerial data. This also helps in tracking points from frame to frame. Correspondence can be find using both Harris feature points and SIFT feature points.

There are two schools of thought for solving the feature correspondence problem. In the first one, features are detected in one image and then correspondences for each of them are sought for in the second image, generally via multi-scale techniques. In the second approach, features are detected independently in both images and then matched up usually by relaxation. Incidentally, recent state-of-the-art work on the fundamental matrix estimation follows this latter avenue for achieving initial correspondences.

Due to its inherent combinatorial complexity and illposedness, feature correspondence is one of the hardest low-level image analysis tasks. The problem can be stated as finding pairs of features in two (or more) perspective views of a scene such that each pair corresponds to the same scene point. Some of the techniques that were analyzed include illustrious works notably by Ullman [5] and Marr and Poggio [6]. In particular, Ullman put forward his minimal mapping 1 theory to implement three intuitive local criteria for establishing good global mapping that are: the principle of similarity, principle of proximity (other things being equal, choose the closest) and the principle of exclusion (only one-to-one matching are allowed). As Marr pointed out, by simple local interactions a good global mapping effect can often be achieved. A vast amount of work has been done on this subject. Most methods have a sometime complicate algorithmic formulation.

5.2 The Scott and Longuet-Higgins Algorithm

In a landmark paper [8], Scott and Longuet-Higgins proposed a neat, direct way of associating features of two arbitrary patterns. The algorithm exploits some properties of the singular value decomposition (SVD) to satisfy both the exclusion and proximity principles set forth by Ullman. A remarkable feature of the algorithm is its straightforward implementation founded on a well-conditioned eigenvector solution, which involves no explicit iterations.

Let *I* and *I'* be two images, containing m features $I_i\ (i = 1...m)$ and n features $I'_j\ (j = 1...n)$, respectively, which are desired to be in one-to-one correspondence. The algorithms consist of three stages.

Build a proximity matrix G of the two sets of features where each element

G_{ij} is Gaussian-weighted distance between two features I_i and I'_j

$$G_{ij} = e^{-r_{ij}^2 / 2\sigma^2} \quad i = 1...m,\ j = 1...n \tag{5.1}$$

where $r_{ij} = \left\| I_i - I'_j \right\|$ is their Euclidean distance if they are regarded as lying on the same plane. G is positive definite and G_{ij} decreases monotonically from 1 to 0 with distance. The parameter σ controls the degree of interaction between the two sets of features i.e the distance between the features: a small value of σ enforces local interactions, while a larger value permits more global interactions.

Perform the singular value decomposition (SVD) of $G \in M_{m,n}$ i.e a matrix of m rows and n columns.

$$G = UDV^T$$

where $U \in M_m$ and $V \in M_n$ are orthogonal matrices and the diagonal matrix $D \in M_{m,n}$ contains the (positive) singular values along its diagonal elements D_{ij}

in descending numerical order. If m < n, only the first m columns of U have any significance.

Convert D to a new matrix E obtained by replacing every diagonal element

Dii with 1 and then compute the product

$$P = UEV^T$$

This new matrix $P \in M_{m,n}$ has the same shape as the proximity matrix G and has the interesting property of sort of 'amplifying' good pairings and 'attenuating' bad ones. ''if P_{ij} is both the greatest element in its row and the greatest element in its column, then those two different features I_i and I'_j are regarded as being in 1:1 correspondence with one another; if this is not the case, it means that features I_i competes unsuccessfully with other features for partnership.

5.3 Rogue point analysis:

There must be some points that are visible in one image but not visible in second image. Such points are called rogue points. To count for such points normalized cross correlation is used. Saying that the Scott and Longuet-Higgins algorithm does not embed the feature similarity principle, so dear to most stereo correspondence approaches, can summarize this behavior. Obviously, this behavior calls for the use of some local measurements to quantify feature similarity, such as the normalized (cross) correlation between gray level patches about the features.

If two w x w areas centered on features I_i and I'_j are represented as two w x w arrays of pixel intensities A and B, respectively, the normalized correlation is defined as

$$C_{ij} = \frac{\sum_{u=1}^{W}\sum_{v=1}^{W}\left(A_{uv} - \overline{A}\right)\bullet\left(B_{uv} - \overline{B}\right)}{W^2 \bullet \sigma(A) \bullet \sigma(B)} \qquad (5.2)$$

where $\overline{A}(\overline{B})$ is the average and $\sigma(A)(\sigma(B))$ the standard deviation of all the elements of A (B). C_{ij} varies from -1 for completely uncorrelated patches to 1 for identical patches. One way of including this correlation information into the proximity matrix is to transform the elements of *G* as follows:

$$G_{ij} = \left[e^{-(C_{ij}-1)^2/2\gamma^2}\right] \bullet e^{-\gamma_{ij}^2/2\sigma^2} \qquad (5.3)$$

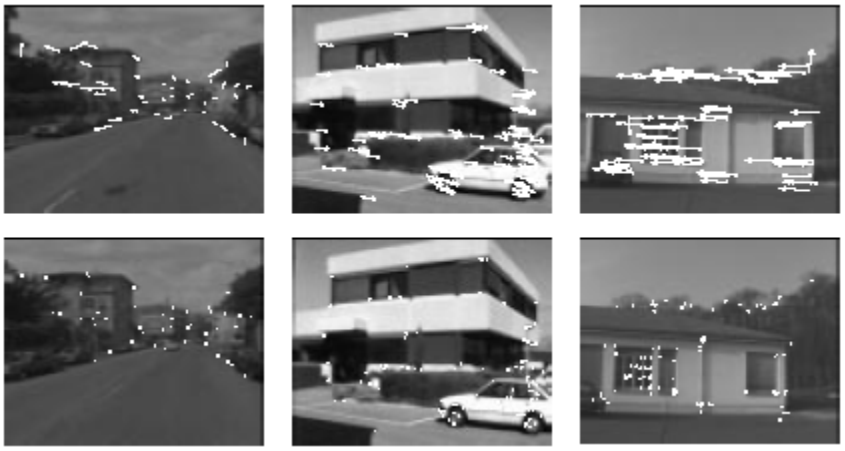


Figure 5.1 Some more test images pairs. Disparities are overlaid onto the top images and matching corners onto the bottom ones.

where term in bracket is a gaussian-weighted function of the correlation C_{ij} in which γ determines how quickly its values decreases with a diminishing C_{ij} (σ = 0.4 (for this algorithm).

This new correspondence strength can be seen as a correlation-weighted proximity. It is easy to see that the elements of G still range from 0 to 1 and, as in

equation (5.1), the closer and the more correlated two features I_i and I'_j are, the higher G_{ij} is going to be. This new correspondence strength now embodies similarity between features and is therefore much more selective than just proximity as in Equation (5.1). In some ways, by applying the algorithm with the said correlation-weighted G, a minimum overall distance mapping is obtained still complying to the proximity and uniqueness principles but under the constraint of similarity. It can be seen in figure that a considerably higher number of 1:1 matches has been found. Figure 5.1 shows corresponding pairs. Figure 5.2 shows correspondence using Harris corner points and Figure 5.3 shows correspondence using SIFT features.

5.4 Experimental Results:

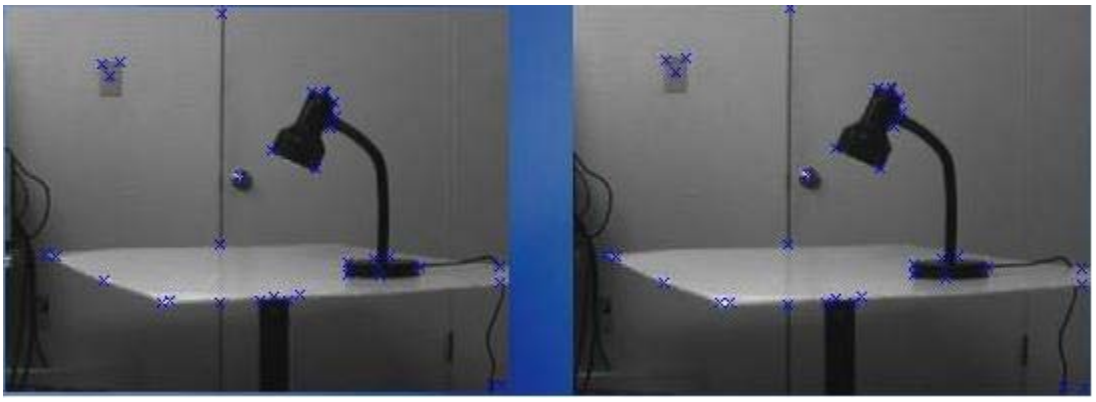


Figure 5.2 Feature correspondences in using Harris feature detector (33 corresponding features on both images)

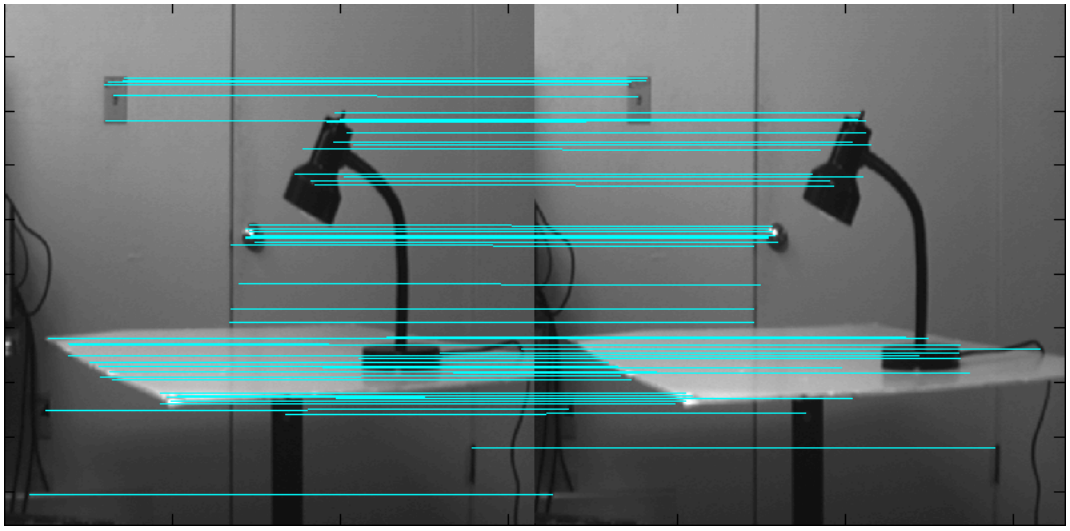


Figure 5.3 Feature correspondence using SIFT features (76 matches)

5.5 Discussion

In figure 5.2 the Harris corner points are used and by exploiting the properties of singular value decomposition correspondence is found between two images. As Scott and Higgins algorithm makes use of Similarity principle so reliable results are found and the highlighted points we get are exactly in 1:1 correspondence with each other. Not even a single noisy result was found in this image when Harris features were used.

In case of SIFT matching, again the algorithm showed more complexity in terms of memory and time. So to get fast reliable image matching with points in 1:1 correspondence Harris matching feature points should be used which serve as basis for further operations on image. SIFT should only be used when there are scale and orientation changes in image pair. In that case SIFT would give excellent results as it is invariant to scale and orientation changes upto a certain level.

Chapter 6

EPIPOLAR GEOMETRY and FUNDAMENTAL MATRIX

6.1 Overview

Epipolar geometry refers to the geometry of stereo vision. When two cameras view a 3D scene from two distinct positions, there are a number of geometric relations between the 3D points and their projections onto the 2D images that lead to constraints between the image points. Epipolar geometry contains all geometric information contained in the two images. A stereo system can compute a great deal of 3-D information without any prior knowledge of the stereo parameters (uncaliberated stereo). In order to deal properly with reconstruction, it is necessary to deal with the geometry of stereo which is epipolar geometry

The epipolar geometry is the intrinsic projective geometry between two views. It is independent of scene structure, and only depends on the cameras' internal parameters and relative pose. So two images of a single scene/object are related by the epipolar geometry, which can be described by a 3x3 singular matrix called the essential matrix if image's internal parameters are known, or the fundamental matrix otherwise. It captures all geometric information contained in two images, and its determination is very important in many applications of computer vision.

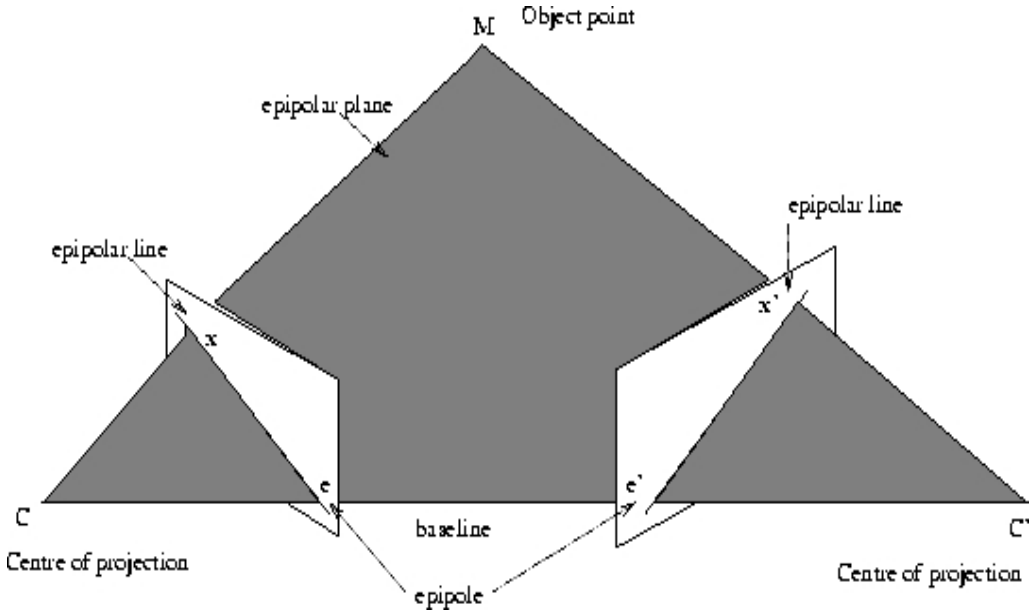


Figure 6.1 The Epipolar Constraint

At first it might seem that correspondence requires a search through the whole image, but the epipolar constraint reduces this search to a single line. Figure 6.1 shows the epipolar constraint between two planes

6.2 Definitions

6.2.1 Epipole

The epipole is the point of intersection of the line joining the optical centres that is the baseline, with the image plane. Thus the epipole is the image, in one camera, of the optical centre of the other camera [9].

6.2.2 Epipolar plane

The epipolar plane is the plane defined by a 3D point M and the optical centres C and C'.

The epipolar line is the straight line of intersection of the epipolar plane with the image plane. It is the image in one camera of a ray through the optical centre and image point in the other camera. All epipolar lines intersect at the epipole[10].

Thus, a point x in one image generates a line in the other on which its corresponding point x' must lie. It can be observed that search for correspondences is thus reduced from a region to a line.

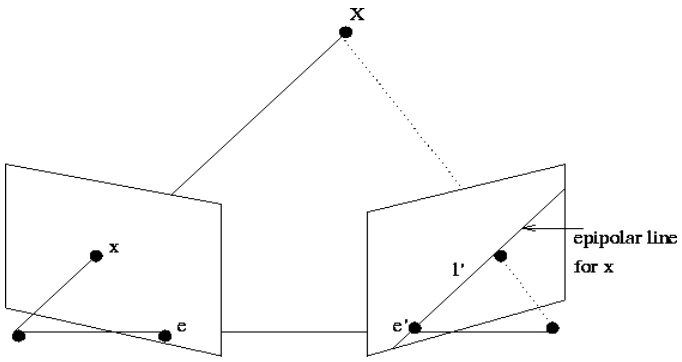


Figure 6.2 The epipolar line along which the corresponding point for X must lie

The fundamental matrix F encapsulates this intrinsic geometry. It is a 3 x 3 matrix of rank 2. Note that in stereovision, projective geometry techniques are notably applied. Figure 6.2 shows the corresponding epipolar line on which the point must lie.

6.3 Basic Equations

Let o and o' be a pair of pinhole cameras in 3D space. Let m and m' be the projections through o and o' of a 3D point M in images I and I' respectively. The geometry of these definitions is shown in a figure given below.

The basic line equation states

$$m^T l=0$$

The fundamental matrix maps points in I to lines in I', and points in I' to lines in I. This is called epipolar constraint i.e.

$$Fm=l'$$

Above two equations lead to

$$m^T F m = 0$$

where *F* is called fundamental matrix and this equation defines the epipolar constraint for all pairs of images correspondences m and m'. The fundamental matrix *F* is a 3×3 rank-2 matrix that maps points in *I* to lines in *I'*, and points in *I'* to lines in *I*. For a fundamental matrix F there exists a pair of unique points

$$Fe = F^T e' = 0$$

where $0 = [0, 0, 0]^T$ is the zero vector. The points e and e' are known as the epipoles of image I and image I' respectively. The epipoles have the property that all epipolar lines in I pass through e, similarly all epipolar lines in I' pass through e'. In 3D space, e and e' are the intersections of the baseline oo' with the planes containing image I and I'. The set of planes containing the line oo' are called epipolar planes. Any 3D point M not on line oo' will define an epipolar plane, the intersection of this epipolar plane with the plane containing I or I' will result in an epipolar line. The objective of determining epipolar geometry is to reduce the search space for finding the correspondences between the 2 images.

Our vivid 3-D perception of the world is due to the interpretation that the brain gives of the computed difference in the retinal position, named disparity between corresponding items. The disparities of all the image points form the so called disparity map. If the geometry of the viewed scene is known, the disparity map can be converted to a 3-D map of the viewed scene called as the 3-D reconstruction.

6.4 Fundamental Matrix

The fundamental matrix is the algebraic representation of epipolar geometry. It captures all geometric information contained in the two images. The essential property of the fundamental matrix is that it conveniently encapsulates the epipolar geometry of the uncalibrated imaging configuration. It can be used to reconstruct the scene structure from two uncalibrated views, image rectification, and computation of projective invariants and so on.

A match m←→m' provides a linear constraint on the coefficients of F[11].

$$m^T F m = 0$$

The fundamental matrix is given as

$$F = \begin{pmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & 1 \end{pmatrix}$$

With the 2 corresponding points $(x,y,1)^T$ and $(x',y',1)^T$ the equation $m^T F m = 0$ becomes

$$\begin{pmatrix} x' & y' & 1 \end{pmatrix} \begin{pmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = 0$$

Solving this results into

$$xx'f_{11}+x'yf_{12}+x'f_{13}+xy'f_{21}+yy'f_{22}+y'f_{23}+xf_{31}+yf_{32}+1=0$$

which in the matrix from can be written as

$$\begin{pmatrix} xx' & x'y & x & xy' & yy' & y' & x & y & 1 \end{pmatrix} \begin{pmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ 1 \end{pmatrix} = 0$$

And for 8 corresponding points

$$\begin{pmatrix} x_1x_1' & x'_1y_1 & x_1' & x_1y_1' & y_1y_1' & y_1' & x & y & 1 \\ . & . & . & . & . & . & . & . & 1 \\ . & . & . & . & . & . & . & . & 1 \\ . & . & . & . & . & . & . & . & 1 \\ . & . & . & . & . & . & . & . & 1 \\ . & . & . & . & . & . & . & . & 1 \\ . & . & . & . & . & . & . & . & 1 \\ x_8x_8' & x'_8y_8 & x'_8 & x_8y'_8 & y_8y_8' & y_8' & x_8 & y_8 & 1 \end{pmatrix} \begin{pmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \end{pmatrix} = 0$$

$$A\, f = 0$$

Each matching pair of points between the two images provides a single linear constraint on F. This allows F to be estimated linearly (up to the usual arbitrary scale factor) from 8 independent correspondences[12].

6.5 Recovering Epipolar geometry

Process of computing fundamental matrix is known as recovering epipolar geometry. The essential property of the fundamental matrix is that it conveniently encapsulates the epipolar geometry of the uncalibrated imaging configuration. Hartley's 8 point normalized algorithm requires the matching block will provide 8 independent correspondences and these correspondences[19].

The eight-point algorithm for computing the essential matrix was introduced by Longuet-Higgins. The essential matrix is used to compute the structure of a scene from two views with calibrated cameras. The great advantage of the eight-point algorithm is that it is linear, hence fast and easily implemented. If eight point matches are known, then the solution of a set of linear equations is involved. With more than eight points, a linear least squares minimization problem must be solved. One notices immediately that the same algorithm may be used to compute a matrix with this property from uncalibrated cameras. In this case of uncalibrated cameras it has become customary to refer to the matrix so derived as the fundamental matrix. Just as in the calibrated case, the fundamental matrix may be used to reconstruct the scene from two uncalibrated views, but in this case only up to a projective transformation.

Apart from scene reconstruction, the fundamental matrix may also be used for many other tasks, such as image rectification computation of projective invariants outlier detection and stereo matching. Unfortunately, despite its simplicity the eight-point algorithm has often been criticized for being excessively sensitive to noise in the specification of the matched points. Indeed this belief has become the prevailing wisdom. Consequently, because of its importance, many alternative algorithms have been proposed for the computation of the fundamental matrix. Without exception, these algorithms are considerably more complicated than the eight-point algorithm. The poor performance of the eight-point algorithm can probably be traced to implementations that do not take sufficient account of numerical considerations, most specifically the condition of the set of linear equations being solved. Hartley showed that a simple transformation (translation and scaling) of the points in the image before formulating the linear equations leads to an enormous improvement in the condition of the problem and hence of the stability of the result. The added complexity of the algorithm necessary to do this transformation is insignificant. It is not claimed that this modified eight-point algorithm will perform quite as well as the best iterative algorithms. However it is shown by Hartley in thousands of experiments on many images that the difference is not very great between the modified eight-point algorithm and iterative techniques. Indeed the eight-point algorithm does better than some of the iterative techniques.

6.6 Linear Solution for the Fundamental Matrix

The fundamental matrix is defined by the equation

$$m^T F m = 0$$

for any pair of matching points $m \leftrightarrow m'$ in two images. Given sufficiently many point matches $m_i \leftrightarrow m_i'$ (at least eight) this equation can be used to compute the unknown matrix F. In particular, writing $m=(x,y,1)^T$ and $m'(x',y',1)^T$ with x showing the column no and y showing the row no, each point match gives rise to one linear equation in the unknown entries of F. The coefficients of this equation

are easily written in terms of the known coordinates. Specifically, the equation corresponding to a pair of points $(x,y,1)^T$ and $(x',y',1)^T$ will be[11]

$$xx'f_{11}+x'yf_{12}+x'f_{13}+xy'f_{21}+yy'f_{22}+y'f_{23}+xf_{31}+yf_{32}+f_{33}=0$$

The row of the equation matrix may be represented as a vector

$$(xx',x'y,x,yx',yy',y,x',y',1)$$

From all the point matches, a set of linear equations of the form

$$Af = 0 \qquad (6.1)$$

Is obtained where f is a nine-vector containing the entries of the matrix *F*, and *A* is the equation matrix. The fundamental matrix *F*, and hence the solution vector f is defined only up to an unknown scale. For this reason, and to avoid the trivial solution f, the additional constraint is made

$$|f| = 1$$

where |f| is the norm of f. Under these conditions, it is possible to find a solution with as few as eight point matches. With more than eight point matches, there is an over-specified system of equations. Assuming the existence of a non-zero solution to this system of equations, it is deduced that the matrix *A* must be rank-deficient. In other words, although *A* has nine columns, the rank of *A* must be at most eight. In fact, except for exceptional configurations the matrix *A* will have rank exactly eight, and there will be a unique solution for f. This previous discussion assumes that the data is perfect, and without noise. In fact, because of inaccuracies in the measurement or specification of the matched points, the matrix A will not be rank-deficient, it will have rank nine. In this case, it is not possible to find a non-zero solution to the equations *A*f = 0.

Instead, a least-squares solution to this equation set is considered. In particular, the vector f that minimizes |Af| subject to the constraint $|f| = f^T f = 1$ is found. It is well known (and easily derived using Lagrange multipliers) that the solution to this problem is the unit eigenvector corresponding to the smallest eigenvalue of A^TA. Note that since

A^TA is positive semi-definite and symmetric, all its eigenvectors are real and positive, or zero. For convenience, (though somewhat inexactly), this eigenvector is called the *least eigenvector* of A^TA. An appropriate algorithm for finding this eigenvector is the algorithm off the Singular Value Decomposition.

6.6.1 The Singularity Constraint

An important property of the fundamental matrix is that it is singular, in fact of rank two. Furthermore, the left and right null-spaces of F are generated by the vectors representing (in homogeneous coordinates) the two epipoles in the two images. Most applications of the fundamental matrix rely on the fact that it has rank two. The matrix *F* found by solving the set of linear equations (6.1) will not in general have rank two, and steps are taken to enforce this constraint. The most convenient way to enforce this constraint is to correct the matrix *F* found by the solution of (6.1). Matrix *F* is replaced by the matrix F' that minimizes the Frobenius norm |F – F'| subject to the condition
det F' = 0. A convenient method of doing this is to use the Singular Value Decomposition (SVD). In particular, let

$$F = UDV^T$$

be the SVD of *F*, where *D* is a diagonal matrix D=diag(r,s,t) satisfying r≥s≥t.
Let

$$F' = Udiag(r,s,0)V^T$$

Minimizing the difference between F and F' in Frobenius norm has little theoretical justification, and in fact there are other methods of enforcing the singularity constraint a posteriori which have more theoretical basis. However, this method gives good results. Thus, the eight-point algorithm for computation of the fundamental matrix may be formulated as consisting of two steps, as follows:

6.6.1.1 Linear Solution

Given point matches $m_i \leftrightarrow m_i'$, solve the equations $m^T F m=0$ to find F. The solution is the least eigenvector, f of $A^T A$, where A is the equation matrix.

6.6.1.2 Constraint Enforcement

Replace F by F', the closest singular matrix to F under Frobenius norm. This is done using the Singular Value Decomposition. The algorithm thus stated is extremely simple and rapid to implement, assuming the availability of a suitable linear algebra library.

6.6.2 A Measure of Comparison or Difference between Two Fundamental Matrices

In order to find the differences between two fundamental matrices, let the two given fundamental matrices be F_1 and F_2. The measure is computed as follows:

Step 1: Choose *randomly* a point *m* in the first image.
Step 2**:** Draw the epipolar line of *m* in the second image using F_1. The line is shown as a dashed line, and is denoted by F_1m.
Step 3**:** If the epipolar line does not intersect the second image, go to **Step 1** i.e choose another point.
Step 4**:** Choose *randomly* a point *m'* on the epipolar line. Note that *m* and *m'* correspond to each other exactly with respect to F_1.
Step 5**:** Draw the epipolar line of *m* in the second image using F_2, i.e., F_2m, and compute the distance, noted by d'_1 between point *m'* and line F_2m.
Step 6**:** Draw the epipolar line of *m'* in the first image using F_2, i.e., $F^T{}_2m'$, and compute the distance, noted by d_1, between point *m* and line $F^T{}_2m'$.
Step 7**:** Conduct the same procedure from **Step2** through **Step 6**, but reversing the roles of *F1* and F_2, and compute d_2 and d'_2.
Step 8**:** Repeat *N* times **Step 1** through **Step 7**.
Step 9**:** Compute the average distance of *d*'s, which is the measure of difference between the two fundamental matrices.

With normalization of the coordinates in order to improve the condition of the problem, the eight-point algorithm performs almost as well as the best iterative algorithms. On the other hand, it runs about 20 times faster and is far easier to

code. There seems to be little advantage in choosing the non-isotropic scaling scheme for the normalization transform, since the simpler isotropic scaling performs just as well. Without normalization of the inputs, however, the eight-point algorithm performs quite badly, often with errors as large as 10 pixels, which makes it virtually useless. If extra accuracy is needed and an iterative algorithm is used, it is best to use the normalized, rather than the unnormalized eight-point algorithm to provide a starting point for iteration. Difficulties with stopping criteria, as well as the risk of finding a local minimum mean that the quality of the iteratively estimated result depends on the initial estimate. The technique of data normalization described above is widely applicable to other problems. Among others it is directly applicable to the following problems: computing the projective transformations between point sets; estimating the trifocal tensor and determining the camera matrix of a projective camera using the DLT algorithm.

6.7 Properties of Fundamental Matrix

The essential and the fundamental matrixes have the following properties:

6.7.1 The fundamental matrix encapsulates both the intrinsic and the extrinsic parameters of the camera, whilst the essential matrix encapsulates only the extrinsic parameters[9]

6.7.2 If F is the fundamental matrix of the pair of cameras (P*;* P'), then F^T is the fundamental matrix of the pair in the opposite order: (P'*;* P) [10]

6.7.3 F has seven degrees of freedom: a 3 x 3 homogeneous matrix has eight independent ratios (there are nine elements, and the common scaling is not significant); however, F also satisfies the constraint det F = 0 which removes one degree of freedom.[10]

6.7.4 F is rank 2 homogeneous matrix.

6.7.5 Point correspondence: If m and m' are corresponding image points, then

$$m'^{T} F m = 0$$

6.7.6 Epipolar lines:

l'=Fm is the epipolar line corresponding to *m*

$l=F^T m'$ is the epioplar line corresponding to m'

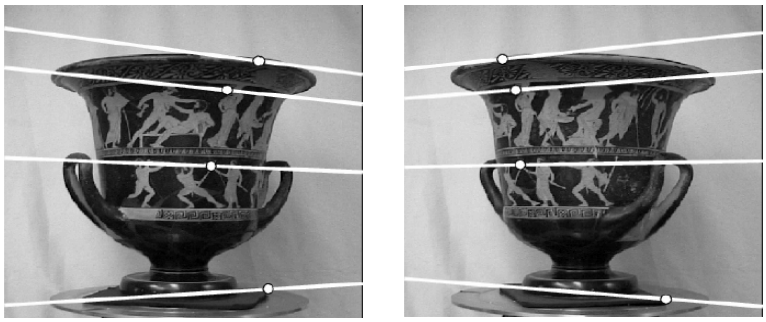


Figure 6.3 Corresponding Epipolar lines in a stereo pair[10]

6.7.7 Epipoles

$Fe=0$ (e is the right null space of F)

$e'^T F= 0$ (e' is the left null space of F)

6.7.8 F is a projective map taking a point to a line. In this case a point in the first image m defines a line in the second $l' = Fm$, which is the epipolar line of m. If l and l' are corresponding epipolar lines then any point m on l is mapped to the same line l'. This means there is no inverse mapping, and F is not of full rank. For this reason, F is not a proper correlation (which would be invertible).

6.8 Experimental results

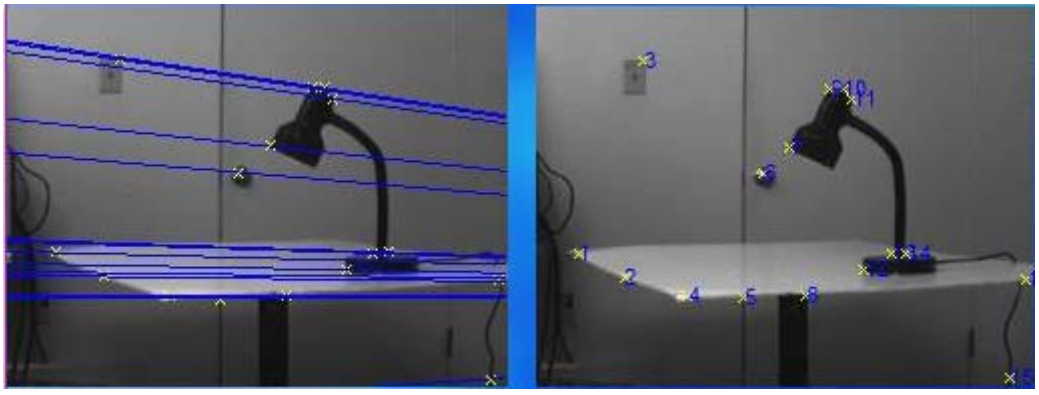


Figure 6.4 Epipolar lines on left image

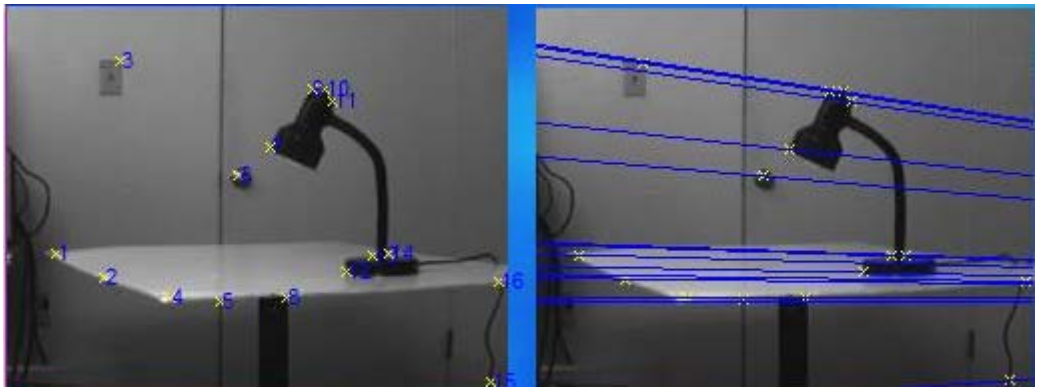


Figure 6.5 Epipolar lines on right image

6.9 Discussion

In figure 6.4 epipolar lines are shown on right image. For each and every highlighted point in left image there exists a line in right image on which its corresponding point must lie. That can be seen from above results.

Similarly for figure 6.5 epipolar lines for left image are shown on right image. These lines would help reduce the search space from 2D to 1D through a process called image rectification. As in this process a matrix is determined which when multiplied with point gives a line (epipolar) on other image. The very same matrix (fundamental matrix will be used as input to next step which is image rectification).

CHAPTER 7

IMAGE RECTIFICATION

7.1 Overview

Image rectification is an important component of stereo computer vision algorithms. As the epipolar geometry has been determined so the corresponding points between the two images must satisfy the so-called epipolar constraint. For a given point in one image, it is required to search for its correspondence in the

other image along an epipolar line. In general, epipolar lines are not aligned with coordinate axis and are not parallel. Such searches are time consuming since it is needed to compare pixels on skew lines in image space. These types of algorithms can be simplified and made more efficient if epipolar lines are axis aligned and parallel. This can be realized by applying 2D projective transforms, or *homographies*, to each image. This process is known as image rectification.

The pixels corresponding to point features from a rectified image pair will lie on the same horizontal scan-line and differ only in horizontal displacement. This horizontal displacement or *disparity* between rectified feature points is related to the depth of the feature. Seitz [12] has shown that distinct views of a scene can be *morphed* by linear interpolation along rectified scan-lines to produce new geometrically correct views of the scene. Zheng's [13] approach to rectification involves decomposing each homography into a projective and affine component. Then the projective component that minimizes a well defined projective distortion criterion is found, the affine component of each homography is decomposed into a pair of simpler transforms, one designed to satisfy the constraints for rectification, the other is used to further reduce the distortion introduced by the projective component. The rectified image points are defined as $\hat{m} = Hm$ and $\hat{m}' = H'm'$ while the fundamental matrix for the rectified pair is

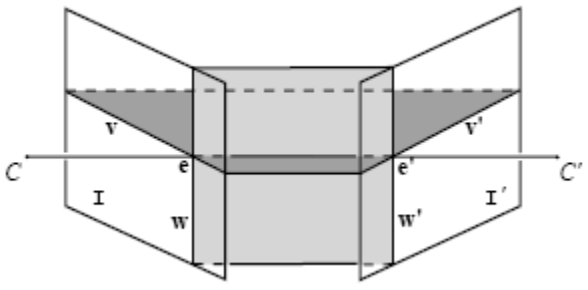


Figure 7.1 Lines on Common Epipolar plane.

$$\hat{F} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix} \tag{7.1}$$

So as per epipolar constraint

$$\hat{m}'^{T} \hat{F} \hat{m} = 0$$

Replacing $\hat{m}$ and $\hat{m}'$

$$m'^{T} H'^{T} \hat{F} H m = 0 \tag{7.2}$$

and so the *F* is defined as

$$F = H'^{T} \hat{F} H$$

Note that the homographies H and H' that satisfy above equation are not unique. The task is to find a pair of homographies H and H' that minimize image distortion. Let *u, v,* and *w* be lines equated to the rows of *H* such that

$$H = \begin{pmatrix} u^T \\ v^T \\ w^T \end{pmatrix} = \begin{pmatrix} u_a & u_b & u_c \\ v_a & v_b & v_c \\ w_a & w_b & 1 \end{pmatrix} \text{ and } H' = \begin{pmatrix} u'^T \\ v'^T \\ w'^T \end{pmatrix} = \begin{pmatrix} u'_a & u'_b & u'_c \\ v'_a & v'_b & v'_c \\ w'_a & w'_b & 1 \end{pmatrix} \tag{7.3}$$

Lines *v* and *v'*, and lines *w* and *w'* must be corresponding epipolar lines. H is decomposed into

$$H = H_a H_p \tag{7.4}$$

where H_p is the projective transform and H_a is the affine transform. H_p is defined as

$$H_p = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ w_a & w_b & 1 \end{pmatrix}$$

and the affine matrix as

$$H_a = HH_p^{-1} = \begin{pmatrix} u_a - u_c w_a & u_b - u_c w_b & u_c \\ v_a - v_c w_a & v_b - v_c w_b & v_c \\ 0 & 0 & 1 \end{pmatrix} \quad (7.5)$$

The affine transform H_a is further decomposed into

$$H_a = H_s H_r$$

where H_s is the shear transform and H_r is the similarity transform. The transform H_r has the form

$$H_r = \begin{pmatrix} v_b - v_c w_b & v_c w_a - v_a & 0 \\ v_a - v_c w_a & v_b - v_c w_b & v_c \\ 0 & 0 & 1 \end{pmatrix}$$

The shear transform H_s is defined as

$$H_s = \begin{pmatrix} s_a & s_b & s_c \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} s_a & s_b & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (7.6)$$

Note that H_s only affects the x coordinate of a point i.e. the column no so it will not affect the rectification.

Now how to compute each component transforms just defined.

7.2 Projective Transform

The transforms H_p and H'_p completely characterize the projective components of *H* and *H'*. These transforms map the epipoles *e* and *e'* to points at infinity (points with *w*-coordinate equal to zero) so make the epipolar lines parallel. By definition, H_p and H'_p are determined by lines *w* and *w'* respectively. The lines *w* and *w'* are not independent. Given a direction $\mathrm{z} = [\lambda \ \mu \ 0 \ ]^{\mathrm{T}}$ in image *I*, it comes as:

$$\mathrm{w}=[\mathrm{e}]_{\mathrm{x}}\mathrm{z} \qquad (7.7)$$

and

$$\mathrm{w'} = F\mathrm{z}$$

where $[\ ]_{\mathrm{x}}$ denotes the antisymmetric matrix. So $[e]_x$ denotes the antisymmetric matrix of the epipole.

The relationship between epipole and the fundamental matrix is

$$Fe = 0$$

As *F* has the rank 2, it follows that the epipole, *e*, ia the null space of *F* and similarly *e'* is the null space of F^T. So *e* can be found by singular value decomposition of *F*.

$$F = UDV^T$$

So *e* is the column of *V* corresponding to the null singular value and *e'* is the column of *U* corresponding to the null singular value.

Any such *z* will define a pair of corresponding epipolar lines *w* and *w'*. The objective is to find *z* that minimizes the distortion defined below.

Let $p_i = [p_{i,x}, p_{i,y}, 1]^T$ be a point in the original image. This point will be transformed by H_p to point $[\frac{p_{i,x}}{w_i}, \frac{p_{i,y}}{w_i}, 1]^T$ with weight $w_i = w^T p_i$. If the weights assigned to points are identical then there is no projective distortion and the homography is necessarily an affine transform. In order to map the epipole *e* from the affine (image) plane to a point at infinity H_p cannot in general be affine. However, as the image is bounded an attempt is to make H_p as affine as possible. This serves as basis of distortion minimization criterion.

7.2.1 Distortion Minimization Criteria

Although having identical weights in general is not possible (except when the epipole is already at infinity), but it is tried to minimize the variation of the weights assigned to a collection of points over both images. All the pixels from both images are used as our collection, but some other subset of important image points can also be used if necessary. The variation is measured with respect to the weight associated with the image center. More formally, computation is:

$$\sum_{i=1}^{n} \left[\frac{w_i - w_c}{w_c} \right]^2$$

Where $w_i = w^T p_i$, $w_c = w^T p_c$ and $p_c = \frac{1}{n} \sum_{i=1}^{n} p_i$ is the average of all the points. In the matrix form the above equation can be rewritten as

$$\frac{w^T P P^T w}{w^T p_c p_c^T w} = \frac{([e]_x z)^T P P^T [e]_x z}{([e]_x z)^T p_c p_c^T [e]_x z} = \frac{(z^T [e]_x^T) P P^T [e]_x z}{(z^T [e]_x^T) p_c p_c^T [e]_x z} \qquad (7.8)$$

where *P* is a 3xn matrix

$$P = \begin{pmatrix} p_{1,x} - p_{c,x} & p_{2,x} - p_{c,x} & \cdots & p_{n,x} - p_{c,x} \\ p_{1,y} - p_{c,y} & p_{2,y} - p_{c,y} & \cdots & p_{n,y} - p_{c,y} \\ 0 & 0 & \cdots & 0 \end{pmatrix}$$

For both the images :

$$\frac{z^T [e]_x PP^T [e]_x z}{z^T [e]_x p_c p_c^T [e]_x z} + \frac{z^T F^T P' P'^T F z}{z^T F^T p'_c p'^T_c F z} = \frac{z^T A z}{z^T B z} + \frac{z^T A' z}{z^T B' z} \qquad (7.9)$$

Where A, B, A', B' are 3 x 3 matrices. ***z*** is defined as $z = (\lambda \quad \mu \quad \nu)^T$ Since the ***v*** coordinate of z is equal to zero as the direction is to infinity only the upper-left 2 x 2 blocks of the matrices A, B, A', B' are important. So *A, B, A'* and *B'* are found and upper 2 x 2 block is extract for further calculations. Denote $z = [\lambda \ \mu \ ]^T$ and as it is defined up to a scalar factor so without loss of generality set μ=1. Because of scaling it is only λ which is important to be found. So taking $z = (\lambda \quad 1)^T$ and putting in above equation and then solve for λ by minimizing the above equation, which is a nonlinear optimization problem. Values of λ are found for which

$$\frac{d}{d\lambda}\left(\frac{z^T A z}{z^T B z} + \frac{z^T A' z}{z^T B' z}\right)$$

is equal to zero. Then the value of λ, which gives the minimum value for, $\frac{z^T A z}{z^T B z} + \frac{z^T A' z}{z^T B' z}$ is the desired criteria. The matrices PP^T and $p_c p_c^T$ are:-

$$PP^T = \frac{wh}{12}\begin{pmatrix} w^2 - 1 & 0 & 0 \\ 0 & h^2 - 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \qquad (7.10)$$

and

$$p_c p_c^{\,T} = \frac{1}{4}\begin{pmatrix} (w-1)^2 & (w-1)(h-1) & 2(w-1) \\ (w-1)(h-1\_ & (h-1)^2 & 2(h-1) \\ 2(w-1) & 2(h-1) & 4 \end{pmatrix}$$

w and *w'* is found as:

$$\mathrm{w}=[\mathrm{e}]_{\mathrm{x}}\mathrm{z} \quad \text{and} \quad \mathrm{w}' = Fz$$

and the projective transformation matrices are

$$H_p = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ w_a & w_b & 1 \end{pmatrix} \text{ and } H_p^{'} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ w_a^{'} & w_b^{'} & 1 \end{pmatrix}$$

7.3 Similarity Transform

The transforms H_p and $H^{'}_{p}$ were found that map the epipoles *e* and *e'* to points at infinity. Let's define a pair of similarity transforms H_r and $H^{'}_{r}$ that rotate these points at infinity into alignment with the direction $\begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^T$ as required for rectification so that the epipolar lines are then horizontally aligned. Additionally, a translation in the y-direction on one of the images is found to exactly align the scan-lines in both images. As

$$H_r = \begin{pmatrix} v_b - v_c w_b & v_c w_a - v_a & 0 \\ v_a - v_c w_a & v_b - v_c w_b & v_c \\ 0 & 0 & 1 \end{pmatrix}$$

Let's assume that the lines *w* and *w'* are known. v_a And v_b from above equation can be eliminated by making use of the following:

$$F = H^{'T}\hat{F}H$$

$$F = \begin{pmatrix} v_a w'_a - v'_a w_a & v_b w'_a - v'_a w_b & v_c w'_a - v'_a \\ v_a w'_b - v'_b w_a & v_b w'_b - v'_b w_b & v_c w'_b - v'_b \\ v_a - v'_c w_a & v_b - v'_c w_b & v_c - v'_c \end{pmatrix} \qquad (7.11)$$

From the last row of the matrix gives

$$v_a = f_{31} + v'_c w_a$$
$$v_b = f_{32} + v'_c w_b$$
$$v_c = f_{33} + v'_c$$

So finally H_r and H'_r are

$$H_r = \begin{pmatrix} f_{32} - w_b f_{33} & w_a f_{33} - f_{31} & 0 \\ f_{31} - w_a f_{33} & f_{32} - w_b f_{33} & f_{33} + v'_c \\ 0 & 0 & 1 \end{pmatrix} \text{ and } H'_r = \begin{pmatrix} w'_b f_{33} - f_{23} & f_{13} - w'_a f_{33} & 0 \\ w'_a f_{33} - f_{13} & w'_b f_{33} - f_{23} & v'_c \\ 0 & 0 & 1 \end{pmatrix}$$

There is a translation term involving v'_c which aligns the scan lines.

7.4 Shearing Transform

The freedom afforded by the independence of *x* and *x'*(column nos) is exploited to reduce the distortion introduced by the projective transforms H_p and H'_p. The

effect of x as the shearing transform is modeled.

$$H_s = \begin{pmatrix} a & b & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Select 4 points on the image which are on the midpoints of the edges of the image *I.* i.e if *w* is the width and *h* is the height of the image then the points are

$$
\begin{aligned}
a &= \left[\frac{w-1}{2}, 0, 1\right]^T \\
b &= \left[w-1, \frac{h-1}{2}, 1\right]^T \\
c &= \left[\frac{w-1}{2}, h-1, 1\right]^T \\
d &= \left[0, \frac{h-1}{2}, 1\right]^T
\end{aligned}
\tag{7.12}
$$

Let $\hat{a} = H_r H_p a$ be a point in the affine plane by dividing through so that $\hat{a}_z = 1$ i.e

$\hat{a} = \left[\frac{\hat{a}_x}{\hat{a}_z}, \frac{\hat{a}_y}{\hat{a}_z}, \frac{\hat{a}_z}{\hat{a}_z}\right]^T$ .Similarly define $\hat{b}, \hat{c}, \hat{d}$. It is tried to preserve the perpendicularity and aspect ratio of the lines $\overline{bd}$ and $\overline{ca}$.

$$
\begin{aligned}
\hat{x} &= \hat{b} - \hat{d} \\
\hat{y} &= \hat{c} - \hat{a}
\end{aligned}
\quad \text{i.e} \quad
\begin{aligned}
\hat{x} &= (\hat{x}_x, \hat{x}_y) \\
\hat{y} &= (\hat{y}_x, \hat{y}_y)
\end{aligned}
$$

So the real solution is

$$
a = \frac{h^2 \hat{x}_y{}^2 + w^2 \hat{y}_y{}^2}{hw(\hat{x}_y \hat{y}_x - \hat{x}_x \hat{y}_y)} \text{ and } b = \frac{h^2 \hat{x}_x \hat{x}_y + w^2 \hat{y}_x \hat{y}_y}{hw(\hat{x}_y \hat{y}_x - \hat{x}_x \hat{y}_y)} \tag{7.13}
$$

7.5 Experimental results:

Figure 7.2 Projective Transformation

Figure 7.3 Similarity Transform

Figure 7.4 Shear Transform

7.6 Discussion

Above three transformations reduce 2D search to 1D as both the images come on same plane. This rectification process has main advantage that it removes projective distortions from image which are present in images when they are taken. Without this process 3D reconstruction is possible but it will be computationally intensive because of 2D search in whole image. Similarity Transform is as such not needed but gives more strength to results if done.

CHAPTER 8

DISPARITY AND DEPTH MAPS

8.1 Overview

Computation of disparity or parallax between the 2 images was our main task of the project as it leads to the depth or elevation information. All the modules through which the stereo pair has passed through have been just to ensure accuracy of computing the disparity. As already defined that by disparity it is meant that if a point is in one image then how much this point has moved in the second image and this movement is only in terms of column coordinates as the horizontal shift has the disparity information. Various techniques exist for this trivial task and some of them, which are the state of art today, are discussed below.

8.2 Sum of Square Differences (SSD)

A template is picked around the pixel of interest from the first image which may be a window of 5x5 or 7x7 or any desirable size. This template is now to be matched with a region in the second image so from that region template of the same size as the template of the first image is picked and then sum of square differences operation is performed.

$$\sum (I_{x,y} - I'_{x',y'})^2 \qquad (8.1)$$

Solving this equation leads to

$$\sum I^2{}_{x,y} + I'^2{}_{x',y'} + 2I_{x,y}I'_{x',y'}$$

$I^2{}_{x,y}$ and $I'^2{}_{x',y'}$ will be positive numbers and will just add to the sum so are not much significant in template matching. The last term $2I_{x,y}I'_{x',y'}$ is of more importance but again the multiplication factor of 2 is not contributing to the matching criteria. So the real part that contributes to the matching criteria is

$$I_{x,y}I'_{x',y'}$$

which leads to the approach of correlation.

8.3 Fast Normalized Cross-Correlation

Normalized cross correlation is computed in the spatial domain for finding the disparity maps. Unfortunately the normalized form of correlation (correlation coefficient) preferred in template matching does not have a correspondingly simple and efficient frequency domain expression. For this reason normalized cross-correlation has been computed in the spatial domain. Due to the computational cost of spatial domain convolution, several inexact but fast spatial domain matching methods have also been developed.

8.3.1 Template Matching by Cross-Correlation

The use of cross-correlation for template matching is motivated by the distance measure (squared Euclidean distance):

$$d^2_{I,t}(x_1, y_1) = \sum_{x,y} \left[I(x,y) - t(x - x_1, y - y_1) \right] \qquad (8.2)$$

(where *I* is the image and the sum is over *x*, *y* under the window containing the feature *t* positioned at x_1, y_1). In the expansion of d^2

$$d^2_{I,t}(x_1, y_1) = \sum_{x,y} \left[I^2(x,y) - 2 f(x,y) t(x - x_1, y - y_1) + t^2(x - x_1, y - y_1) \right]$$

the term $\sum t^2(x - x_1, y - y_1)$ is constant. If the term $\sum I^2(x,y)$ is approximately constant then the remaining cross-correlation term

$$c(x_1, y_1) = \sum_{x,y} I(x,y) t(x - x_1, y - y_1) \qquad (8.3)$$

is a measure of the similarity between the image and the feature. There are several disadvantages to using this for template matching:

- If the image energy $\sum I^2(x,y)$ varies with position, matching using above equation can fail. For example, the correlation between the feature and an

exactly matching region in the image may be less than the correlation between the feature and a bright spot.

- The range of $c(x_1,y_1)$ is dependent on the size of the feature.
- Above equation is not invariant to changes in image amplitude such as those caused by changing lighting conditions across the image sequence.

8.3.2 Normalized Cross Correlation

The correlation coefficient overcomes these difficulties by normalizing the image and feature vectors to unit length, yielding a cosine-like correlation coefficient

$$\gamma(x_1, y_1) = \frac{\sum_{x,y}[f(x,y) - \bar{I}_{x_1,y_1}][t(x-x_1, y-y\ ) - \bar{t}]}{\sqrt{\{\sum_{x,y}[I(x,y) - \bar{I}_{x_1,y_1}]^2 \sum_{x,y}[t(x-x_1, y-y\ ) - \bar{t}]^2\}}} \qquad (8.4)$$

where $\bar{t}$ is the mean of the feature and $\bar{I}_{x_1,y_1}$ is the mean of $I_{x,y}$ in the region under the feature. This equation is referred to as *normalized cross-correlation*.

8.3.3 Disadvantages of Normalized Cross Correlation

Because of it's over sensitivity to the slight changes in the template, it gives errors in the disparity maps. So the sum of square differences is more advocated for template matching.

8.4 Sum of Absolute Differences (SAD)

The implementation is similar to the SSD approach but the function that is computed for both the templates is

$$\sum (I_{x,y} - I'_{x',y'})$$

The accuracy of results varied as window size was changed. However more accurate results were achieved as window size of 5x5 or 7x7 was used. An increase in window size beyond this resulted into missing matches and smaller details while a decrease in window size resulted in more wrong matches.

8.5 Dynamic Programming

It is reasonable to assume that the order of matching features along a pair of epipolar lines is the inverse of the order of the corresponding surfaces attributes along the curve where the epipolar plane intersects the observed object's boundary this is so called the *ordering constraint.*

INTERESTINGLY ENOUGH, IT MAY NOT BE SATISFIED BY REAL SCENES, IN PARTICULAR WHEN SMALL SOLIDS OCCLUDE PARTS OF LARGER ONES OR, MORE RARELY AT LEST IN ROBOT VISION, WHEN TRANSPARENT OBJECTS ARE INVOLVED.

DESPITE THESE RESERVATIONS, THE ORDERING CONSTRAINTS REMAIN A REASONABLE ONE, AND IT CAN BE USED TO DEVICE EFFICIENT ALGORITHMS RELYING ON *DYNAMIC PROGRAMMING* TO ESTABLISH STEREO CORRESPONDENCES. LET US ASSUME THAT A NUMBER OF FEATURE POINTS HAVE BEEN FOUND ON CORRESPONDING EPIPOLAR LINES. OUR OBJECTIVE HERE IS TO MATCH THE INTERVAL SEPARATING THOSE POINTS ALONG THE TWO INTENSITY PROFILES. ACCORDING TO THE *ORDERING CONSTRAINT* THE ORDER OF THE FEATURE POINTS MUST BE THE SAME ALTHOUGH THE OCCASIONAL INTERVAL IN EITHER IMAGERY MAY BE REDUCED TO A SINGLE POINT

CORRESPONDING TO MISSING CORRESPONDENCES ASSOCIATED WITH OCCLUSION AND/OR NOISE.

8.6 Graph Cuts

Reconstructing an object's 3-dimensional shape from a set of cameras is a classic vision problem. In the last few years, it has attracted a great deal of interest, partly due to a number of new applications both in vision and in graphics that require good reconstructions. While the problem can be viewed as a natural generalization of stereo, it is considerably harder. The major reason for this is the difficulty of reasoning about visibility. In stereo matching, most scene elements are visible from both cameras, and it is possible to obtain good results without addressing visibility constraints. In the more general scene reconstruction problem, however, very few scene elements are visible from every camera, so the issue of visibility cannot be ignored.

The origin of Graph Cuts in image processing was for image segmentation. Segmentation is a very fundamental problem in vision. Intuitively, segmentation is to group up similar components such as image pixels, image regions or even video clips. However, this problem becomes very complicated when it is difficult to define the similarity measurements, e.g., define similarity in terms of intensity, color, texture or motion , and when people are ambitious to expect some semantics from segmentation, e.g., Segmentation of people from image is required. Image segmentation is to group up similar pixels together to form a set of coherent image regions, given a single image. The pixel similarity could be measured based on the consistency of location, intensity, color, and texture of different pixels. Generally, these elements can be compound together to represent an image pixel, or use some of them. For example, only use color components or use both location and intensities.

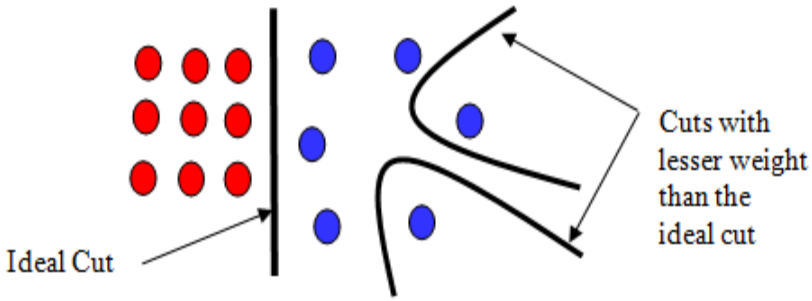


Figure 8.1 The Ideal Graph Cut

Segmentation can be done by by clustering, graph cuts and by EM algorithms

As per graph theory, each pixel in the image is connected to its neighbors. This results in a grid-like graph representation of the image. Each link between two pixels is given a weight; the weight is high if the pixels are similar and low if they differ. One (or several) pixels are marked as object and one (or several) pixels are marked as background. A cut of minimum cost surrounding the object pixels is found using minimum graph cut theory. The image analysis part is to make a good choice of weights. The cut defines the boundary of the object .

Ramin Zabih's approach of Graph Cuts [15] looks at the scene reconstruction problem from the point of view of energy minimization. Energy minimization has several theoretical advantages, but has generally been viewed as too slow for early vision to be practical. This approach is motivated by some recent work in early vision, where fast energy minimization algorithms have been developed based on graph cuts [15, 16]. The energy that is minimize has three important properties.It treats the input images symmetrically. Secondly it handles visibility properly and it imposes spatial smoothness while preserving discontinuities.

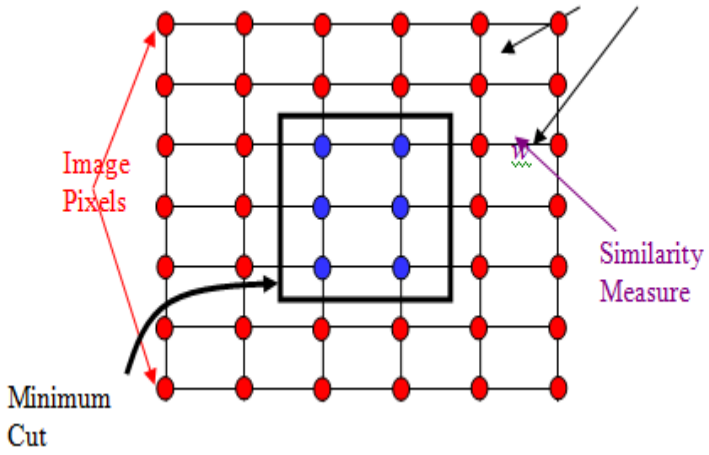


Figure 8.2 Image Segmentation using Graph Cuts

The problem of reconstructing a scene from multiple cameras has received a great deal of attention in the last few years. One extensively explored approach to this problem is voxel occupancy. In voxel occupancy [17, 18] the scene is represented as a set of 3-dimensional voxels, and the task is to label the individual voxels as filled or empty. Voxel occupancy is typically solved using silhouette intersection, usually from multiple cameras but sometimes from a single camera with the object placed on a turntable. It is known that the output of silhouette intersection even without noise is not the actual 3-dimensional shape, but rather an approximation called the visual hull [19].

One major limitation of voxel coloring and space carving is that they lack a way of imposing spatial coherence. This is particularly problematic because the image data is almost always ambiguous. Another (related) limitation comes from the fact that these methods traverse the volume making “hard” decisions concerning the occupancy of each voxel they analyze. Because the data is ambiguous, such a decision can easily be incorrect, and there is no easy way to undo such a decision later on. Now energy function will be defined that is to be minimized. It will consist of three terms:

$$E(f) = E_{data}(f) + E_{smoothness}(f) + E_{visibility}(f) \qquad (8.5)$$

The data term will impose photo-consistency(i.e the pixels corresponding to same 3D point have similar pixel intensity). It is

$$E_{data}(f) = \sum_{(p,f(p)),(q,f(q))\in I} D(p,q) \tag{8.6}$$

where *D*(*p, q*) is a non-positive value depending on intensities of pixels *p* and *q*. It can be, for example,

$$D(p,q) = \min\{0, (Intensity(p) - Intensity(q))^2 - K\} \tag{8.7}$$

for some constant *K* >0.

Since the energy is minimized, terms *D*(*p, q*) that are summed up will be small. These terms are required to be non-negative. Thus, pairs of pixels *p*, *q* which come from the same scene point according to the configuration *f* will have similar intensities, which cause photo-consistency. The smoothness term involves a notion of neighborhood; Assumption is made that there is a neighborhood system on pixels

$$N \subset \{\{p,q\} \,|\, p,q \in P\} \tag{8.8}$$

This can be the usual 4-neighborhood system: pixels $p = (p_x, p_y)$ and $q = (q_x, q_y)$ are neighbors if they are in the same image and $|p_x - q_x| + |p_y - q_y| = 1$. Smoothness term is written as:

$$E_{smoothness}(f) = \sum_{\{p,q\}\in N} V_{p,q}(f(p), f(q)) \tag{8.9}$$

Term *V{p,q} is required* to be a metric. This imposes smoothness while preserving discontinuities, as long as an appropriate robust metric is picked. For example, the robustified *L*1 distance $V(l_1, l_2) = \min(|l_1 - l_2|, K)$ for constant *K can be used*.

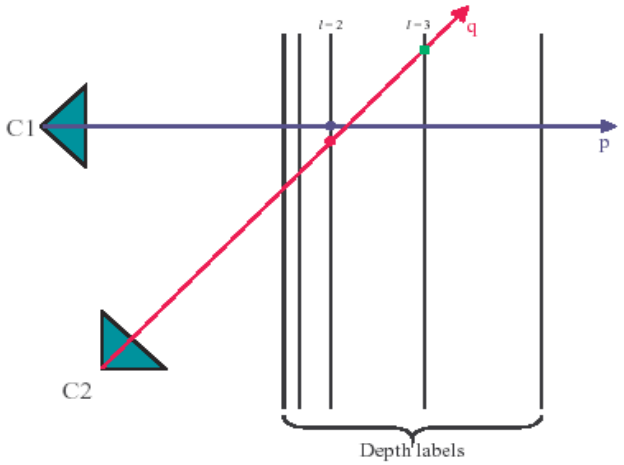


Figure 8.3 Example of pixel interactions.

There is a photo-consistency constraint between the red round point and the blue round point, both of which are at the same depth (*l* = 2). The red round point blocks camera C2's view of the green square point at depth *l* = 3

The last term will encode the visibility constraint (taking care of occlusion): it will be zero if this constraint is satisfied and infinity otherwise. This can be written using another set of interactions I_{vis} *w*hich contains pairs of 3D-points violating the visibility constraint:

$$E_{visibility}(f) = \sum_{(p,f(p)),(q,f(q)\in I_{vix}} \infty \qquad (8.11)$$

The set I_{vis} is required to meet following condition:

Only 3D-points at different depths can interact, i.e. if $\{(p_1,l_1),(p_2,l_2)\} \in I_{vis}$ then $l_1 = l_2$.

The visibility constraint says that if a 3D-point (*p, l)* is present in a configuration *f* (i.e. *l* = *f*(*p*)) then it "blocks" views from other cameras: if a ray corresponding to a pixel *q* from the other image goes through (or close to) (*p, l)* then its depth is at most *l* because it is the pixel *p* which is occluding the pixel *q*. An example of our

problem formulation in action is shown in figure 1. There are two cameras C1 and C2. There are 5 labels shown as black vertical lines. As in our current implementation, labels are distributed by increasing distance from a fixed camera. Two pixels, *p* from C1 and *q* from C2 are shown, along with the red round 3D-point (*q,* 2) and the blue round 3D-point (*p,* 2). These points share the same label, and interact (i.e., *{(p,* 2)*, (q,* 2)*} . I*). So there is a photoconsistency term between them. The green square point (*q,* 3) is at a different label (greater depth), but is behind the red round point. The pair of 3D-points *{ p,* 2)*, (q,* 3)*}* is in I_{vis}. So if the ray *p* from camera C1 sees the red round point (*p,* 2), the ray *q* from C2 cannot see the green square point (*q,* 3). Zabih's approach is to construct an approximation algorithm based on graph cuts that finds a strong local minimum.

8.6.1 Graph construction

How to efficiently minimize *E* among all configurations using graph cuts i.e select different configurations and minimize *E* for each and the configuration for which *E* is minimized is selected. The output of this method will be a local minimum in a strong sense. In particular, consider an input configuration f and a disparity α. Another configuration f' is defined to be within a single α*-expansion* of *f* when for all pixels $p \in P$ either *f'*(*p*) = *f*(*p*) or *f'*(*p*) = α . This notion of an expansion forms the basis for several very effective stereo algorithms.This algorithm is very straightforward; it simply selects (in a fixed order or at random) a disparity α , and find the unique configuration within a single α-expansion move (our local improvement step). If this decreases the energy, then go there; if there is no α that decreases the energy, its done. One restriction on the algorithm is that the initial configuration must satisfy the visibility constraint. This will guarantee that all subsequent configurations will satisfy this constraint as well, since energy is minimized, and configurations that do not satisfy the visibility constraint have infinite energy. The critical step in this method is to efficiently compute the α-expansion with the smallest energy.

8.6.2 Experimental Results

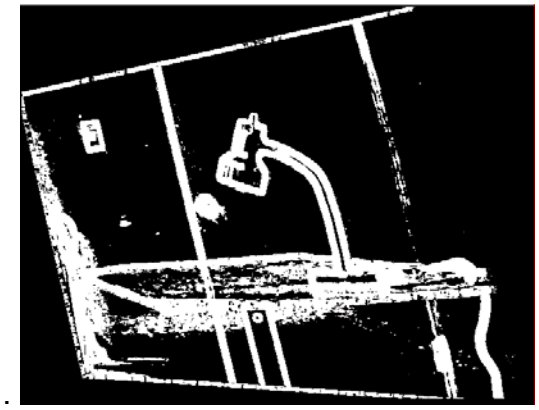


Figure 8.4 Absolute Disparity map

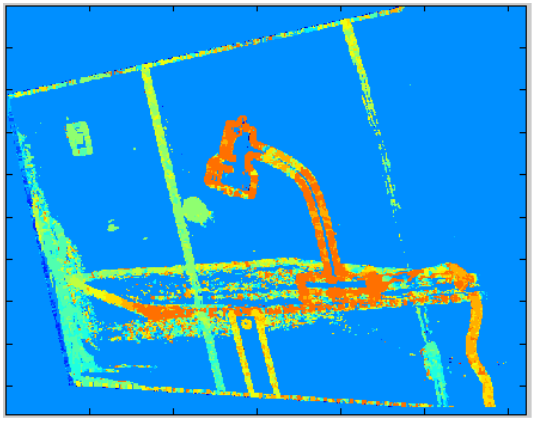


Figure 8.5 Disparity map

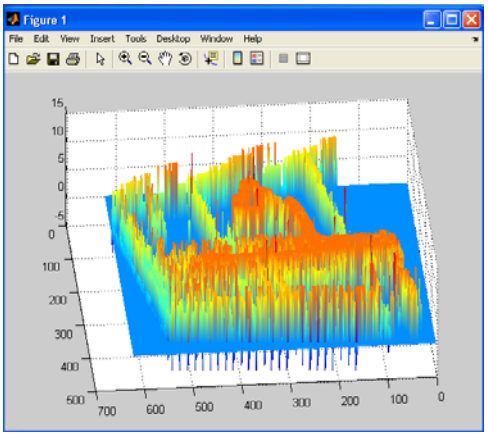


Figure 8.6 Depth map

8.7 Discussion

Figure 8.4 shows the absolute disparity map of stereo pair. Figure 8.5 shows disparity map and then finally depth is extracted in Figure 8.6. The more the disparity greater is the depth and vice versa. In figure 8.6 the more intense regions show more relative depth in reference to some plane. This particular result is achieved using Cross correlation technique. Hence it is seen that depth can be extracted from stereo pair by following some steps. If exact dimensions of the object are required, then camera parameters like orientation, interior and exterior parameters are required. Anyhow it can be seen that successful depth extraction system has been implemented.

Research work includes feature extraction, image matching to find correspondences, epipolar geometry, image rectification and finally disparity and depth calculation.

References:

1. Shi and C. Tomasi (June 1994). "Good Features to Track
2. C. Harris and M. Stephens (1988). "A combined corner and edge detector"
3. Lowe, David G. (1999). "Object recognition from local scale-invariant features”
4. Lowe, D.G. (2004), “Distinctive Image Features from Scale-Invariant Key points”
5. S. Ullman. The interpretation of Visual Motion. MIT Press, Cambridge, MA,1979.
6. D. Marr and T. Poggio. A computational theory of human stereo vision. Proc.Royal Society London, B 204:301{328, 1979.

7. G. Scott and H. Longuet-Higgins. An algorithm for associating the features of two patterns. In Proc. Royal Society London, volume B244, pages 21{26, 1991}

8. http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/OWENS/LECT10/node3.html

9. http://www.robots.ox.ac.uk/~vgg/hzbook/hzbook1/HZepipolar.pdf

10. http://lear.inrialpes.fr/people/triggs/pubs/isprs96/node47.html

11. http://ai.stanford.edu/~mitul/cs223b/fm.html

12. Steven M. Seitz Charles R. Dyer, “View Morphing”, Department of Computer Sciences, University of Wisconsin—Madison1

13. Z. Zhang, “Computing Rectifying Homographies for Stereo Vision”, Microsoft Research, June, 2001

14. Vladimir Kolmogorov and Ramin Zabih, “Multi-camera Scene Reconstruction via “Graph Cuts”, Computer Science Department, Cornell University, Ithaca, NY 14853

15. Yuri Boykov, Olga Veksler, and Ramin Zabih. Markov Random Fields with efficient approximations. In IEEE Conference on Computer Vision and Pattern Recognition, pages 648–655, 1998.

16. R. Szeliski. Rapid octree construction from image sequences. Computer Vision, Graphics and Image Processing, 58(1):23–32, July 1993.

17. W.N. Martin and J.K. Aggarwal. Volumetric descriptions of objects from multiple views. IEEE Transactions on Pattern Analysis and Machine Intelligence, 5(2):150–158, March 1983.

18. A. Laurentini. The visual hull concept for silhouette-based image understanding. IEEE Transactions on Pattern Analysis and Machine Intelligence, 16(2):150–162, February 1994.

19. Richard I. Hartley, ”In Defense of the Eight-Point Algorithm”, IEEE Transactions On Pattern Analysis And Machine Intelligence, Vol. 19, No. 6, June 1997