3D object reconstruction using structured light and two 2D images

by

Philippe Lavoie

A thesis submitted to the
School of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Applied Science

Ottawa-Carleton Institute for Electrical Engineering
Department of Electrical Engineering
Faculty of Engineering
University of Ottawa
December 1995

National Library
of Canada

Acquisitions and
Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale
du Canada

Direction des acquisitions et
des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

ISBN 0-612-16417-9

Canada

UNIVERSITÉ D'OTTAWA
UNIVERSITY OF OTTAWA

ABSTRACT

With the advent of new technologies, stereo vision became an important field of research for both machine vision and graphic applications. For machine vision and for graphical representation of real objects in a computer environment, the three-dimensional (3D) reconstruction of the image of a real object became a key technique. A few methods such as range finding, which use laser scanners, computer tomography, based on CTR or MRI machines, computational stereo, etc. were developed.

Computational stereo is broadly defined as the recovery of 3D characteristics of a scene from a series of images obtained from different points in the three dimensional (Euclidean) space.

In this thesis, a new algorithm and system is introduced and developed for the 3D reconstruction of the images of real objects from two 2D images acquired with two cameras. The algorithm is based on a new matching method, a new procedure for the determination of the fundamental matrix used in stereo vision, and a new technique for stereo fusion. The novelty of the matching procedure, and of the determination of the camera alignment, consists of the projection of a structured light pattern on the real object, the pattern being created using a pseudo-random encoded mesh (PRBA) [34].

The novelty of the stereo fusion algorithm consists of the application of the dynamic programming principle (DP) [5] [27] using a cost function which contains the information obtained from the list of matched points. It is also proposed to use an autoregressive (AR) modeling technique for calculating the stereo disparity of each pixel of the two images. The autoregressive filter helps the DP part of the algorithm to calculate the disparity of the pixels when the above pixels are occluded. The above proposed methods offer three distinctive advantages over a conventional stereo system:

- It easily generates a list of matching points.
- It adds structure to an object without textures.
- It is less computational intensive.

ACKNOWLEDGMENT

I would like to thank my supervisor Dr. Dan Ionescu an my co-supervisor Dr. Emil Petriu for their guidance and continuous support and my family for their love and patience.

Contents

List of Figures

List of Tables

LIST OF ABBREVIATIONS

Various abbreviations used frequently in this thesis are summarized below. All notations are defined in the text when they first appear.

2D	Two Dimensions
3D	Three Dimensions
AR	Autoregressive
CTR	Computer Tomography Radiosity
DP	Dynamic Programming
LMedS	Least Median of Squares
MRI	Magnetic Resonance Imaging
PRBA	Pseudo Random Binary Array
PRBS	Pseudo Random Binary Sequence
SVD	Singular Value Decomposition

LIST OF SYMBOLES

Various symboles are oftened used in this text. A list of these is given below.

A	The intrinsic matrix
F	The fundamental matrix
m	The vector location of a point in an image
M	The vector location of a point in 3D space
R	The rotation matrix
t	The translation vector

Chapter 1

Introduction

It is possible to reconstruct a three dimensional (3D) object using only the information contained in a two bi-dimensional (2D) images of the same real object. The analysis of the content of video images in stereo has emerged as an important passive method to extract the 3D representation of a scene. This method requires only 2 cameras and, therefore, can be a low cost technique to observe objects from a 3D perspective. The structured light method is an alternative low cost possibility. The presentation of a method which uses both techniques to recreate a 3D object is the purpose of this thesis.

The 3D content of a scene can be recreated using stereo images. In stereoscopy, the depth is estimated by using triangulation. The estimation process requires the knowledge of the global position and orientation of each camera, the model of the camera and the correspondence between all the same feature points in both images. The general stereo process for 3D reconstruction involves the following three steps: preprocessing, establishing correspondences and determining the depth.

The preprocessing involves the finding of the global position and orientation of each camera, and the calibration of the cameras. The establishment of a correspondence between points from 2 images is traditionally done using two different methods: area-based and feature-based matching.

Each method usually uses the epipolar constraint to limit the search space. The epipolar constraint implies that the points residing on an epipolar line of the left image will also be present on the corresponding epipolar line of the right image. This limits the search space to one dimension.

The correspondence between each pixel of both images gives their disparity. It corresponds to the distance between two corresponding pixels. This distance is used in the triangulation process to reconstruct the depth of the image perceived.

(a) The stereo fusion method

(b) The structured light method

Figure 1.1: The similarities between the stereo fusion method and the structured light method.

The other method for 3D reconstruction using a camera is done with the help of structured lights. A structured light is the projection of a pattern onto a scene. The pattern can be made out of dots, lines or grids. This method can be viewed as a stereoscopic process where one of the camera is active (the projector) and the other one is passive (the camera). The similarity between the stereo fusion method and the structured light method can be seen in Figure 1.1.

When using the structured light method, the precision is limited to the pattern being projected. It is also limited to indoor situations where such a light can be projected.

A match between points from the structured light and the camera view of it can lead to ambiguities. The order in which the dots, strips or grid are projected might not be perceived identically by the camera.

The ambiguities in finding a correspondence between the image and the structured light can be removed by encoding the projected pattern. Different methods have been proposed to encode the structured light such as grid node identification by space encoding of projected rays [36] [20], color indexing of the grid lines [6], grid line labeling by thickness [20], grid line identification by pseudo random binary sequences (PRBSs) encoding [43], or by grid points identification using pseudo random binary arrays (PRBAs) encoding [41].

From a complexity point of view, the encoded structured light method is much simpler than the stereoscopy method when trying to establish a correspondence between points. However, the

stereoscopy method gives a matching between all the points of both images and is not limited to the points seen from the structured light. A logical step would be to mix these two methods to simplify the matching problem while keeping the ability to find a correspondence between all the pixels of the images. This is the approach developed and proposed in this thesis.

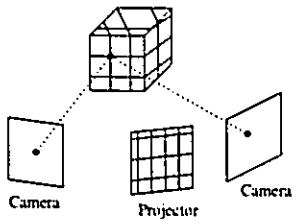


Figure 1.2: The setup proposed by the thesis.

The proposed setup is illustrated in Figure 1.2. It uses 2 cameras to capture the left and right images of the same object. The correspondence between points is done using stereo fusion and a structured light illuminates the object of interest to help finding a correspondence between points.

The computing complexity required by the preprocessing step is greatly reduced by illuminating the object. The correspondence between some of the points contained in the left and right images is required to determine the global or the relative orientation of both cameras. Traditionally, this is done by either one of the following methods:

1. Calibrating the position of the cameras with a known object: the object must contain easily recognizable features, or

2. Complex search in 2D space from both images to find correspondences. This method usually involves correlations of areas around the points of interest. Edge points and corner points are frequently used as points of interest.

Eventually, method 1 has to be used if a complete model of the 3D world has to be generated (global orientation) . Method 2 is sufficient if a scaled version is acceptable (relative orientation).

The structured light has the advantage of simplifying both method 1 and 2. In each step a matching between feature points must be achieved. The structured light provides easily recognizable feature points.

Another problem faced by the preprocessing step is the calibration of the cameras: finding their focal points and their internal parameters. These parameters along with the epipolar geometry of the two cameras can be determined by estimating the *fundamental matrix* F (see Section 4.2).

The method proposed here for 3D reconstruction involves the following steps:

- Projection of an encoded structured light using a pseudo random binary array (PRBA).
- Estimation of the F matrix using the feature points perceived from the projected structured light. The estimation process involves a non-linear *least median of squares* (LMedS) optimization.
- Epipolarization of the images. Traditionally an epipolarization involves the projection of an image onto an epipolar plane. In Section 4.3, a discussion on why this approach can be problematic is given and a solution is offered. The epipolarization operation is necessary since the matching process assumes the validity of the epipolar constraint.
- Stereo fusion of both images using dynamic programming (DP). The matching obtained is improved using an autoregressive modeling of the disparity.
- De-epipolarization of the disparity and recovery of the 3D position of each pixels. The 3D position is valid to a scale factor since no calibration with a real object is performed.

The thesis is structured as follows. In Chapter 2, a literary review of the techniques for stereo fusion and structured lighting is presented.

The structured light used here is encoded with a pseudo random binary array (PRBA), the supporting theory along with a description on how it can be extracted from its image, is given in Chapter 3.

The epipolar geometry, the camera model and the F matrix are explained in Chapter 4. This chapter contains also the description of a method to estimate the F matrix and of a method to epipolarize and de-epipolarize two images.

In Chapter 5, the approach used for stereo fusion is described. The method involves both dynamic programming (DP) and autoregressive modeling of the disparity. In this chapter, the description of how a 3D model can be constructed from the disparity information is also presented.

In Chapter 6, the results obtained for four test cases are presented and discussed as well as a description of the overall algorithm is given.

Finally, a conclusion and possible future research directions are presented in Chapter 7.

Chapter 2

Methods for 3D reconstruction

Several methods are used to solve the problem of recreating a 3D view of an object from the information content of two 2D images. Barnard and Fischler [4] and, more recently, Dhond and Aggarwal [9] reviewed some of the methods for stereo fusion. Jarvis [15] wrote a perspective on range finding techniques, such as: structured lights, stereo fusion, focusing the cameras, camera motion, etc... For the purpose of this chapter, the structured light and stereo fusion methods will be reviewed and a description of the newer techniques will be presented.

2.1 Methods for Stereo fusion

Establishing a correspondence between the points in the stereo images is the most important and difficult step toward reconstructing a 3D scene from stereo images. Matching strategies can be differentiated by the primitive used for matching as well as by the imaging geometry used. The difference in matching primitive separates area-based from feature-based matching. Imaging geometry creates a distinction between parallel axis and non-parallel axis stereo systems.

Dhond and Aggarwal [9] have described the difference between area- and feature-based matching.

> Area-based stereo techniques use correlation among brightness (intensity) patterns in the local neighborhood of a pixel in one image with brightness patterns in a corresponding neighborhood of a pixel in the other image. First, a point of interest is chosen in one image. A cross-correlation measure is then used to search for a point with a matching neighborhood in the other image. The area-based techniques have

a disadvantage in that they use intensity values at each pixel directly, and are hence sensitive to distortions as a result of changes in viewing position (perspective) as well as changes in absolute intensity, contrast, and illumination. Also, the presence of occluding boundaries in the correlation window tends to confuse the correlation-based matcher, often giving an erroneous depth estimate.

Feature-based stereo techniques use symbolic features derived from intensity images rather than image intensities themselves. Hence, these systems are more stable towards changes in contrast and ambient lightning. The features used most commonly are either edge points or edge segments (derived from connected edge points) that may be located with subpixel precision. Also feature-based methods allow for simple comparisons between attributes of the features being matched, and are hence faster than correlation-based area matching methods.

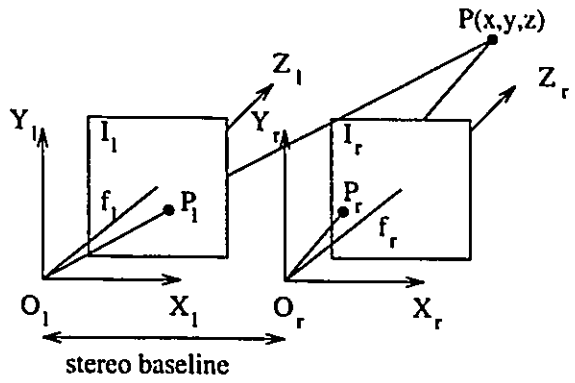


Figure 2.1: Conventional stereo system.

The image geometry being used affects the stereo matching paradigm. The conventional stereo system involves a pair of cameras with their optical axes parallel and horizontally separated. The stereo baseline denote the horizontal distance between them. The cameras have their optical axis perpendicular to the stereo baseline. And their image scanline is parallel to the stereo baseline. Such a system is shown in Figure 2.1.

In the conventional system shown in Figure 2.1, O_l and O_r represent the origin of the left and the right camera respectively, f_l and f_r represent their focal length, and P_l and P_r show where the point P is located in both images.

The epipolar constraint is valid for each scanline in a conventional system. This means that a point in a scanline from the left image is present in the corresponding scanline of the right image.

The epipolar constraint simplifies the matching problem to one dimension. This explains why the majority of the stereo matching algorithms assume the validity of the epipolar constraint. However, this requires instruments of very high precision and, in practical situations, the epipolar constraint for each scanline is generally not valid. A method to transform stereo images from a non-parallel system to an epipolar constrained one is proposed in Section 4.3.

2.1.1 Area matching

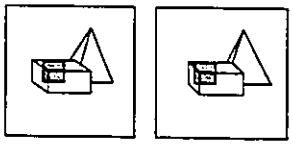


Figure 2.2: Area matching

Most of the earlier work used a simple correlation measure to match neighborhoods of points. To improve the results, Moravec [31] used a coarse-to-fine technique. Initially, an interest operator is used to identify feature points. For each feature points from the left image, a search is made in the right image at various resolutions ($\times 16, \times 8, \ldots, \times 1$) starting from the coarsest. At each resolution, the position yielding the highest correlation coefficient is enlarged to the next resolution. The process continues until the resolution $\times 1$ is reached. A disparity was found for each of the feature points with such a technique.

Further improvements were made by Gennery [10] who developed a high-resolution correlator which gave better results and also was able to estimate the accuracy of the match in the form of a variance and a covariance of the point (x, y) of the match in the second image. Other improvements include the ones made by Hannah [12] who was able to obtain subpixel matching by parabolic interpolation of correlation values.

2.1.2 Relaxation process

Rosenfeld and al. [38] proposed a fairly general model for scene labeling called relaxation labeling. To match stereo pairs using relaxation labeling, a set of feature points (nodes) are identified in

each images. The problem involves assigning unique labels (or matches) to each node out of a list of possible matches. For each candidate pair of matches, a matching probability is updated iteratively depending upon the matching probabilities of neighboring nodes. Strong neighboring matches improve the chances of weaker matches in a globally consistent manner.

Marr and Poggio [29] [28] use the neighborhood information of matchable primitives in a simple iterative scheme. A two dimensional network of nodes is set up for each scanline pair in the stereo images. The horizontal and vertical connections are described as inhibitory. The diagonal connections are termed excitatory. All the horizontal and vertical nodes inhibit each other until finally only one match remains on each horizontal and vertical line.

Barnard and Thompson [3] suggest to extract features points (nodes) using the Moravec[31] interest operator from each image. Each node in the left image is assigned a set of labels that represent the possible candidate matches from the right image within a disparity range. A relaxation process is then applied to impose global consistency. The consistency of a node is given by a probability factor. The iterative procedure used is continued until the probabilities reach a steady state or a maximal number of iteration is reached.

Kim and Aggarwal [17] propose a relaxation scheme that combines three disambiguity constraints: continuity of disparity, figural disparity and smoothness of probability (certainty) of matching. The probability of matching corresponds to a correlation factor.

2.1.3 Using Edge Segments

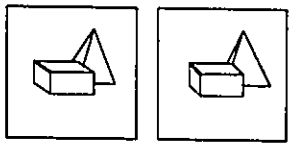


Figure 2.3: Matching using edge segments

It has been shown by Ayache and Faverjon [1] and by Medioni and Nevatia [30] that piecewise-linear approximation to connected edge points as a matching primitive is a viable alternative to matching individual edge points. Linear edge segments have some advantages over single edge points in the matching process:

- A positional error at an isolated point has little effect on the position and orientation of

the edge segment;

- Most of the edge points lie very close to the best fit of the edge segment;
- The edge connectivity constraint is always met.

On the other hand, the edge segments might be fragmented in the preprocessing stage. Provision must be taken to allow a segment to match one and two or more segments in the other image and vice-versa.

Two main approaches are used for edge segment matching: the minimal differential disparity criterion [30] and the Ayache-Faverjon algorithm [1]. The main differences between the two is that the Ayache-Faverjon method utilizes a generalized non-parallel axis imaging geometry and uses disparity between midpoints of matching line segments rather than average disparity between corresponding points that lie on matching line segments.

2.1.4 Integrated methods

Integrated methods are methods where the disparity map is obtained by joining the information received from two or more different approaches.

Cochran and Medioni [8] developed a method where an area-based approach is fused with a feature-based approach. They use a coarse-to-fine approach. Initially the feature and area based method are computed separately. Their results are combined to form a dense disparity map. For the finer levels, the dense disparity map of the previous level is used to improve the matching.

Maître and Luo [26] propose to combine the stereoscopic and monocular information of a scene. Their stereoscopic information (the disparity image) is obtained by any conventional stereo algorithm. The monocular information is used to generate homogeneous regions in the left and right image. A model of each region is made using the disparity map. The models can be of a planar or a quadratic form. The model is used to refine the disparity map.

2.1.5 Dynamic Programming

Baker and Binford[2] use the Viterbi algorithm, a dynamic programming technique, to partition the stereo matching problem recursively based upon the constraint that a left-to-right ordering of edges is preserved along a scanline in a pair of stereo images. A first pass is done to match half edges in the left image to those in the right image and vice-versa, then a cooperative procedure

identifies surface contours that are not continuous in disparity. Finally, an intensity-based Viterbi correlation, performed between intensity pixels from scanline intervals lying between a match pair of edges, yields a denser depth map.

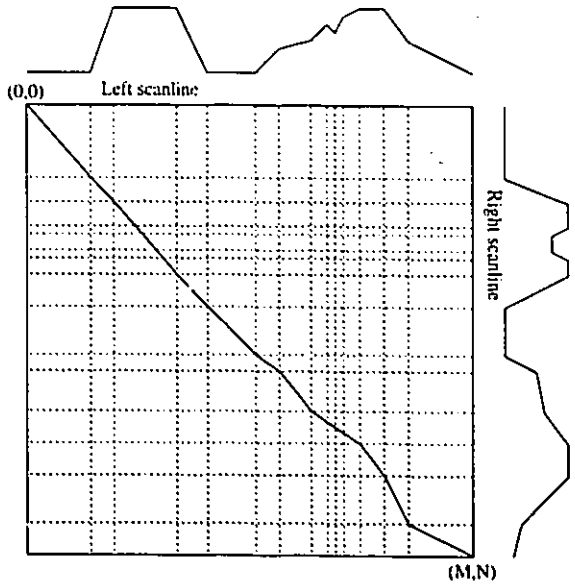


Figure 2.4: 2D search plane for intra-scanline search

Ohta and Kanade [32] use pixel intensities of scanline intervals (delimited by edge points) to guide the intra-scanline matching search by dynamic-programming. It is formulated as a path finding problem in a 2D search space in which the horizontal and vertical axes are respectively the right and left scanlines. The 2D search plane is shown in Figure 2.4. A cost function is associated with each partial path. The edges are numbered from left to right on each scanline. The end points of each scanline are also numbered as nodes. The solution can be represented as a sequence of straight lines from node $(0,0)$ to node (M,N) where M is the number of nodes in the left image and N the number of nodes in the right image. Next inter-scanline matching is done to impose consistency among matches obtained at each scanline using edge connectivity.

Recently, Roy and Meunier [40] proposed a Dynamic programming method where the cost function depends on the luminosity of each pixels. Their method generates a dense disparity map from a sequence of two or more stereoscopic images. They allow the camera displacement between each image to be any combination of rotation and translation.

2.2 Methods using Structured Light

In a laboratory or an industrial situation, special lighting effects can be used to reduce the computational complexity and to improve the reliability of 3-D object analysis. This class of methods involves illuminating the scene with a controlled light and interpreting the pattern of the projection in terms of the surface geometry of the objects.

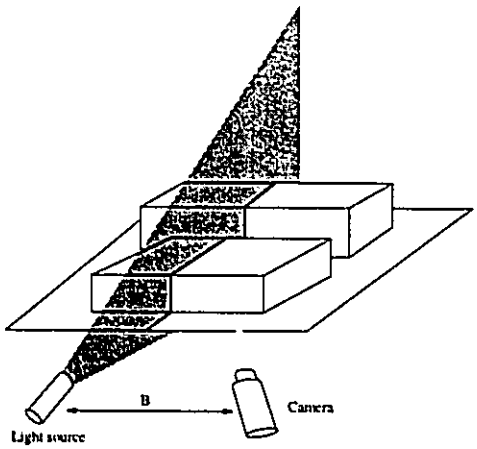


Figure 2.5: Range measurement by a structured light approach.

One of the controlled light scheme is to use a single ray of light scanned over the scene. When the light source is displaced from a viewing TV camera, the camera view of the stripe shows displacements along a stripe which are proportional to the depth. A kink in the displacement indicates a change of plane and a discontinuity, a physical gap between surfaces. The proportionality constant between the beam displacement and depth, is dependent upon the displacement between the light source and the camera. A more accurate depth measurement can be made with larger displacements. One of the drawbacks of this method is the need to analyze a huge sequence of images in order to obtain a 3D representation of the scene.

The number of images to analyze is reduced by projecting multiple lines onto the scene. However, line identification for tracing purposes becomes difficult. The strips patterns are encoded to simplify the identification process. The encoding techniques used are color coding[6], space encoding[36] [20], thickness labeling [20] and identification by pseudo random binary sequences[43]. In the case of grid patterns, the encoding used can be color coding, feature

coding and the identification by pseudo random binary arrays (PRBAs) coding [41][34].

Will and Pennington[45] describe a method by which the locations and orientation of planar areas of polyhedral solids are extracted through linear frequency domain filtering applied to images of scenes illuminated by a high contrast rectangular grid of lines. The edges are defined to be the intersections of the extracted planes.

More recently, Chen and al. [7] proposed a method to reconstruct and model a polyhedral surface of the scene using grid encoding. They model the polyhedral plane surface based on a model first described by Tseng and Chen [42]. They then use sensitivity analysis and a method to fuse line segments from different planes to refine the modeling. The sensitivity analysis is a function based on the Normal Vector estimation and the depth estimation.

A pseudo random binary array (PRBA) identification is used by Petriu and al. [34] to reconstruct a 3D object. This method finds the 3D locations of all the intersection points of the grid. It doesn't model the surfaces on which the grid is projected.

2.3 Conclusion

None of the methods mentioned suggests the utilization of two uncalibrated images. The knowledge of the relative rotation and translation between the cameras or an unknown epipolarization step is always assumed by the methods.

Zhang and al. [46] developed a method to robustly estimate the fundamental matrix F. The F matrix gives the relation between the location of two different views of the same scene. This method can also be used to compute the stereo disparity of some feature points. However, their method does not find a disparity for all the points of the images.

Hence, a method to generate the 3D locations of the points from two uncalibrated images is proposed in this thesis. The method allow to keep the computational complexity of the process to a minimum. Therefore, a structured light is used in conjunction with stereo-fusion to obtain the 3D location of the points.

The structured light simplifies the estimation of F which enables the epipolarization of the images. The stereo fusion is done using dynamic programming. Dynamic programming (DP) is a fast and optimal method to determine an optimal path. Here, the optimal path corresponds to the disparity between two epipolar lines.

Chapter 3

Pseudo random binary array encoding

The epipolar geometry between the two views of the object has to be estimated. The estimation process requires a list of matching points. The use of a structured light simplifies the generation of the list of matching points required for the epipolar geometry.

The structured light used here is a grid encoded with a pseudo random binary array (PRBA). This encoding allows the determination of the locations of the points in the grid without any ambiguities.

In this chapter, a description of how a PRBA is generated (in Section 3.2) is given. A description of how the information from the PRBA, projected onto an object, can be retrieved, is also presented. The information is the location in the PRBA of grid points in the image.

The retrieval of the information is done in two steps: extracting the raw information from the image (Section 3.3) and processing this information to match it to the PRBA projected onto it (Section 3.4). To extract the raw information, a dynamic threshold algorithm is used.

3.1 Thresholding

A thresholding operator is used to transform a gray image into a black and white image. The black and white image contains the foreground and the background information. Thresholding is done by setting all the pixels with a gray value greater than or equal to the threshold value T to a '1', and all value below it to a '0'.

The histogram of a gray image can be seen as a mixture of two Gaussian distributions. Their distributions having each a mean and variance (μ_1, σ_1^2) and (μ_2, σ_1^2) respectively and a respective proportion of q_1 and q_2. Kittler and Illingworth [18] use this hypothesis to determine the threshold value T. Their method consists of minimizing the Kullback [19] directed divergence J of the observed histogram $P(1), \cdots, P(I)$ from the unknown distribution f.

A mixture distribution f having a fraction q_1 of a distribution h_1 and a fraction q_2 of a distribution h_2 can be represented as:

$$f(i) = q_1 h_1(i) + q_2 h_2(i) \tag{3.1}$$

The mixture of two Gaussian distributions then takes the form

$$f(i) = \frac{q_1}{\sqrt{2\pi}\sigma_1} e^{-\frac{1}{2}(\frac{i-\mu_1}{\sigma_1})^2} + \frac{q_2}{\sqrt{2\pi}\sigma_2} e^{-\frac{1}{2}(\frac{i-\mu_2}{\sigma_2})^2} \tag{3.2}$$

The Kullback directed divergence J is defined by

$$J = \sum_{i=1}^{I} P(i) \log \frac{P(i)}{f(i)} \tag{3.3}$$

The parameters of the mixture distribution $f(i)$ can be estimated by minimizing J. J can be rewritten as

$$J = \sum_{i=1}^{I} P(i) \log P(i) - \sum_{i=1}^{I} P(i) \log f(i) \tag{3.4}$$

The first term does not depend on the unknown parameters. Therefore one can only minimize the second term

$$H = -\sum_{i=1}^{I} P(i) \log f(i) \tag{3.5}$$

The minimization process assume that the Gaussian distributions are well separated. Hence for some threshold t that separates the two distributions, we have

$$f(i) \approx \begin{cases} \frac{q_1}{\sqrt{2\pi}\sigma_1} e^{-\frac{1}{2}(\frac{i-\mu_1}{\sigma_1})^2} & i \leq t \\ \frac{q_2}{\sqrt{2\pi}\sigma_2} e^{-\frac{1}{2}(\frac{i-\mu_2}{\sigma_2})^2} & i > t \end{cases} \tag{3.6}$$

Rewriting $H(t)$ to be

$$H(t) = -\sum_{i=1}^{t} P(i) \frac{q_1}{\sqrt{2\pi}\sigma_1} e^{-\frac{1}{2}(\frac{i-\mu_1}{\sigma_1})^2} - \sum_{i=t+1}^{I} P(i) \frac{q_2}{\sqrt{2\pi}\sigma_2} e^{-\frac{1}{2}(\frac{i-\mu_2}{\sigma_2})^2} \tag{3.7}$$

And upon simplifying

$$H = \frac{1 + \log 2\pi}{2} - q_1 \log q_1 - q_2 \log q_2 + \frac{1}{2}(q_1 \log \sigma_1^2 + q_2 \log \sigma_2^2) \tag{3.8}$$

The value t that minimizes H is found by estimating the unknown values for each t. The unknown values are $q_1, q_2, \sigma_1, \sigma_2, \mu_1$ and μ_2.

3.1.1 Masked thresholding

Sometimes it is desirable to compute the threshold value for only a certain part of the object. In this case, a mask is used when computing the histogram of the image.

The histogram is normally computed by looking at all the pixels in the image. With a mask, the histogram is computed only for the pixels for which a corresponding pixel in the mask is defined.

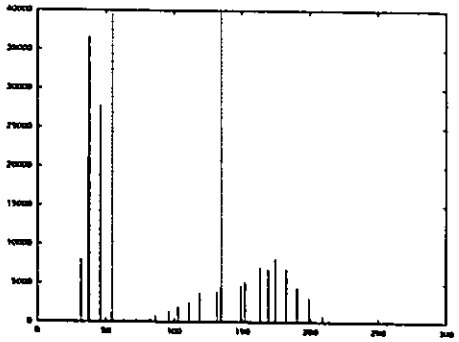
A test image is shown in Figure 3.1(a), its histogram in Figure 3.1(b), the result of applying a dynamic threshold operator in Figure 3.1(c), and the result of applying a dynamic threshold operator using Figure 3.1(c) as a mask in Figure 3.1(d). The threshold value used for (c) and (d) are 55 and 135 respectively. These values are shown in the histogram (Figure 3.1(b).

3.2 Pseudo-random encoding

The camera view of a projected grid is only a limited portion of the one projected. To create a 3D view of the object, a correspondence from the points viewed by the camera and the one projected by the projector must be done. When the grid is not encoded (as shown in Figure 3.2(a)) the process becomes nearly impossible. When it is encoded with a pseudo random binary array (PRBA) (as shown in Figure 3.2(b)), a correspondence can easily be made since each small array of size $k_1 \times k_2$ is unique in the PRBA. Therefore, by looking at the pattern present in the small array, the exact position from the PRBA can be deduced. In the case shown in Figure 3.2(b), a small array of 3×2 uniquely defines a space of the PRBA.

A pseudo-random binary array (PRBA) is defined by a n_1 by n_2 binary array encoded using a pseudo-random sequence such that a k_1 by k_2 window sliding over the array is unique and fully identifies the window's absolute coordinate (i, j) within the array. The following relations hold

(a) The test image

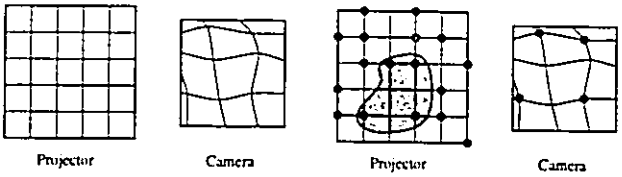


(b) The histogram of the image

(c) Applying a dynamic threshold operator

(d) Applying a dynamic threshold operator using a mask.

Figure 3.1: The thresholding operator



(a) The grid is not encoded and the camera views the grid projected over a smooth object.

(b) The grid is encoded with dots to form a PRBA and the camera view of the PRBA is shown as a grey area over the grid.

Figure 3.2: Camera view of a grid.

for the PRBA [25].

$$2^n - 1 = 2^{k_1 k_2} - 1 \tag{3.9}$$

$$n_1 = 2^{k_1} - 1 \tag{3.10}$$

$$n_2 = \frac{2^n - 1}{n_1} \tag{3.11}$$

where n_1 and n_2 must be relatively prime. Two numbers are relatively prime when they are not multiples of each other.

A $2^n - 1$ pseudo-random binary sequence (PRBS) is generated by the "primitive polynomials modulo 2" method [44]. Table 3.1 is based on results obtained by Watson [44] and gives the non-zero coefficients of some of the primitive polynomials used by the method up to order 30 .

A primitive polynomial defines a recurrence relation for obtaining a new random bit from the n preceding ones. This relation is guaranteed to produce a sequence of maximal length, *i.e.* to cycle through all possible sequences of n bits (except all zeros) before it repeats and get $2^n - 1$ random bits before the entire sequence is repeated.

The process by which the "primitive polynomials modulo 2" method generates random bits from a primitive polynomial is described by Press and al. [37]. The following is an example of how the method is used with a polynomial of degree 18

$$a_0 = a_{18}$$

$$a_5 = a_5 \oplus a_0$$

1,0	11,2,0	21, 2, 0
2,1,0	12,6,4,1,0	22,1,0
3,1,0	13,4,3,1,0	23,5,0
4,1,0	14,5,3,1,0	24,4,3,1,0
5,2,0	15,1,0	25,3,0
6,1,0	16,5,3,2,0	26,6,2,1,0
7,1,0	17,3,0	27,5,2,1,0
8,4,3,2,0	18,5,2,1,0	28,3,0
9,4,0	19,5,2,1,0	29,2,0
10,3,0	20,3,0	30,6,4,1,0

Table 3.1: Non-zero coefficients of some primitive polynomials

$$a_2 = a_2 \oplus a_0$$

$$a_1 = a_1 \oplus a_0$$

where $\oplus$ is the exclusive-or operator. To get the new value for a_0, a left-shift on the 18 bit number is performed before using the relationship above. This relationship is graphically represented in Figure 3.3. It is also well suited to be implemented by a computer.

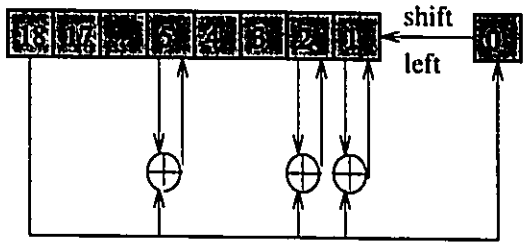


Figure 3.3: Random bit generation from a shift register and a primitive polynomial.

To create the PRBA, we fill an array with a PRBS by writing down the main diagonal and continuing from the opposite side whenever an edge is reached starting from the (0,0) coordinate. The process is depicted in figure 3.4. An empty circle in the figure represents a '0' and a full

circle represents a '1' from the PRBS. The i and j coordinate can be defined from the following relationship

$$i = n \bmod n_1 \tag{3.12}$$

$$j = n \bmod n_2 \tag{3.13}$$

where n is the n^{th} element of the PRBS sequence, and n_1,n_2 are as defined in equation 3.11.

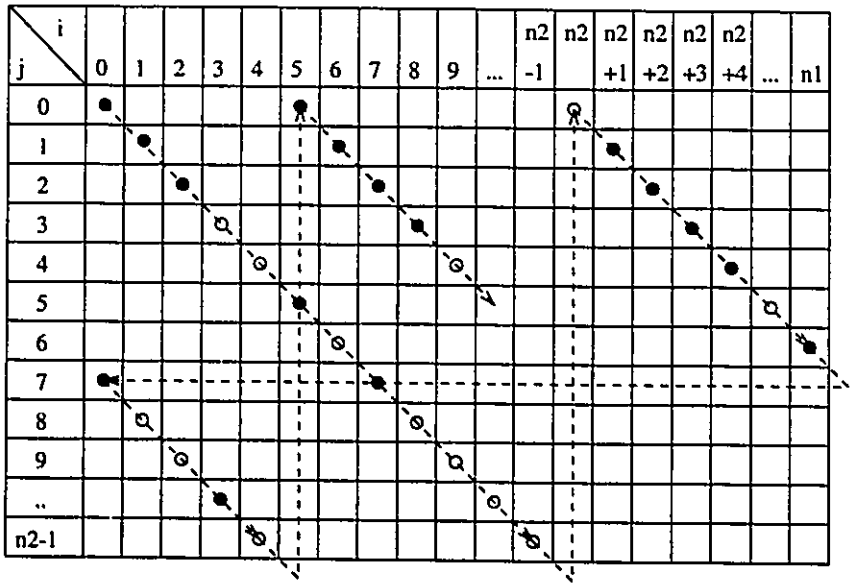


Figure 3.4: Creating the PRBA

Once a PRBA is created, the i,j coordinates of a $k_1 \times k_2$ window can be recovered. This is done using a memory store table. The window is converted into a $n = k_1 k_2$ bit number, this number is compared with the numbers stored in a table to find its coordinate.

A number is generated from the window as follows:

- A value is assigned to each of the position within the window. Each value corresponds to a bit in a binary number (*i.e.* $2^0, 2^1, 2^2, \ldots$).
- If a '1' is present at a position in the window, the corresponding value is added to the number.

As an example, a 3×2 window yields $n = 2^0 + 2^2 + 2^5 = 37$ when the values assigned and

the window are defined as

$$\text{values} = \begin{bmatrix} 2^4 & 2^5 \\ 2^2 & 2^3 \\ 2^0 & 2^1 \end{bmatrix}, \text{window} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 1 & 0 \end{bmatrix}$$

Here is an example showing how to locate a $k_1 \times k_2$ window using a primitive polynomial of degree 10. With $n = 10$, we choose k_1 and k_2 to be 5 and 2. This yields a value for n_1 and n_2 to be 31 and 33 respectively. Using these values to construct a PRBA, we obtain the array shown in figure 3.5.

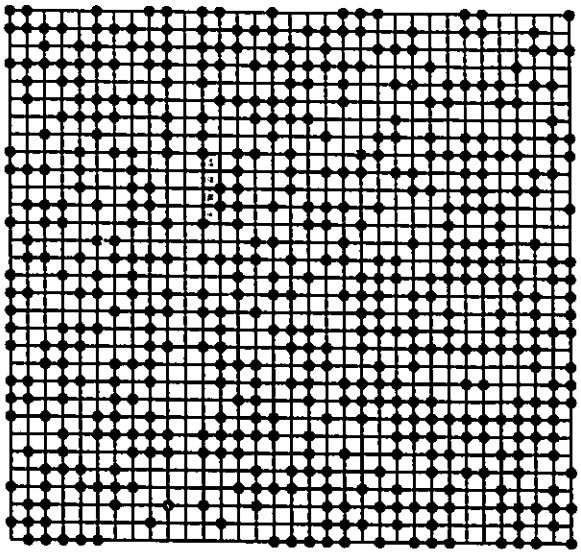
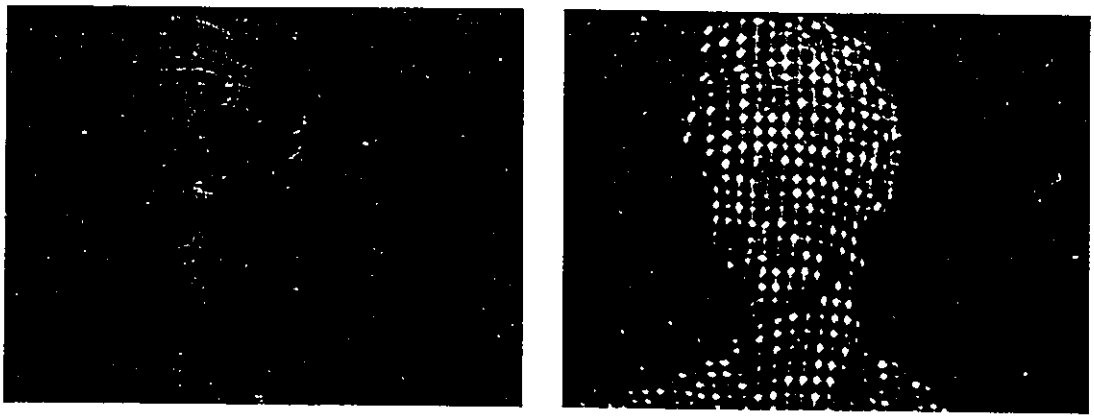


Figure 3.5: PRBA using primitive polynomial of degree 10

In the shaded area, we have a 2×5 window which can be used to find the (i, j) coordinate of the lower left corner of that window. The shaded region has the number $2^8 + 2^5 + 2^3 + 2^0 = 169$. From the lookup table, this number has the position $(11, 18)$ associated with it. The origin of the PRBA is defined to be on the lower left corner.

3.3 Extracting the encoded grid

In this section, a description of how the PRBA projected onto the object is extracted is presented. In Figure 3.6(a), the image of the actual object is shown and in Figure 3.6(b), the object with



(a) Object with good lighting and with no encoding

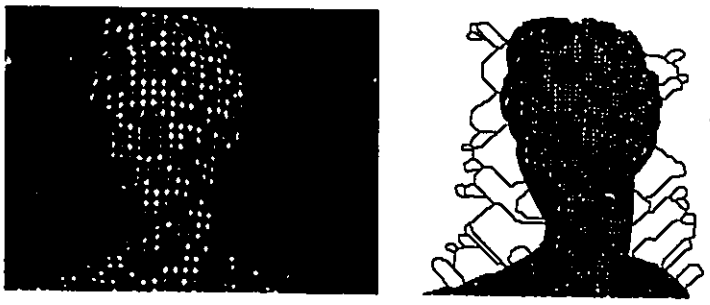
(b) Log of the image of the object with the projected PRBA.

Figure 3.6: Test image

the projected PRBA is shown. The extraction process is done in two steps: extracting the grid and extracting the disks.

3.3.1 Extracting the grid

The extraction of the grid from the image (Figure 3.6(b)) is done in two steps: a smoothing and an extraction process.



(a) Closing the image with a disk of size 5

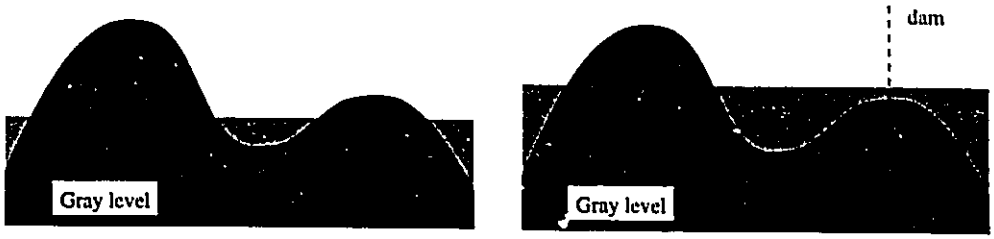
(b) The Watershed result overlapping the log of the object

Figure 3.7: Grid extraction

The smoothing process consists of applying a morphological closing operator with a disk structuring element of size 5. A greater disk size will tend to over-smooth the image and some

of the grid information will be lost.

The extraction process is done with the watershed algorithm. The watershed algorithm represents the image as if it was a 3D terrain. In the terrain, the gray level of a pixel represents the height at that point. The algorithm tries to flood the terrain with water. The water level is set to a low value and gradually increases. When two different water zones (or lakes) connect with each other, a dam is built to keep them separate. This algorithm is shown in Figure 3.8.



(a) The terrain represented by the gray level is flooded.

(b) A dam is built to keep the two different lakes separated.

Figure 3.8: The watershed algorithm.

The result of the watershed algorithm is a network of dams built to keep the lakes separate. The algorithm is very sensitive to noise and therefore requires a smoothing process.

The result of the smoothing process is shown in Figure 3.7(a). The grid extraction process is shown in Figure 3.7(b) overlapping the input image.

3.3.2 Background extraction

The watershed algorithm works on the whole image. Since the only interesting information lies within the object, the background can be removed. The background information is also important for disk extraction and stereo-fusion (see Section 3.3.3 and 5.1.1).

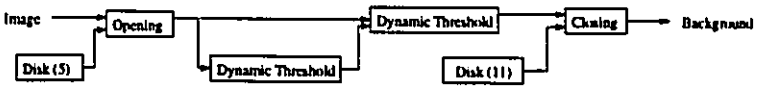


Figure 3.9: Extracting the background.

Extracting the background can be seen as detecting the foreground. The foreground corresponds to the object of interest. Detecting the object is done in three steps: filtering, thresholding and smoothing.

Filtering is used to eliminate small bright areas in the background. It is done by applying a morphological opening operator with a disk structuring element of size 5.

Thresholding is done in two steps. The first step is to do a dynamic threshold of the image. This creates a mask. The second step is to use this mask and do another dynamic threshold of the image. The mask limits the region for which a dynamic threshold value is calculated (see Section 3.1.1).

Smoothing involves a morphological closing operator with a disk of size 11. This operation closes the small gaps in the foreground.

The object extraction process is depicted in Figure 3.9. The result of each step is shown in Figure 3.10.

3.3.3 Extracting the disks

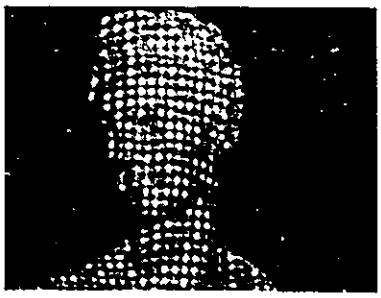
Disk extraction is a two steps process: opening and thresholding.

The morphological opening operator is applied using a disk of size 5. This tends to diminish small bright regions (for example, the intersection of two lines) while preserving the larger ones (the disks).

A global threshold is applied using the background image as a mask. The mask defines the regions of interest in the images (the object). When limited to the masked region, the threshold operator will pickup only the brighter spots on the object. These brighter spots correspond mostly to the projected disks from the PRBA. A bright spot might also come from a bright intersection between two lines. Some of the projected disks might have a low intensity level and might be missed by this process. This is considered when the disk information is used.

3.4 Simplifying the grid

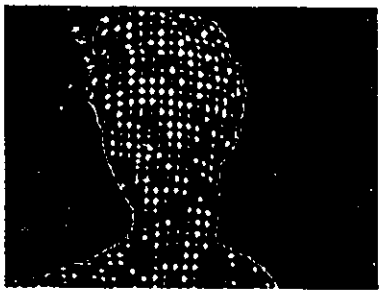
In Section 3.3, it is discussed how the raw PRBA information contained in the image is extracted from the input image. This information cannot be used directly. More information on the precise location of the intersection points, their connectivity and their type is needed. To obtain these



(a) Opening the image with a disk of size 5

(b) Dynamic threshold of (a)

(c) Dynamic threshold of (a) using (b) as a mask



(d) The smoothed result overlapping the log of the input image

Figure 3.10: The results of each background extraction process.

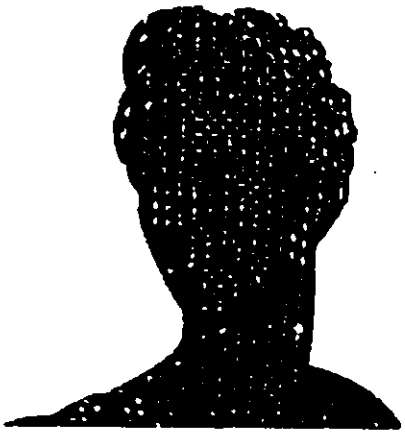
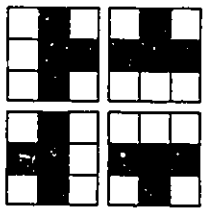
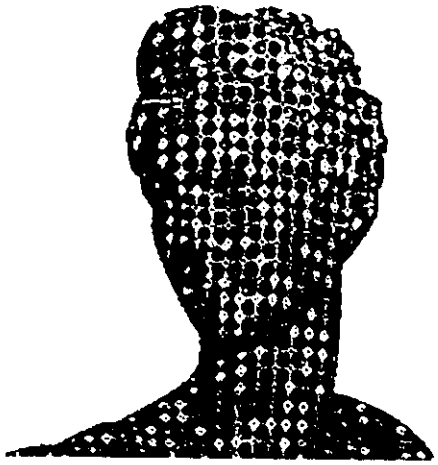


Figure 3.11: Disks extracted superposed to the log of the input image

informations many steps are required.



(a) The kernels used to extract the intersections



(b) The intersections found are shown as black points in the image

Figure 3.12: The different kernels used with the erosion operator to find the intersection points.

The intersections are found by using the morphological erosion operator. This operator removes all points in the grid not belonging to a kernel. This operator is used with 4 different kernels (figure 3.12(a)). The union of the erosion operator from all the kernels gives the location of all the intersection points. Figure 3.12(b) shows all the intersections found by the kernels.

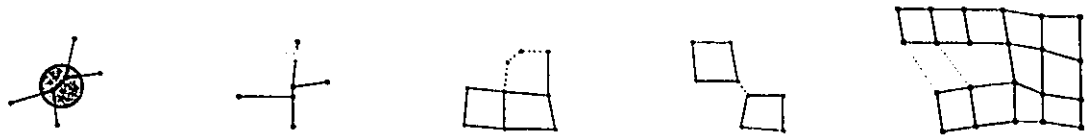
The type of an intersection defines if the intersection corresponds to a '1' or a '0' in the PRBA. A disk represents a '1', the absence of it represents a '0'. All the intersection points are labeled properly depending on the presence or the absence of a disk at their location.

The connectivity pattern between each intersection is found by determining the presence and the location of another intersection directly connected to each other. This is done by following the grid in the proper direction (up, down, right and left) , starting from an intersection until another intersection is found. Following the grid is equivalent to finding a direct closed path between two intersection points. Figure 3.13(a) shows all the intersections and how they are connected to each other.

From Figure 3.13(a), we see that too many intersection points and too many connections are

Figure 3.13: The grid information

found. The following sequential steps remove or improve the location and/or the connectivity information:



(a) Two points in the same disk.

(b) Two points close to each other.

(c) Points not part of a square.

(d) Connection not part of a square.

(e) Connections not agreeing with the setup.

Figure 3.14: Situations for which an intersection or a connection must be removed

1. The merging of the intersections
   - If 2 or more intersections are part of the same disk. (Figure 3.14(a))
   - If 2 intersections points are close to each other. (Figure 3.14(b))
2. The removal of intersection points if they're not part of a square (Figure 3.14(c))
3. The removal of the connections if they are not part of the same square (Figure 3.14(d))
4. The removal of a connection if it does not agree with the grid setup. The grid setup defines

how the grid points are related to each other. If a point is defined to be at (i, j) then the point above it should be defined to be at $(i-1, j)$. If this is not the case, then the connection is removed. In Figure 3.14(e), the two dotted connections are removed because they are not agreeing with the darker connections. The left connection is removed since it makes an upward connection between the point (i, j) and $(i-1, j-1)$, the right connection is eliminated for a similar reason.

The resulting grid information is shown in Figure 3.13(b). In the figure, a '1' is shown as a wider disk.

3.5 Matching the grid to the PRBA

With the simpler grid obtained in the above section, the intersection points found in the image can then be matched to the projected PRBA. The grid information usually contains some errors: some of the intersection points may have a wrong type associated with them and/or some of the connectivity information may be missing. An intersection point has a wrong type when an error occurred in determining the presence or the absence of a disk (see Section 3.3.3).

Matching the grid to the PRBA is done in three steps: (1) labeling $k_1 \times k_2$ areas to their corresponding location in the PRBA, (2) finding the single intersection point which has the largest number of support and finally, (3) spreading the information from that point.

The first step is to find all $k_1 \times k_2$ areas in the grid. In the example in Figure 3.6(b), the area is a 2×7 region. The area is associated with a binary number of length $k_1 k_2$: the upper left corner is the lowest bit, the lower right corner is the highest bit. This number is compared to a lookup table for which a x and y location is associated in the PRBA. This is explained in Section 3.2. Each upper left corner of the $k_1 \times k_2$ regions are labeled with their corresponding (x, y) location in the PRBA.

In this first step, a PRBA location for all the $k_1 \times k_2$ regions in the grid is found. Due to errors in finding if a disk is present or not at a certain location on the grid, the result of the operation yields incompatible matches. An incompatible match means that two neighbors in the grid are not neighbor in the PRBA. The result of the first operation is shown in Figure 3.15.

The second step is to find an intersection point P which has the largest number of support. The number of support is the number of labeled points which are compatible with themselves.

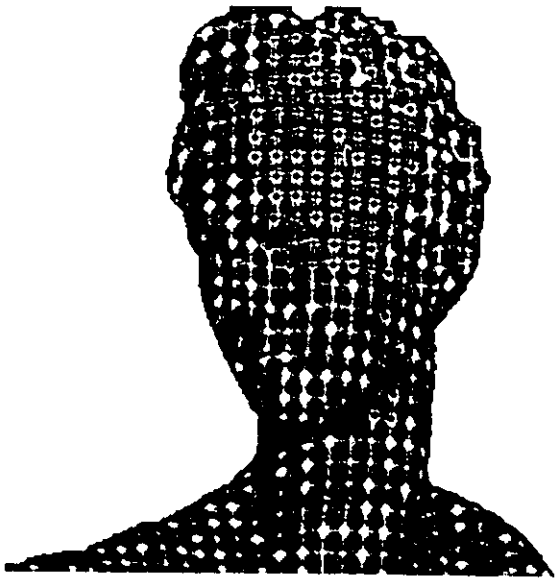


Figure 3.15: Log of the input image with the label points obtained after the first step

Being compatible with a point means that if a point is labeled (x, y) at a grid location (X, Y) then the grid point at location $(X + i, Y + j)$ must be labeled $(x + i, y + j)$.

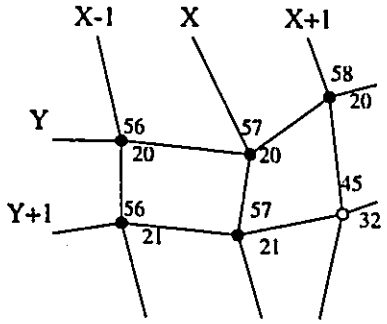


Figure 3.16: The white point $(45, 32)$ is not compatible with the other labeled points

In Figure 3.16 all the black dots are compatible. However, the white point labeled $(45, 32)$ at $(X + 1, Y + 1)$ isn't compatible with the other labeled points at (X, Y), $(X + 1, Y)$, ...

The final step is to spread the information from P. This operation consists in setting, for all the points in the grid, a label for that point in accordance with the label at the point P.

Sometimes, this operation will label two different points identically. If this happens, they are both removed from the set of labeled points. The result of this operation is shown in Figure 3.17. In this figure, the only label points shown are the points which were originally compatible with P. This operation also spreads the type information (a '0' or a '1'). The intersections where an error was originally made about the type are encircled.

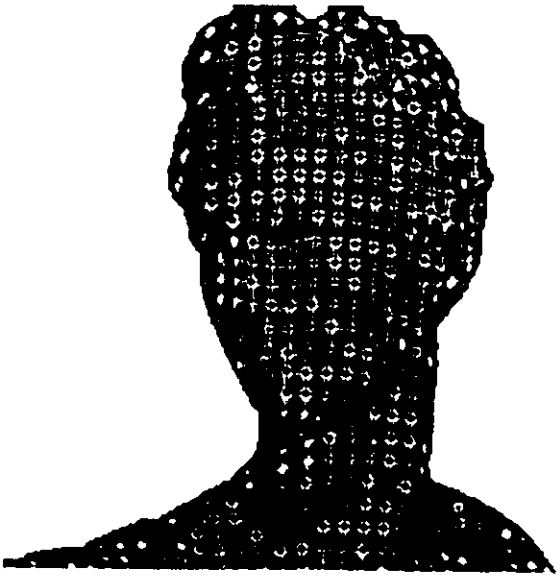


Figure 3.17: Result of spreading the pseudo-random information from the point P

In the Figure 3.17 , there are 264 intersections and 23 errors were originally made in the determination of the presence (or absence) of a disk. It should be noted that if there are too many errors in finding if a disk is present or not on an intersection, then it is possible that the grid points will be mislabeled.

The process generates the PRBA from the information it has received. This information might have some errors. For example, on the left shoulder of the statue a group of points are false matches. A false match indicates that a point is wrongly match to the PRBA, i.e. it is labeled (i, j) when it should be labeled (u, v). A false match usually occurs when two meshes are seen as one by the grid extraction process.

The grid projected onto the object acts like a digitizer for an audio signal. It must have a sampling frequency higher than twice the frequency of the signal otherwise it looses some

information. Therefore if the depth changes too rapidly between grid points, it can easily create problem regions for the grid extraction process; like the one seen on the lower left shoulder of the statue. However, this effect can be minimized when using the information of the PRBA.

3.6 Matching two PRBAs

When two images of an object with a PRBA projected onto it are taken, a disparity map can be easily obtained for the two images. As it will be seen later, the disparity of a point gives a scaled version of its 3D location.

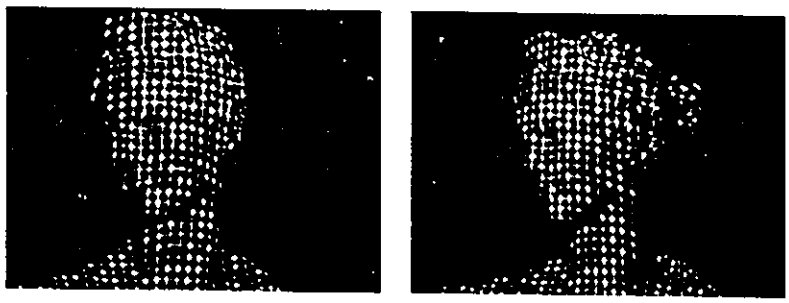


Figure 3.18: Two views of an illuminated object.

In Figure 3.18, two images of the same object are taken. The PRBA is extracted from these two images using the algorithm shown in this chapter. The PRBA extracted is shown in Figure 3.19. The disparity value of the grid points are determined if a matching grid point exists in the other image. The disparity value is set to the absolute distance between the location of the point in the left image to its corresponding point in the right image.

A linear interpolation between disparity points of the grid must then be computed. The method for linear interpolation is

1. Finding 4 connected points lying in the same square.
2. Computing the mean intensity value of these points.
3. Finding the location of the middle point.
4. Dividing the square into 4 triangles having the corner points (middle, up-left, up-right),

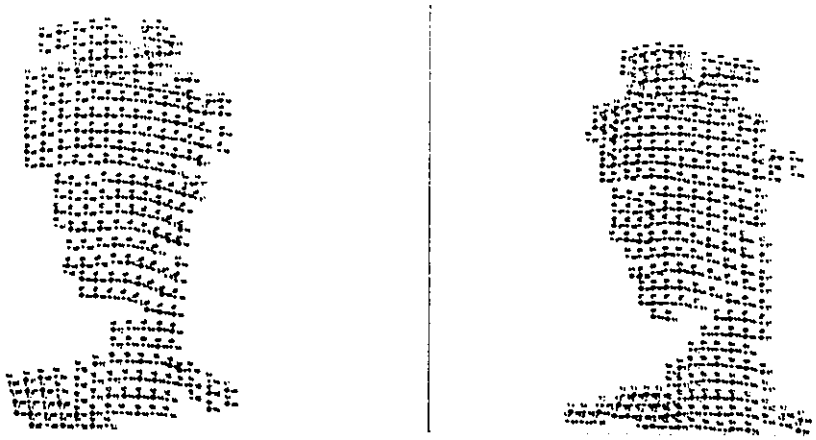
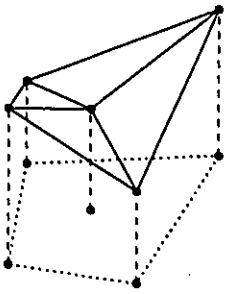


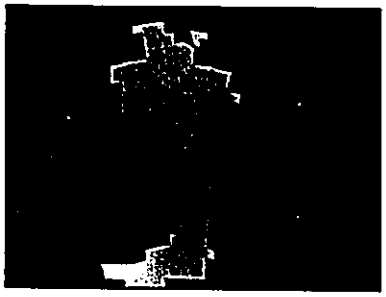
Figure 3.19: The two PRBAs extracted from the images above.

(middle, up-left, down-left), (middle, down-left, down-right) and (middle, up-right, down-right).

5. Fitting a planar surface through each triangle and assign the disparity value of a point inside a triangle to be the plane height at that point.



(a) Linear interpolation for 4 connected grid points.



(b) The coarse disparity obtained.

Figure 3.20: The coarse disparity obtained from two PRBAs.

The linear interpolation method is shown in Figure 3.20(a). The coarse disparity obtained from matching the two images is shown in Figure 3.20(b). The coarse disparity gives a scaled 3D perspective of the object. The lighter points corresponds to points which are closer to the left

camera and the darker ones are farther away from the left camera. the points for which there is no 3D information available are shown in black.

3.7 Conclusion

In this chapter, a description of how a PRBA can be constructed and later on extracted from an image is given. It is also demonstrated how matching two views of the same object illuminated with a PRBA can generate a list of matching points.

The list of matching points can be used to generate a coarse disparity map of the object. It can also be used to estimate the epipolar geometry between the two views of the object. The estimation process is described in the next chapter.

Chapter 4

The epipolar geometry

To extract the depth from a pair of stereo images the orientation of the cameras must be determined. Unfortunately, this can not be done from only two uncalibrated images. However, the epipolar geometry can be estimated. The epipolar geometry gives the relation between the two cameras. This relation takes into account the relative orientation of the cameras and their internal parameters, such as: focal length, linear distortions, ... Therefore, a scaled version of the the depth can be estimated.

The depth is extracted by using the information provided by the stereo fusion process. To simplify this process, the epipolarity constraint has to be valid.

This chapter presents a description of how the epipolar geometry can be extracted from two images. First, a camera model has to be defined and, then, the epipolar geometry by estimating the fundamental matrix F. Finally a method to transform the pair of stereo images into a pair of stereo epipolar images is proposed.

4.1 The Camera Model

In the 3D reconstruction literature, a camera model is described by the pinhole model.This model relates a 3D point $M = [x, y, z]^t$ from a world coordinate system to its retinal image coordinate

$m = [u, v]^t$. They are related by

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathcal{P} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{4.1}$$

where s is an arbitrary scaling factor and $\mathcal{P}$ is 3×4 matrix called the projective projection matrix.

The matrix $\mathcal{P}$ can be written as

$$\mathcal{P} = A[R\_t] \tag{4.2}$$

where A is a 3×3 matrix, mapping the coordinates of the normalized image to the coordinates of the retinal image and $[R\_t]$ is the 3D displacement (rotation and translation) from the world coordinate system to the camera coordinate system, where R is a 3×3 matrix and t is a vector of size 3.

In the photogrammetric literature, the pan, swing and tilt notation is often used to describe the rotation matrix R. This notation corresponds to the following sequence of rotations:

1. A clockwise rotation of κ around the z-axis (swing).
2. A clockwise rotation of ϕ around the y-axis (pan).
3. A clockwise rotation of ω around the x-axis (tilt).

The resultant rotation matrix R is

$$R = \begin{bmatrix} cos\phi cos\kappa & sin\omega sin\phi cos\kappa + cos\omega sin\kappa & -cos\omega sin\phi cos\kappa \\ -cos\phi sin\kappa & -sin\omega sin\phi sin\kappa + cos\omega cos\kappa & cos\omega sin\phi sin\kappa + sin\omega cos\kappa \\ sin\phi & -sin\omega cos\phi & cos\omega cos\phi \end{bmatrix} \tag{4.3}$$

To avoid long trigonometric expressions, the following convention is used

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

Using this convention,

$$[R\_t] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \tag{4.4}$$

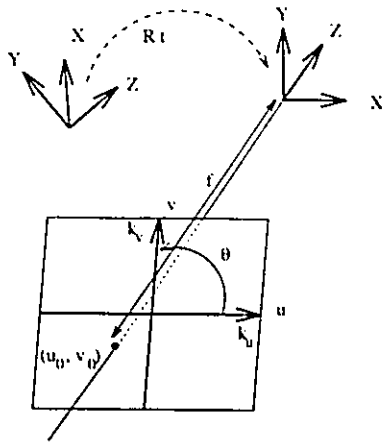


Figure 4.1: The general projective camera model.

The most general intrinsic matrix A can be written as (see Figure 4.1)

$$A = \begin{bmatrix} -fk_u & fk_u cot\theta & u_0 \\ 0 & -\frac{fk_v}{sin\theta} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.5}$$

where

- f is the focal length of the camera.
- k_u and k_v are the horizontal and vertical scale factors, whose inverses characterize the size of the pixel in the world coordinate unit.
- u_0 and v_0 are the coordinates of the principal point of the camera, *i.e.* the intersection between the optical axis and the image plane.
- θ is the angle between the retinal axes. In practice, this parameter is very close to $\pi/2$.

4.2 The epipolar geometry and the Fundamental matrix

The epipolar geometry is the basic constraint which arises from the existence of two viewpoints. Figure 4.2 describes the linear projection of a point M in a three dimensional system onto two images taken from two cameras located at different locations. Let C and C' be the optical center

of each camera. The line $<C, C'>$ is projected onto e in the left image and e' in the right image. The points e and e' are the epipoles. The lines going through e in the left image and e' in the right images are the epipolar lines. The epipolar constraint is thus stated as: for each point m lying on the first retina, its corresponding point m' lies on its epipolar line l'_m on the second retina.

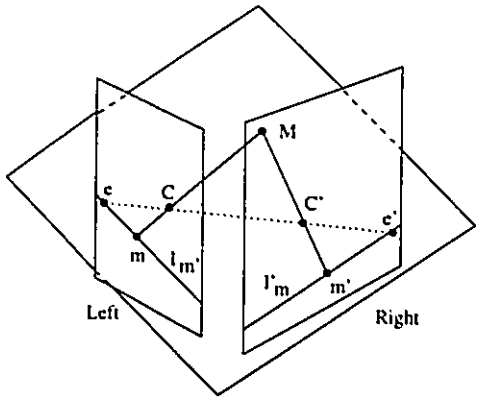


Figure 4.2: The epipolar geometry

The relationship between m and its corresponding line l'_m is projective linear. Because the relations between m and $<C, M>$, and m and $<C, M>$ and its projection l'_m are both projective linear. The 3×3 matrix F which describes this correspondence is called the *fundamental matrix*. The epipolar constraint can now be expressed using the *fundamental matrix* F. For a given point m in the first image (the left image), the projective representation l'_m of its corresponding epipolar line in the second image is given by

$$l'_m = Fm$$

Since the point m' corresponding to m belongs to the line l'_m, it follows that

$$m'^T F m = 0 \tag{4.6}$$

The fundamental matrix can also be derived from the analysis of a stereo system under the assumption of the pinhole model . Let the displacement from the first camera to the second be (R,t). Let m and m' be the images of a 3D point M on the cameras. M is expressed from the the coordinate frame of the first camera. Using the pinhole model, the following two equations

are defined

$$sm = A[I0]\begin{bmatrix} M \\ 1 \end{bmatrix} \tag{4.7a}$$

$$s'm' = A'[Rt]\begin{bmatrix} M \\ 1 \end{bmatrix} \tag{4.7b}$$

where A and A' are the intrinsic matrices. Eliminating s, s' and M we obtain

$$m'^T A'^{-T} T R A^{-1} m = 0 \tag{4.8}$$

where T is an antisymmetric matrix defined by t such that $Tx = t \times x$ for all 3D vectors x. The proof of Equation (4.8) can be seen in Appendix A. T is defined by the following matrix

$$T = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix} \tag{4.9}$$

From the above equation, F is defined as

$$F = A'^{-T} T R A^{-1} \tag{4.10}$$

The fundamental matrix F is the only information available from two uncalibrated images. It can be shown that F is related to the essential matrix $E = t \times R$[14][21] by

$$F = A'^{-T} E A^{-1} \tag{4.11}$$

4.2.1 Linear estimation of F

The equation 4.6 can be written as

$$U^T f = 0 \tag{4.12}$$

where

$$U = [m_x m'_x, m_y m'_x, m'_x, m_x m'_y, m_y m'_y, m'_y, m_x, m_y, 1]$$

$$f = [F_{11}, F_{12}, F_{13}, F_{21}, F_{22}, F_{23}, F_{31}, F_{32}, F_{33}]$$

From (4.12) it can be seen that given 8 matches, the F matrix can be determined up to a scale factor. However, in practice a lot more than 8 matching points are observed. Therefore a

least-square method is used to solve

$$\min_F \sum_i (m_i'^T F m_i)^2 \tag{4.13}$$

which can be written as

$$\min_f ||\tilde{U} f||^2 \tag{4.14}$$

where

$$\tilde{U} = \begin{bmatrix} U_1^T \\ \vdots \\ U_n^T \end{bmatrix}$$

This can be solved using singular value decomposition (SVD).

From the SVD theorem, any matrix A can be written as

$$A = USV^T$$

or as

$$A = U[\text{diag}(s_j)]V^T$$

If $Ax = b$ where A is a $M \times N$ matrix and x and b are vectors, then for the case where $b = 0$, SVD gives an immediate solution for x. Any columns of V whose corresponding s_j is zero yields a solution for x. In the case where there is no zero value, the lowest value is used to find a corresponding column.

The linear criterion of Equation (4.12) gives a non-iterative solution for F. However, it was found by Faugeras [46][23] that it is very sensitive to noise, even with a large set of data points ([1]). Faugeras states the two main reasons for this:

- The constraint $\det(F) = 0$ is not satisfied, this causes inconsistencies of the epipolar geometry near the epipoles.
- The criterion is not normalized, which causes a bias in the localization of the epipoles.

[1]This result is also consistent with the result obtained in this paper

4.2.2 Non-linear estimation of the F matrix

The matrix F can be parameterized as (see Section 4.3.1):

$$F = \begin{bmatrix} b & a & -ae_y - be_x \\ -d & -c & ce_y + de_x \\ de'_y - be'_x & ce'_y - ae'_x & -ce_y e'_y - de'_y e_x + ae_y e'_x + be_x e'_x \end{bmatrix} \tag{4.15}$$

The above definition of F shows that the matrix has a rank equal or less than two. Since, if c_1, c_2 and c_3 represent the columns of F, then $e_x c_1 + e_y c_2 + c_3 = 0$. To determine F using the rank limitation, non-linear optimization must be performed. The idea is to minimize $\sum_i d^2(m'_i, Fm_i)$, where $d(m'_i, Fm_i)$ is the Euclidean distance from the point m'_i to its epipolar line Fm_i. It is given by

$$d(m'_i, Fm_i) = \frac{|m_i'^T F m_i|}{\sqrt{(Fm_i)_1^2 + (Fm_i)_2^2}} \tag{4.16}$$

where $(Fm_i)_j$ is the j-th component of vector Fm_i. The two images do not have a symmetric role. To obtain a consistent epipolar geometry, it is necessary and sufficient that by exchanging the two images, the fundamental matrix is changed to its transpose (Faugeras [46]). This yields the following criterion

$$\sum_i d^2(m'_i, Fm_i) + d^2(m_i, F^T m'_i) \tag{4.17}$$

which operates on both images simultaneously. If we use the relationship $m'^T F m = m^T F^T m'$, the criterion can be rewritten as

$$\sum_i \left(\frac{1}{(Fm_i)_1^2 + (Fm_i)_2^2} + \frac{1}{(F^T m'_i)_1^2 + (F^T m'_i)_2^2} \right) (m_i'^T F m_i)^2$$

or using the parameterizing variables as

$$\sum_i J_i \tag{4.18}$$

where $J_i = AB^2$, A and B are defined as

$$\begin{aligned} A &= \left(\frac{1}{(bm_{i_x} + am_{i_y} - ae_y - be_x)^2 + (-dm_{i_x} - cm_{i_y} + ce_y + de_x)^2} \right. \\ &\quad \left. + \frac{1}{(bm'_{i_x} - dm'_{i_y} + de'_y - be'_x)^2 + (am'_{i_x} - cm'_{i_y} + ce'_y - ae'_x)^2} \right) \end{aligned}$$

$$\begin{aligned} B &= (m_{i_x} bm'_{i_x} - m_{i_x} dm'_{i_y} + m_{i_x} de'_y - m_{i_x} be'_x + m_{i_y} am'_{i_x} - m_{i_y} cm'_{i_y} + m_{i_y} ce'_y \\ &\quad - m_{i_y} ae'_x - m'_{i_x} ae_y - m'_{i_x} be_x + m'_{i_y} ce_y + m'_{i_y} de_x + ce_y e'_y - de'_y e_x + ae_y e'_x + be_x e'_x) \end{aligned}$$

The estimation of F is done by minimizing 4.18. The minimization is done using the CFSQP library. The CFSQP library is the implementation of an algorithm based on Sequential Quadratic Programming. The library requires the derivative of the equation to limit the search space.The partial derivatives of J_i in respect to each variable are

$$\begin{aligned}\frac{\delta J_i}{\delta a} &= \left(-2\frac{(bm_{i_x}+am_{i_y}-ae_y-be_x)(m_{i_y}-e_y)}{((bm_{i_x}+am_{i_y}-ae_y-be_x)^2+(-dm_{i_x}-cm_{i_y}+ce_y+de_x)^2)^2}\right.\\ &\left.-2\frac{(am'_{i_x}-cm'_{i_y}+ce'_y-ae'_x)(m'_{i_x}-e'_x)}{\left((bm'_{i_x}-dm'_{i_y}+de'_y-be'_x)^2+(am'_{i_x}-cm'_{i_y}+ce'_y-ae'_x)^2\right)^2}\right)B^2\\ &+2AB(m_{i_y}m'_{i_x}-m_{i_y}e'_x-m'_{i_x}e_y+e_ye'_x)\end{aligned}$$

$$\begin{aligned}\frac{\delta J_i}{\delta b} &= \left(-2\frac{(bm_{i_x}+am_{i_y}-ae_y-be_x)(m_{i_x}-e_x)}{((bm_{i_x}+am_{i_y}-ae_y-be_x)^2+(-dm_{i_x}-cm_{i_y}+ce_y+de_x)^2)^2}\right.\\ &\left.-2\frac{(bm'_{i_x}-dm'_{i_y}+de'_y-be'_x)(m'_{i_x}-e'_x)}{\left((bm'_{i_x}-dm'_{i_y}+de'_y-be'_x)^2+(am'_{i_x}-cm'_{i_y}+ce'_y-ae'_x)^2\right)^2}\right)B^2\\ &+2AB(m_{i_x}m'_{i_x}-m_{i_x}e'_x-m'_{i_x}e_x+e_xe'_x)\end{aligned}$$

$$\begin{aligned}\frac{\delta J_i}{\delta c} &= \left(-2\frac{(-dm_{i_x}-cm_{i_y}+ce_y+de_x)(-m_{i_y}+e_y)}{((bm_{i_x}+am_{i_y}-ae_y-be_x)^2+(-dm_{i_x}-cm_{i_y}+ce_y+de_x)^2)^2}\right.\\ &\left.-2\frac{(am'_{i_x}-cm'_{i_y}+ce'_y-ae'_x)(-m'_{i_y}+e'_y)}{\left((bm'_{i_x}-dm'_{i_y}+de'_y-be'_x)^2+(am'_{i_x}-cm'_{i_y}+ce'_y-ae'_x)^2\right)^2}\right)B^2\\ &+2AB(-m_{i_y}m'_{i_y}+m_{i_y}e'_y+m'_{i_y}e_y+e_ye'_y)\end{aligned}$$

$$\begin{aligned}\frac{\delta J_i}{\delta d} &= \left(-2\frac{(-dm_{i_x}-cm_{i_y}+ce_y+de_x)(-m_{i_x}+e_x)}{((bm_{i_x}+am_{i_y}-ae_y-be_x)^2+(-dm_{i_x}-cm_{i_y}+ce_y+de_x)^2)^2}\right.\\ &\left.-2\frac{(bm'_{i_x}-dm'_{i_y}+de'_y-be'_x)(-m'_{i_y}+e'_y)}{\left((bm'_{i_x}-dm'_{i_y}+de'_y-be'_x)^2+(am'_{i_x}-cm'_{i_y}+ce'_y-ae'_x)^2\right)^2}\right)B^2\\ &+2AB(-m_{i_x}m'_{i_y}+m_{i_x}e'_y+m'_{i_y}e_x-e'_ye_x)\end{aligned}$$

$$\frac{\delta J_i}{\delta e_x} = -\frac{(-2(bm_{i_x} + am_{i_y} - ae_y - be_x)b + 2(-dm_{i_x} - cm_{i_y} + ce_y + de_x)d)\ B^2}{((bm_{i_x} + am_{i_y} - ae_y - be_x)^2 + (-dm_{i_x} - cm_{i_y} + ce_y + de_x)^2)^2} + 2AB(-bm'_{i_x} + dm'_{i_y} - de'_y + be'_x)$$

$$\frac{\delta J_i}{\delta e_y} = -\frac{(-2(bm_{i_x} + am_{i_y} - ae_y - be_x)a + 2(-dm_{i_x} - cm_{i_y} + ce_y + de_x)c)\ B^2}{((bm_{i_x} + am_{i_y} - ae_y - be_x)^2 + (-dm_{i_x} - cm_{i_y} + ce_y + de_x)^2)^2} + 2AB(-am'_{i_x} + cm'_{i_y} + ce'_y + ae'_x)$$

$$\frac{\delta J_i}{\delta e'_x} = -\frac{\left(-2(bm'_{i_x} - dm'_{i_y} + de'_y - be'_x)b - 2(am'_{i_x} - cm'_{i_y} + ce'_y - ae'_x)a\right)\ B^2}{\left((bm'_{i_x} - dm'_{i_y} + de'_y - be'_x)^2 + (am'_{i_x} - cm'_{i_y} + ce'_y - ae'_x)^2\right)^2} + 2AB(-bm_{i_x} - am_{i_y} + ae_y + be_x)$$

$$\frac{\delta J_i}{\delta e'_y} = -\frac{\left(2(bm'_{i_x} - dm'_{i_y} + de'_y - be'_x)d + 2(am'_{i_x} - cm'_{i_y} + ce'_y - ae'_x)c\right)\ B^2}{\left((bm'_{i_x} - dm'_{i_y} + de'_y - be'_x)^2 + (am'_{i_x} - cm'_{i_y} + ce'_y - ae'_x)^2\right)^2} + 2AB(dm_{i_x} + cm_{i_y} + ce_y - de_x)$$

A non-linear optimization only guarantees to optimize around a local minimum. Therefore, great care must be taken when choosing a starting point. The starting point chosen is the one obtained with the linear criterion of Equation (4.12).

4.2.3 Taking into Account Errors in the Correspondence

The matched points used to determine the F matrix can be erroneous; they might be at a wrong location or be a false match. The watershed algorithm is used to find the grid from the gray images (see Section 3.3.1). This algorithm finds the lines and the intersections of the grids, but can have difficulties finding the "true" center of an intersection or a disk. To limit this effect, a method (see Section 3.4) is used but still leaves some errors. The smoothing operator used before the watershed algorithm might oversmooth two lines which would lead the watershed algorithm into generating only one line. This usually leads to false matches (see Section 3.5). These effects must be taken into account when determining the F matrix.

Matching points which have a "bad" location are called outliers. A "bad" location means that one or both of the matched points are believed to be in a certain position in the image. This position does not correspond to either the center of a disk or the center of an intersection in the image. A set of false matched points are also called outliers.

There exists two popular methods to estimate J in order to minimize the effect of outliers: *M-estimators* and the *least-median-of-squares* (LMedS) methods.

Let d_i be the distance between the points m_i, m_i' and their corresponding epipolar line. The standard least-squares method tries to minimize $\sum_i r_i^2$, which is unstable if outliers are present in the data.

The M-estimators replace the above function with

$$\min \sum_i \rho(d_i) \tag{4.19}$$

where ρ is a symmetric, positive-definite function with a unique minimum at zero. The M-estimators can be implemented as a weighted least-square problem. In [33][22], the following weight was used for the estimation of the epipolar geometry

$$w_i = \begin{cases} 1 & |d_i| \leq \sigma \\ \sigma/|d_i| & \sigma < |d_i| \leq 3\sigma \\ 0 & 3\sigma < |d_i| \end{cases} \tag{4.20}$$

where σ is some estimated standard deviation of errors. Faugeras [46] states that the M-estimators method has been found to be robust to outliers due to bad localization and that it isn't robust to outliers due to a false match. The LMedS method is therefore chosen to perform the optimization.

The LMedS method estimates the parameters by solving the nonlinear minimization problem:

$$\min \text{med}_i \, r_i^2 \tag{4.21}$$

This states that the estimator must yield the smallest value for the median of squared distances the F matrix even with the presence of outliers. The LMedS method can not be stated with a straightforward formula. It must be solved by a search in the space of possible estimates generated from the data. For practical reasons, only a randomly chosen subset of data can be analyzed. The algorithm used is based on the one used by Faugeras [46].

Given n correspondences (m_i, m_i'), a Monte Carlo type technique is used to draw m random

subsamples of $p = 8$ ([2]) different corresponding points. For each subsample, indexed by J, a fundamental matrix F_J is determined, then the median of the squared residuals M_J for each F_J found. The M_J is computed in respect with the whole set of point correspondences, i.e.,

$$M_J = \text{med}_{i=1\ldots n}[d^2(m_i', F_J m_i) + d^2(m_i, F_J^T m_i')] \tag{4.22}$$

We retain the estimate F_J for which M_J is minimal among all m M_J's.

A subsample is "good" if it consists of p good correspondences. Assuming that the whole set of correspondences may contain up to a fraction ϵ of outliers, the probability that at least one of the m subsamples is good is given by

$$P = 1 - [1 - (e - \epsilon)^p]^m \tag{4.23}$$

By requiring that P must be close to 1, we can determine the value of m given values for p and ϵ. In the implementation of the proposed algorithm, these values were used: $P = 0.99$, $p = 8$, $\epsilon = 30\%$ and thus $m = 78$.

As noted by Rousseeuw and Leroy [39], the LMedS *efficiency* is poor in the presence of Gaussian noise. The efficiency of a method is defined as the ratio of the lowest achievable variance for the estimated parameters and the actual variance provided by the given method. To compensate for this deficiency, a weighted least-square procedure is used. The *robust standard deviation* estimate is given by

$$\hat{\sigma} = 1.4826[1 + \frac{5}{n - p}]\sqrt{M_J} \tag{4.24}$$

where M_J is the minimal median. Rousseeuw and Leroy [39] give an explanation for the number used. Using σ, we can assign a weight for each correspondence:

$$w_i = \begin{cases} 1 & \text{if } d_i^2 \leq (2.5\hat{\sigma})^2 \\ 0 & \text{otherwise} \end{cases} \tag{4.25}$$

where

$$d_i^2 = d^2(m_i', F_J m_i) + d^2(m_i, F_J^T m_i') \tag{4.26}$$

and $d(\ldots)$ is defined in equation 4.16. The correspondence i having a weight $w_i = 0$ is an outlier and should not be further taken into account for the estimation of F. The fundamental matrix

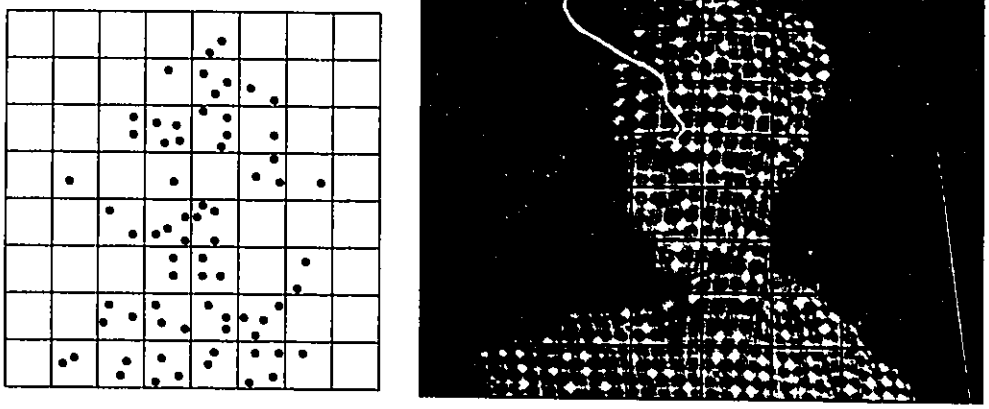
[2]The number 8 is chosen since it's the minimal number of data points necessary to obtain a unique solution for the estimation of F.

F is finally estimated by solving the weighted least-squares problem:

$$\min \sum_i w_i d_i^2 \tag{4.27}$$

As mentioned above, the computational efficiency of the LMedS method can be achieved by applying a Monte Carlo type technique. However, the eight points of a subsample chosen might lie very close to each other. This situation should be avoided because the estimate of F generated by such a subsample would be highly unstable (and the result useless). To avoid such a situation Faugeras [46] developed a *regularly random selection method* based on a bucketing technique.

Bucketing consists of dividing one of the images into $b \times b$ buckets (see figure 4.3). To each bucket, a set of points is attached. The buckets having no set of points are removed. To generate a set of 8 points, 8 buckets are randomly chosen and then a point is randomly selected in each of the selected buckets. In the implementation $b = 8$.



(a) Illustration of a bucketing technique

(b) Example of a sample selected using the bucketing technique

Figure 4.3: The bucketing technique

Equation 4.23 holds if the outliers are uniformly distributed in space and each bucket has the same number of samples. However, the number of matches in one bucket may be different from another one. Therefore a preference to buckets with a high number of samples should be given when deciding from which bucket to choose from. This would give each sample almost the same probability to be selected as any other sample.

The procedure presented above is implemented by the following algorithm. If we have a total of l buckets, we divide the interval $[0, 1]$ into L intervals such that the width of the i^{th} interval is equal to $n_i / \sum_i n_i$ where n_i is the number of sample attached to the i^{th} bucket. During the selection procedure a random number falling in the i^{th} interval is associated with the i^{th} bucket. The procedure is depicted in Figure 4.4.

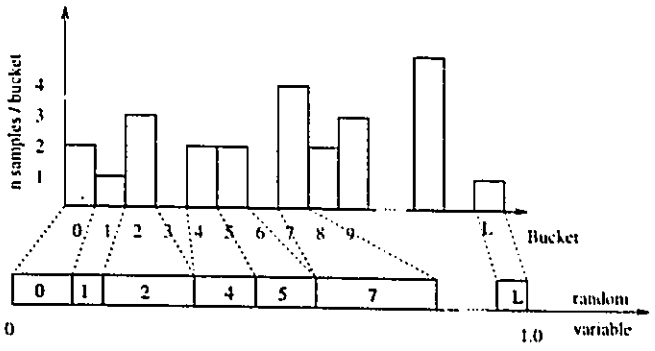


Figure 4.4: Interval and bucket mapping

4.3 The epipolar transformation

The stereo fusion algorithm uses the epipolar constraint to simplify the matching process. There are two different ways to insure that the epipolar constraint holds: (1) match along corresponding epipolar lines or (2) transformation of the images into epipolar images.

Matching along corresponding epipolar lines implies that the stereo fusion process follows the epipolar lines on the original images. This method is not well suited if dynamic programming is used for the stereo fusion algorithm. Here, dynamic programming has been chosen for stereo fusion. Therefore the second method is preferred.

The second method consists of transforming the images so that an horizontal scanline of one of the images is an epipolar line and has a corresponding epipolar line in the scanline of the other image. There are two different methods to transform the images: the use of a normal plane or the epipolar transformation.

The use of a normal plane is the preferred method in the literature [13][8]. This method relies on the knowledge of the relative orientation of both cameras. From this knowledge, a normal

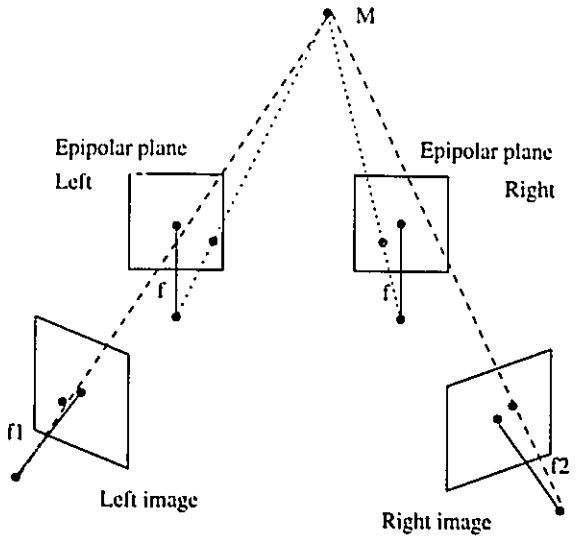
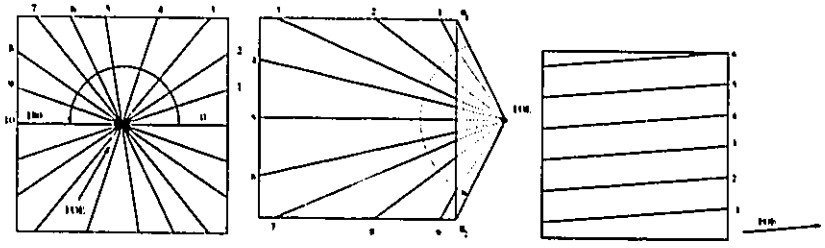


Figure 4.5: Normal plane projection.

plane parallel to both focal points is created. The plane is usually perpendicular to one of the camera's focal axis. The images are projected onto this plane and the projections are now two epipolar images. In Figure 4.5 the method is depicted.

As stated in Section 4.2, the only information available from two uncalibrated images is the fundamental matrix F. We can estimate $[R\_t]$ from F. However the estimation introduces errors (particularly when computing t[24]) and requires a self-calibration of the stereo system using a sequence of stereo images. Since this method introduces errors, it is not used here.



(a) The epipole is inside the image.

(b) The epipole is outside the image.

(c) The epipole is at infinity.

Figure 4.6: The possible locations of the epipole.

The other method to transform images relies only on the knowledge of F. In Section 4.3.1, the

relationship between the matrix F and the epipolar transform is shown. The epipolar images are generated by using this relationship. The transformation consists of drawing each corresponding epipolar line from the left and the right image onto a left and right epipolar image. Three scenarios may arise when choosing the epipolar lines:

1. If the epipole e is in the visible part of the image plane, the selected lines have angles in the interval 0 to 180 degrees (Figure 4.6(a)).

2. If the epipole e is not in the visible part of the image plane and is not at infinity, the selected lines have angles in the interval $[\theta_1, \theta_2]$ where θ_1 and θ_2 are the angles of the lines that are at the extremity of the visible image plane(Figure 4.6(b)).

3. If the epipole e is at infinity, all the selected lines are parallel and oriented toward the epipole (Figure 4.6(c)).

4.3.1 Relationship with the fundamental matrix

The epipolar transformation is a homography between the epipolar lines in the first image and the epipolar lines in the second image. In the practical case where epipoles are at finite distance, the epipolar transformation is characterized by the affine coordinates of the epipoles e and e' and by the coefficients of the homography between the two sets of epipolar lines, each line being parameterized by its direction:

$$\tau \rightarrow \tau' = \frac{a\tau + b}{c\tau + d} \tag{4.28}$$

where

$$\tau = \frac{m_y - e_y}{m_x - e_x} \qquad \tau' = \frac{m'_y - e'_y}{m'_x - e'_x} \tag{4.29}$$

and m and m' is a pair of corresponding points. The coefficient of F can be written in terms of the parameters describing the epipoles and the homography:

$$F_{11} = b \tag{4.30a}$$

$$F_{12} = a \tag{4.30b}$$

$$F_{13} = -ae_y - be_x \tag{4.30c}$$

$$F_{21} = -d \tag{4.30d}$$

$$F_{22} = -c \tag{4.30e}$$

$$F_{23} = ce_y + de_x \tag{4.30f}$$

$$F_{31} = de'_y - be'_x \tag{4.30g}$$

$$F_{32} = ce'_y - ae'_x \tag{4.30h}$$

$$F_{33} = -ce'_y e_y - de'_y e_x + ae_y e'_x + be_x e'_x \tag{4.30i}$$

The parameters of the epipolar transformation can be expressed using the F coefficients:

$$a = F_{12} \tag{4.31a}$$

$$b = F_{11} \tag{4.31b}$$

$$c = -F_{22} \tag{4.31c}$$

$$d = -F_{21} \tag{4.31d}$$

$$e_x = \frac{F_{23}F_{12} - F_{22}F_{13}}{F_{22}F_{11} - F_{21}F_{12}} \tag{4.31e}$$

$$e_y = \frac{F_{13}F_{21} - F_{11}F_{23}}{F_{22}F_{11} - F_{21}F_{12}} \tag{4.31f}$$

$$e'_x = \frac{F_{32}F_{21} - F_{22}F_{31}}{F_{22}F_{11} - F_{21}F_{12}} \tag{4.31g}$$

$$e'_y = \frac{F_{31}F_{12} - F_{11}F_{32}}{F_{22}F_{11} - F_{21}F_{12}} \tag{4.31h}$$

(4.31i)

The determinant $ad - bc$ of the homography is $F_{22}F_{11} - F_{21}F_{12}$. In the case of finite epipoles, it is non zero. In theory, the rank of the F matrix is less or equal to 2. In practice it is equal to 2, since a rank of 1 would imply that all the epipolar lines are the same. Therefore, the determinant of F is 0. In the case of infinite epipoles, τ' can be written as a function of τ from the relation $m'_\infty F m_\infty = 0$ which occurs at the infinity points $m_\infty = [1, \tau, 0]^T$ and $m'_\infty = [1, \tau', 0]^T$ of corresponding epipolar lines to obtain the homographic relation.

4.3.2 Epipolarization

In the control environment of the laboratory, the epipole of the left image is always on the left-end side of the 2D plane. This allows a speed up in the epipolarization process by eliminating the search for θ_1 and θ_2. The size of the images are similar and are of (I_{max}, J_{max}). Using the F matrix, the epipolarization process becomes:

1. $I = 1$

2. $i = I$

   $j = 0$

   $\tau = \frac{i-y}{j-x}$

   $o = y - \tau x$

   $\tau' = \frac{a\tau+b}{c\tau+d}$

   $o' = y' - \tau' x'$

3. $J = 1$

4. $leftEpipolarImage(I, J) = leftImage(i, j)$

   $j' = j$

   $i' = j'\tau + o'$

   $rightEpipolarImage(I, J) = rightImage(i', j')$

   $j = j = j + 1$

   $i = j\tau + o$

5. $J = J + 1$

6. Goto 4, unless $J = J_{max}$

7. $I = I + 1$

8. Goto 2, unless $I = I_{max}$

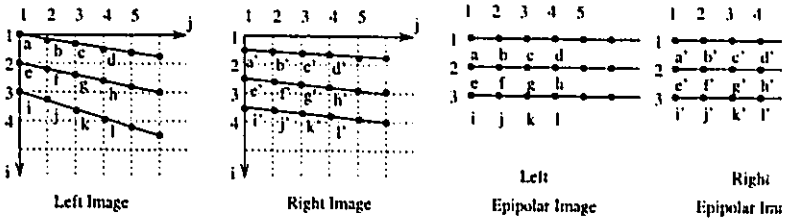


Figure 4.7: Generating epipolar images.

The method is graphically represented in Figure 4.7. The value of (i, j) and (i', j') might not be decimal, thus linear interpolation must be performed to obtain a value at that point. However, this method requires a minimal use of linear interpolation since it is only required in

the y direction. In the horizontal direction, the increment is always fixed to 1 and therefore doesn't requires any linear interpolation.

The de-epipolarization is the same operation as the epipolarization except the assignment is made in the other direction. Therefore, replace $leftEpipolarImage(I, J) = leftImage(i, j)$ with $leftImage(i, j) = leftEpipolarImage(I, J)$ and replace $rightEpipolarImage(I, J) = rightImage(i, j)$ with $rightImage(i, j) = rightEpipolarImage(I, J)$ to obtain the de-epipolarization operation.

For the de-epipolarization, a linear interpolation can't be performed. Instead a value is assigned to a point if it lies within 1/2 pixel from (i, j).

4.4 Conclusion

The epipolar geometry described in this chapter gives the relationship between two views of the same object. A non-linear method to estimate the F matrix (which fully represents the epipolar geometry) is outlined. The estimation process uses the list of matching points provided by a method elaborated in the previous chapter. An epipolarization method to generate epipolar images from the estimated F matrix is also described. This method is used to simplify the stereo fusion problem which is described in the following chapter.

Chapter 5

Stereo fusion and 3D reconstruction

Stereo fusion consists of matching each pixel from the left image to one from the right image. This matching generates a disparity value for the pixels of the left image. The disparity value shows the distance between a pixel (a point) in the left image and its corresponding pixel in the right image along their epipolar line. The method used here to compute the disparity image is dynamic programming (DP). It assumes that the images have epipolar scanlines. The disparity of a point then becomes an horizontal shift from the left image to the right image.

In this chapter, a description of how dynamic programming is used to generate a disparity image and how the disparity image is related to the 3D locations of the points in the image is given. Improvements made over an existing implementation of the dynamic programming method are also described.

5.1 Dynamic Programming

Dynamic programming (DP) is a technique to solve an n-variable optimization problem under the assumption of optimality as n-cascaded mono-variable optimization problems [5]. In our case DP is used to find the optimal path in a graph where that optimal path corresponds to the matching of two epipolar lines. DP is expressed in a recursive form as

$$f_t(i) = \min\{(\text{cost during stage } t) + f_{t+1}(\text{new state at stage } t+1)\} \tag{5.1}$$

The minimum of Equation (5.1) is formed over all possible decisions made when the state at stage t is i. $f_t(i)$ is the minimum cost found at stage t to the end of the problem, given that at

stage t the state is i.

If we put the 2 epipolar lines into a graph form, where the x axis corresponds to the epipolar line of image 1 (the left image) and the y axis corresponds to the epipolar line of image 2 (the right image) , the matching process corresponds to finding a path joining the points $(0,0)$ and (i_{max}, j_{max}). A point (i,j) along the path means that the point at index i of the epipolar line of image 1 corresponds to the point at index j of the epipolar line of image 2. When an occlusion in one of the images occurs it corresponds to a vertical or an horizontal portion on the path. For this case, DP is used to determine the optimal path.

Roy [40] suggests the following method to determine the optimal path using DP. Let $I_1(i)$ and $I_2(j)$ be the intensity level of the epipolar line at index i and j for the image 1 and 2 respectively. The cost to match the point i from image 1 to point j in image 2 is defined as the difference in illumination between the two points:

$$B(i,j) = |I_1(i) - I_2(j)| \tag{5.2}$$

Letting t_n define the location of a point along the path after n stages. We define $C_n(t_n)$ as the total cost of a path $t_n \ldots T$ at stage t (T defines the end stage). The total cost is defined as

$$C_n(t_n) = C_{n+1}(t_{n+1}) + B(t_n) + \text{OccPenality}(t_n, t_{n+1}) \tag{5.3}$$

where

$$\text{OccPenality}(t_n, t_{n+1}) = \begin{cases} 0 & \text{if } t_n - t_{n+1} = (-1,-1) \\ 1 & \text{otherwise} \end{cases} \tag{5.4}$$

The occlusion penalty $\text{OccPenality}(t_n, t_{n+1})$ is an extra cost added only when the preceding match along the path creates an occlusion (*i.e.* horizontal or vertical path segments). This factor is very low, but can be increased when the images are expected to be highly corrupted by noise.

Roy [40] defines a relationship between t_n and t_{n+1}. The successive locations on a path must be at $(-1,0)$, $(0,-1)$ or $(-1,-1)$ apart from each other. This is represented graphically in Figure 5.1

The relationship between successive t_n in the path is true if the following criteria are met:

- Opaque objects: A point considered visible can only match one point. When it matches more than one point, it is considered occluded. The problem of transparency detection is avoided.

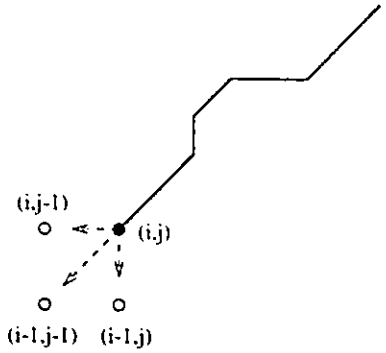


Figure 5.1: The potential directions a path can take.

- Continuous objects: The points in the left image correspond to points in the right image in the same sequence. In Figure 5.2 a situation where this is not true is depicted: the sequence of the points a and b along the epipolar line is broken.

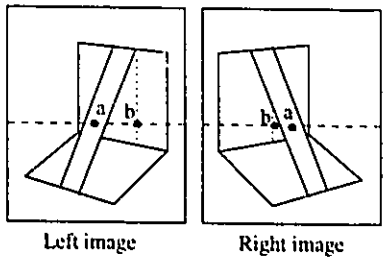


Figure 5.2: A situation where the object is not continuous.

Roy's method finds the optimal solution for the path going from $(0,0)$ to (i_{max}, j_{max}), where i_{max} is the size of the left epipolar line and j_{max} is the size of the right epipolar line.

A disparity value is obtained for all the points in the epipolar line by looking at the optimal path. Any points i of the left epipolar line is assigned a disparity value of $j - i$ if it is not part of an occluded region of the path (a vertical or an horizontal segment). If a point belongs to an occluded region the disparity value is set at 0.

5.1.1 DP based on a coarse disparity

The cost function used by Roy has a few limitations. The path always goes from the point $(0,0)$ to (i_{max}, j_{max}). And Roy's cost function doesn't take into account any a priori knowledge about the disparity. Therefore, the disparity obtained by his method can be improved by taking into account the disparity found from the PRBA and limiting the search space using the background information.

In Section 3.6 a coarse disparity is obtained by matching the PRBA information from the two images. However, this coarse disparity image includes outliers (see Section 4.2.3). A new disparity image can be generated when we eliminate the outliers. Let $D(u,v)$ be the disparity at location (u,v) of the left image obtained by eliminating the outliers. Let $d(i)$ correspond to $D(line,i)$ where *line* is the current epipolar line for which an optimal path is sought. The cost $B(i,j)$ associated to the location (i,j) in the path can be written to be

$$B(i,j) = \begin{cases} |I_1(i) - I_2(j)| & \text{if } d(i) = 0 \\ |I_1(i) - I_2(j)| + occ\,||d(i)| - |j-i|| & \text{if } |d(i)| > 0 \end{cases} \tag{5.5}$$

where *occ* is a factor to increase the weight of the difference between the disparity $j-i$ and the coarse disparity $D(row,i)$. The *occ* factor has the same value than the one associated with the occlusion penalty which is described next.

Roy's cost function is also modified to give a non-linear cost for the occlusions. The cost function used is

$$C_n(t_n) = \min_{t_{n+1}} C_{n+1}(t_{n+1}) + B(t_n) + OccPenality(t_n, t_{n+1}, t_{n+2}, \ldots) \tag{5.6}$$

where t_{n+1} obeys the following relationship: $t_n - t_{n+1} = (-1,0), (0,-1) \text{or} (-1,-1)$ and

$$OccPenality(t_n, t_{n+1}, t_{n+2}, \ldots) = \begin{cases} 0 & \text{If } t_n - t_{n+1} = (-1,-1) \\ M \times occ & \text{If } t_{n+m} - t_{n+m+1} = (-1,0) \text{or} (0,-1) \text{ for } m = 0 \ldots M \end{cases} \tag{5.7}$$

and *occ* is the value of the occlusion penalty. In our implementation, the value chosen for *occ* is 1.5.

The last method to improve the results is to limit the search space. If we use the foreground information obtained in Section 3.3.2 we can delimit a region for which the disparity is sought.

Let b_1 and B_1 be the starting and ending locations of the foreground object on the epipolar line of image 1, and let b_2 and B_2 be the starting and ending locations of the foreground object

on the epipolar line of image 2. The optimal solution is then the path going from (b_1, b_2) to (B_1, B_2).

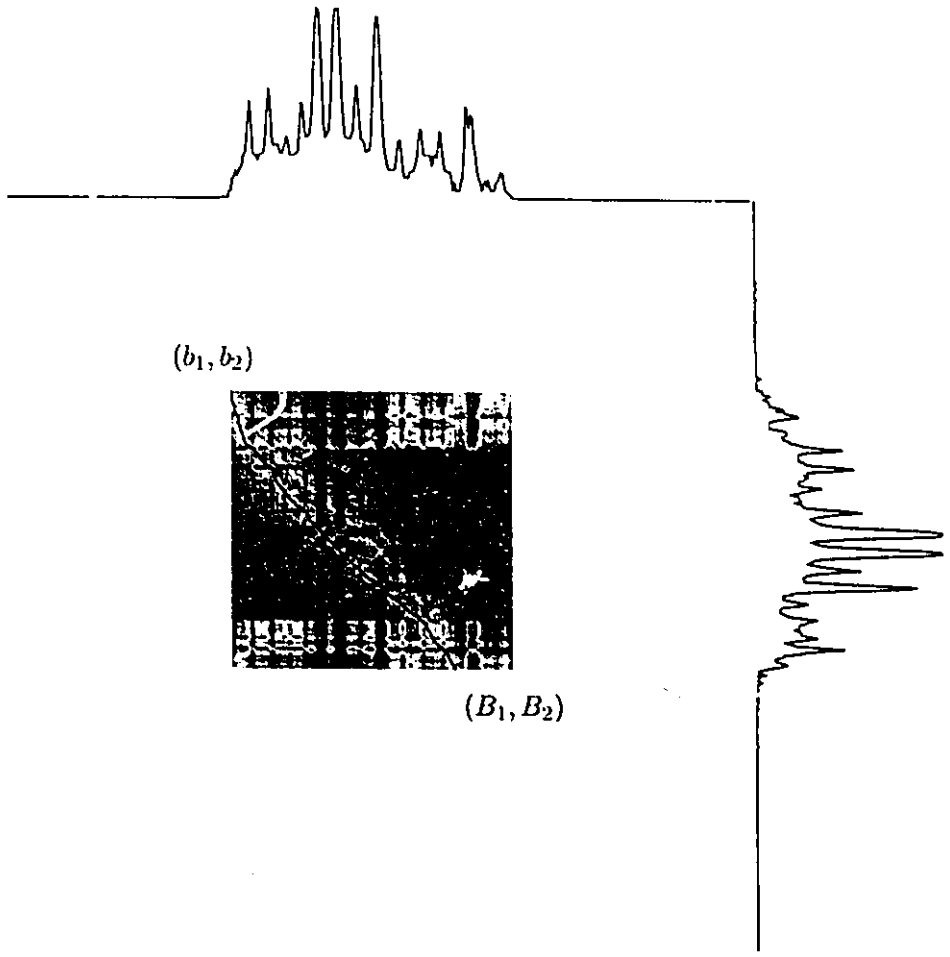


Figure 5.3: Illustration of the optimal path found using the proposed DP cost function.

The Figure 5.3 shows the optimal path found for the path going from (b_1, b_2) to (B_1, B_2) using the proposed DP algorithm. The cost function $B(i, j)$ is shown for the region of the search space. Darker regions indicate a higher cost then lighter regions. The left scanline is the one on the right of the figure, the right scanline is on top of the figure.

5.2 Autoregressive modeling

The DP algorithm gives a good matching of the epipolar lines, however it is not perfect. Some points are not matched and are seen as occluded points. These points are left "blank", i.e. without any disparity value. In our case this is mainly due to the saturation of the image due to the projection of the grid pattern onto the object. One way to effectively "fill in the blanks" is to use an Autoregressive (AR) model.

Maitre and Wu [27] describe an algorithm which uses AR modeling to register a picture and a map. The algorithm is more powerful since both DP and AR are used. The DP ability to tolerate local distortions is used in the identification process of AR models. And the prediction error of AR models is used as a local dissimilarity measure in the evaluation of the path cost in DP. Their algorithm does not use any *a priori* knowledge or model about the deformation between the picture and the map.

Several factors limit their algorithm for a stereo-fusion purpose:

- They assume that a global deformation is performed on the picture. The deformation between two epipolar lines are not global, they're usually a succession of local deformations along the line.

- They assumed that they can register it to a "perfect" map. They assumed that occlusions only occur in the picture to be registered. This is not the case for two epipolar lines since occlusions can occur in both images.

However, their merit is the introduction of the idea of using an AR model for measuring the deformation function when matching two images. This method can be applied to model the deformations from a pixel in the left image to the one in the right image to match two epipolar lines.

5.2.1 The Auto regressive model

The auto regressive model of order p of a discrete process $s(i)$ is represented by

$$s(i) = -\sum_{k=1}^{p} a(p,k)s(i-k) + e(i) \tag{5.8}$$

where $e(i)$ is a white noise process. The Burg method can be used to estimate the AR model.

The Burg method consists of estimating the reflection coefficients and then using the Levinson recursion to obtain the AR parameter estimates (see [16]). This method can be summarized as

$$f_{0,k} = x_k \quad k = 1,..N$$

$$b_{0,k} = x_k \quad k = 1,..N$$

$$e_0 = 2\sum_{k=1}^{N} |x_k|^2$$

$$a_{0,0} = \frac{-2\sum_{k=1}^{N} b^*_{m-1,k} f_{m-1,k+1}}{\sum_{k=1}^{ii}(|f_{m-\ ,k+1}|^2 + |b^*_{m-1,k}|^2)} = -1$$

$$\mathrm{DEN}_0 = \sum_{k=1}^{N} |x_k|^2$$

$$g = 1$$

For $m = 1, \ldots, p$

$$f_{m,k} = \textstyle\sum_{i=0}^{m} a_{m,i} x_{k+m-i} \quad k = 1, \ldots, N-m$$

$$b_{m,k} = \textstyle\sum_{i=0}^{m} a^*_{m,i} x_{k+i} \quad k = 1, \ldots, N-m$$

$$\mathrm{NUM} = \sum_{k=1}^{N-m} b^*_{m-1,k} f_{m-1,k+1}$$

$$\mathrm{DEN}_m = \mathrm{DEN}_{m-1}\, g - |b_{m-1,N-m+1}|^2 - |f_{m-1,1}|^2$$

$$a_{m,m} = -2\frac{\mathrm{NUM}}{\mathrm{DEN}_m}$$

$$g = 1 - |a_{m,m}|^2$$

$$e_m = e_{m-1}\, g$$

$$a_{m,i} = a_{m-1,i} + a_{m,m} a^*_{m-1,m-i} \quad \text{for } i = 1, \ldots, m-1$$

where N is the number of data, e_m is the error and a is the AR model.

5.2.2 Describing the process

The AR model must characterize the deformations between two epipolar lines. A disparity vector representing the transformation between a point in the left epipolar line and a point in the right epipolar line performs such a characterization. Since a 2D vector can be represented by a complex number, the disparity vector between point i and j is defined as

$$s(i,j) = j - i + \vec{i}(I_2(j) - I_1(i)) \tag{5.9}$$

Picinbono and Bouvet [35] have given a precise discussion of the relationship between complex signals and real vectors of dimension 2. They have shown that a complex AR model is equivalent to a real vector AR model if the complex model is not only a white noise but also analytical. If the noise is not analytical, then the prediction of a complex AR model is only an approximation of the optimal vectorial one.

The complex approximation of the vector might not yield an optimal solution, however it reduces the computational complexity of the modeling process and is thus chosen.

The noise present in the equation of the AR process (Equation (5.8)) has to be wide sense stationary to be valid. The process of the unknown deformation is, however, very unlikely to be stationary. Some methods to relax the stationary constraint imposed by AR exist. Grenier [11] summarizes the following four major techniques to model non-stationary problems:

1. adaptive methods where the model is updated every time a new realization is available so that it dynamically describes the trends of the process;

2. evolutive methods where the model coefficients are expressed on a time function basis where the non-stationary behavior is held by the functions which are chosen to fit the problem at hand;

3. random coefficient methods where the model is divided into two levels, one being a classical model for generating the process, the other representing the model coefficients by another random process; and

4. piecewise stationary methods where the non-stationary process is cut into several segments, each of them being stationary.

The last method was chosen since it offers a low computational cost. Experiments by Maître and Wu [27] suggest that a segment of size $L = 10$ gives adequate results in conjunction of a model of size $p = 3$. This suggestion was tested here, and found to be true for the modeling of two epipolar lines.

5.2.3 Improving DP results

Maître and Wu use the AR model as a step in the DP process, but here it is used to improve the results obtained by the DP algorithm; it is not part of the DP process. The improvement consists in finding if a point defined by the optimal path as occluded is really occluded. The AR process tries to "fill in the blanks".

This is achieved by modeling the disparity from the left and from the right of an occluded point. The same procedure applies when modeling from the left or to the right of the occluded point. The procedure described here is the one modeling from the left of the occluded point.

An AR model is estimated using $L = 10$ data points close to an occlusion point by following Burg's method and a new disparity vector is estimated. The estimation of the disparity vector is given by

$$\hat{s}(i) = -\sum_{k=1}^{p} a(p,k)s(i-k) \tag{5.10}$$

$\hat{s}(i)$ is accepted as a valid estimate of the disparity vector if

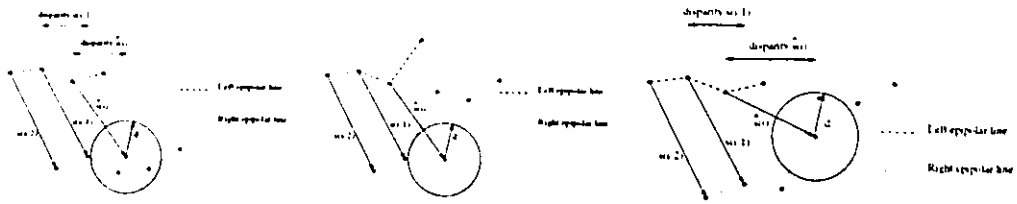
1. A real point lies within a distance d of the estimate.

2. The estimate's disparity is close to the disparity of the last data point. In the implementation, a value greater than 4 between the two values was considered too far.

3. The closest point to the estimate is not already matched to another point.

If $\hat{s}(i)$ is considered a valid estimate, then the point closest to it is selected as a matching point. The occlusion point is than replaced with the disparity given by the match. If another occlusion point is present next to the new matched point, a new AR model is computed using the last point found. The old model is *not* reused to compute an estimate.

In Figure 5.4, three different scenarios are described. In (a), the estimate is valid. In (b) the estimate is not valid because it is too far from any real points. Finally, in (c) the estimate is not valid because the disparity of $\hat{s}(i)$ is too big compared to the disparity of $s(i-1)$.



(a) The estimate is valid

(b) The estimate is not valid due to the distance d.

(c) The estimate is not valid due to the disparity

Figure 5.4: Possible scenarios for the estimate $\hat{s}(i)$.

This technique tries only to find an AR model if there are enough disparity points next to an occlusion point. However, a region with enough disparity points can fill the gap between itself and a region with not enough disparity points. This creates a bigger region which has now enough successive disparity points to compute an AR model.

5.3 3D reconstruction

The pinhole model shown in Section 4.1 gives a relation between a 3D point M and it's corresponding point m of the camera retina. Knowing the disparity vector between two points, the 3D location of the point 3D can be estimated using the relation of Equation (4.6). The simple case of a conventional stereo setup is described in Section 2.1.

In a conventional stereo setup, the equation relating the point M and its retina view in the left image can be written, using the pinhole model, as:

$$s_1 \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{5.11}$$

thus,

$$s_1 \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} = \begin{bmatrix} fX \\ fY \\ Z \end{bmatrix} \tag{5.12}$$

which gives

$$u_1 = fX/Z \tag{5.13a}$$

$$v_1 = fY/Z \tag{5.13b}$$

and

$$X = u_1 Z/f \tag{5.14a}$$

$$Y = v_1 Z/f \tag{5.14b}$$

For the right image of a conventional stereo setup, using the pinhole model the following relation holds

$$X + b = u_2 Z/f \tag{5.15a}$$

$$Y = v_2 Z/f \tag{5.15b}$$

The equations (5.14) and (5.15) can be combined into

$$X + b - X = u_2 Z/f - u_1 Z/f \tag{5.16a}$$

$$Z = b - \frac{f}{u_2 - u_1} \tag{5.16b}$$

$$\tag{5.16c}$$

therefore,

$$Z = b - \frac{f}{d} \tag{5.17}$$

where d is the disparity.

Equation (5.17) gives the direct correspondence between the disparity and the depth of a point for a conventional stereo system. In the case of a non-parallel axis system, the pinhole model gives the following correspondence

$$s\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} -fk_u & fk_u cot\theta & u_0 \\ 0 & -\frac{fk_v}{sin\theta} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{5.18}$$

which expands to

$$\begin{aligned} su &= \begin{array}{l} ((-u\,r_{11} + ku\,\cot(\theta)\,r_{21})\,f + u_0\,r_{31})\,X \\ +\,((-u\,r_{12} + ku\,\cot(\theta)\,r_{22})\,f + u_0\,r_{32})\,Y \\ +\,((-u\,r_{13} + ku\,\cot(\theta)\,r_{23})\,f + u_0\,r_{33})\,Z \\ +\,(-u\,tx + ku\,\cot(\theta)\,ty)\,f + u_0\,tz \end{array} && (5.19a) \\ sv &= \left(-\tfrac{kv\,r_{21}\,f}{\sin(\theta)} + v_0\,r_{31}\right)X + \left(-\tfrac{kv\,r_{22}\,f}{\sin(\theta)} + v_0\,r_{32}\right)Y \\ &\quad + \left(-\tfrac{kv\,r_{23}\,f}{\sin(\theta)} + v_0\,r_{33}\right)Z - \tfrac{kv\,ty\,f}{\sin(\theta)} + v_0\,tz && (5.19b) \\ s &= r_{31}\,X + r_{32}\,Y + r_{33}\,Z + tz && (5.19c) \end{aligned}$$

hence,

$$X = -\frac{\begin{array}{l}((r_{32} + f\,r_{12})\,Y + (r_{33} + f\,r_{13})\,Z + tz + f\,tx)\,u + (-f\,ku\,\cot(\theta)\,r_{22} - u_0\,r_{32})\,Y \\ +\,(-u_0\,r_{33} - f\,ku\,\cot(\theta)\,r_{23})\,Z - u_0\,tz - f\,ku\,\cot(\theta)\,ty\end{array}}{(r_{31} + f\,r_{11})\,u - f\,ku\,\cot(\theta)\,r_{21} - u_0\,r_{31}} \quad (5.20a)$$

$$Y = -\frac{\begin{array}{l}(r_{33}\,Z + tz + r_{31}\,X)\,v + \left(\tfrac{kv\,r_{21}\,f}{\sin(\theta)} - v_0\,r_{31}\right)X + \left(\tfrac{kv\,r_{23}\,f}{\sin(\theta)} - v_0\,r_{33}\right)Z \\ -\,v_0\,tz + \tfrac{kv\,ty\,f}{\sin(\theta)}\end{array}}{v\,r_{32} + \tfrac{kv\,r_{22}\,f}{\sin(\theta)} - v_0\,r_{32}} \quad (5.20b)$$

In the case of the left image, the following relation applies

$$X = -\frac{(u_1 - u_0)\,Z}{f\,u_1} + \frac{Y\,ku\,\cot(\theta)}{u_1} \quad (5.21a)$$

$$Y = -\frac{(v_1 - v_0)\,\sin(\theta)\,Z}{f\,kv} \quad (5.21b)$$

and in the case of the right image, X and Y are defined by Equations (5.20). Using these equations, a value of Z can be obtained.

Without the calibration of the cameras, locating the exact 3D depth of a point is infeasible. Hence, for this thesis, only the disparity images are shown.

5.4 Conclusion

In this chapter, a description of the stereo fusion algorithm used with two epipolar images is presented.

The DP process is improved over other works using the information of a coarse disparity. This coarse disparity is made out of the points which were not discarded by the LMedS method when estimating the F matrix (see Section 4.2.3).

A method to improve the results obtained using AR modeling is also described. Finally, the relationship between disparity and 3D location is shown.

Chapter 6

Results and methodology

In the previous chapters, the elements of an algorithm required to generate a disparity map from two uncalibrated 2D images are described. In this chapter, the algorithm is presented along with results obtained for different test cases. A discussion of the effects of varying certain parameters will also be presented.

The implementation of this algorithm was done using the KHOROS environment. This environment is well suited for the creation and testing of the different steps required to generate a disparity map.

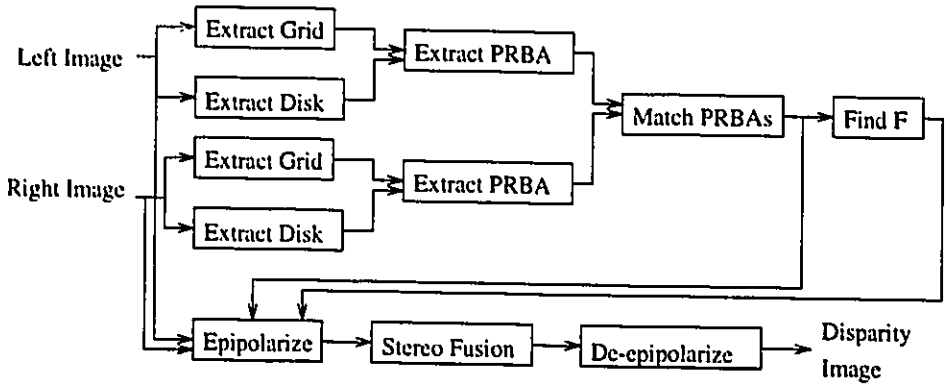


Figure 6.1: The overall algorithm.

The overall approach consists of the following steps:

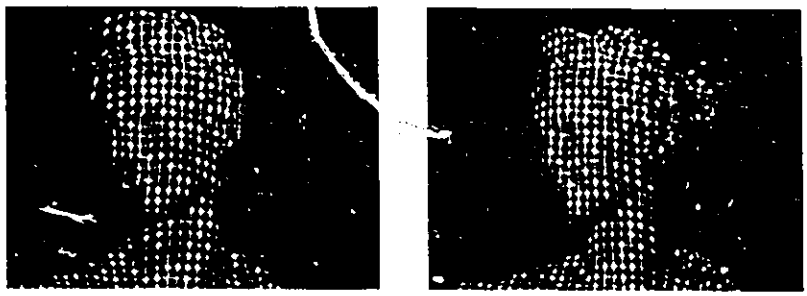
- The extraction the PRBA from both images (Section 3.5).

- The match of the two PRBAs obtained and generation of a coarse disparity map and of a list of matching points (Section 3.6).
- The estimation F (Section 4.2.3).
- The generation of epipolar images (Section 4.3.2).
- The generation a disparity map using DP and AR modeling (Section 5.1.1 and Section 5.2).
- The de-epipolarization of the disparity image obtained (Section 4.3.2).

The general approach is shown in Figure 6.1.

Four test cases are shown in this chapter. For the first case, details on the effects of varying some of the parameters are given.

6.1 Test Case I

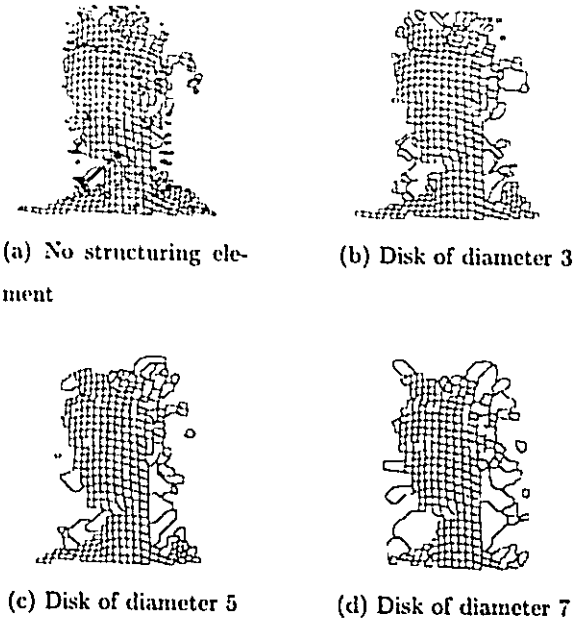


(a) Log of the left image. (b) Log of the right image.

Figure 6.2: Left and right test images for case I.

The left and right input images for the first test case are shown in Figure 6.2. For this case, the cameras were installed on each side of the projector. They are tilted a bit upward and have a small inward pan to look at the object placed directly in front of the projector.

The grid extraction process involves the smoothing and the watershed algorithms. The smoothing consist of the closing operator with a disk structuring element. Changing the size of the element affects the result of the watershed algorithm. A small disk element leaves small noisy areas in the images. A larger disk tends to oversmooth and removes some of the details



(a) No structuring element

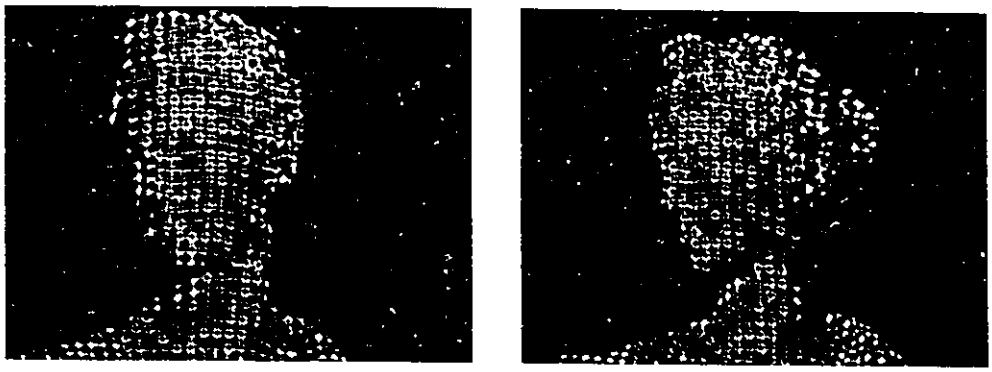
(b) Disk of diameter 3

(c) Disk of diameter 5

(d) Disk of diameter 7

Figure 6.3: The effects of varying the disk structuring element for the grid extraction process of the right image.

of the grid projected onto the object. In Figure 6.3, the effects of varying the size of the disk structuring element is shown when trying to extract the grid of the right image.



(a) Left PRBA (b) Right PRBA

Figure 6.4: The PRBA extracted for the left and right image.

The PRBA extracted for the left and the right images is shown in Figure 6.4. On the right image, only one error has been made and it is located on the left side of the forehead. On the left image there are three problem areas: the lower lip were an error was made, the lower left shoulder were a small square is badly located and above the ear where most points to the right are false matches.

The F matrix is then estimated. A linear estimation is first computed to generate a starting point for the LMedS non-linear optimization. The non-linear optimization is then performed. In the implementation, this function was optimized using the LMedS method with the CFSQP library

$$\begin{array}{ll} \text{Minimize} & J = \sum_i w_i d_i^2 \\ \text{Subject to} & |det(F)| < 0.1 \end{array} \tag{6.1}$$

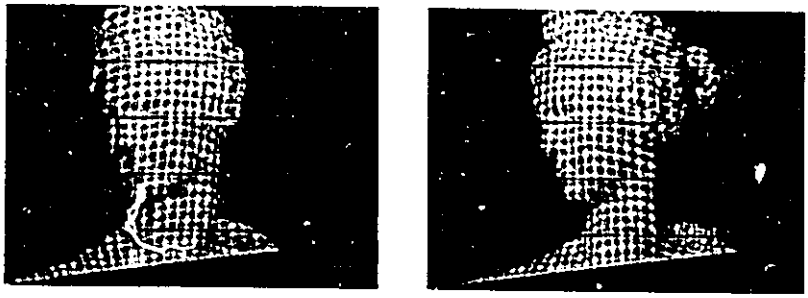
The LMedS methods requires an estimation of F_j for m samples and the estimation of F once the outliers are eliminated (see Section 4.2.3). When estimating F_j, 250 iterations are performed and when estimating F, 1000 iterations are used. Increasing the number of iterations required to estimate F_i influences the quality of the estimation of F. The number 250 was chosen since it gives a good balance between speed and performance.

The Table 6.1 gives the number of point matched from the two PRBAs, the value of J

Total matches	Linear J	LMedS matches	Non-linear J	Average d per match
239	2.0231×10^7	222	145.98	0.3288

Table 6.1: Results of case I for the estimation of F.

obtained with the linear estimation, the number of matches remaining after elimination with the LMedS method, the result for J obtained with the non-linear optimization and finally the average distance between a point not eliminated by the LMedS method and its corresponding epipolar line.



(a) The log of the left epipolar image.

(b) The log of the right epipolar image.

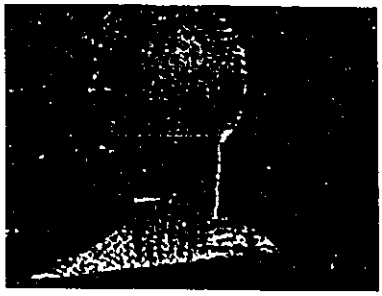
Figure 6.5: Epipolar images for case I.

The epipolar images, as shown in Figure 6.5 are generated using the estimated F matrix. They are then processed using the dynamic programming algorithm.

The cost function used for DP is an improvement over the one made by Roy [40]. Roy's method obtains the disparity map shown in Figure 6.6(a). The improved DP yields the disparity map shown in Figure 6.6(b). The effect of using AR to "fill-in-the-blank" is shown in Figure 6.6(c). The occluded points seen in the middle of the statue correspond mostly to points lying on the projected grid.

In figure 6.6 comparative results of applying the DP method as proposed in this thesis and as used by Roy [40] are shown. The difference between two epipolar lines can be observed in the Figure 6.7(a) on the scanline of a segment obtained from the left epipolar image and Figure 6.7(b)

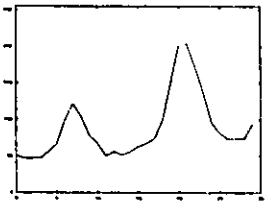
(a) The disparity obtained with Roy's method.



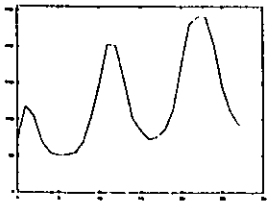
(b) The disparity obtained using only the proposed DP.

(c) The disparity obtained using the proposed DP and AR modeling.

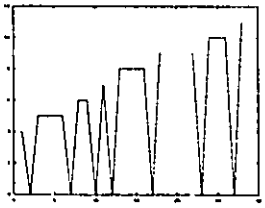
Figure 6.6: Generating the epipolar disparity map.



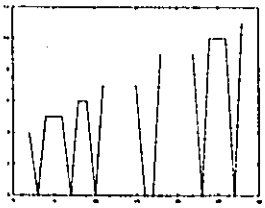
(a) The left scanline



(b) The right scanline



(c) The disparity for the method without AR modeling.



(d) The disparity obtained by Roy

Figure 6.7: The disparity for a small portion of the epipolar images.

compared with a similar segment obtained from the right epipolar image. The segments seen are the ones going from the location (150, 100) to the location (180, 100) of the epipolar images. The disparity values computed using the DP method proposed here, without using AR modeling are shown in Figure 6.7(c). The ones obtained by Roy's method are shown in Figure 6.7(d). As can be seen from these two figures, the two methods are different. The differences arise at index location 2,3,11,12,13,14,15 and 16.

The Figure 6.8(a) and Figure 6.8(b) shows the matching of the two scanlines using the proposed method and using Roy's method. The series of differences, listed above, have the following effect: the proposed method distribute more evenly the disparity for the region starting at index location 11 and ending at location 16.

Finally the disparity in Figure 6.6(c) is de-epipolarized to yield a scaled version of the 3D locations of the points in the left image. The result of this operation is shown in Figure 6.9.

6.2 Test Case II

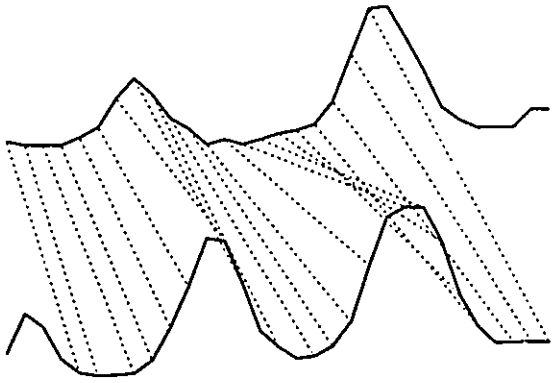
The left and right input images for the second test case are shown in Figure 6.10. As in the first case, the cameras were installed on each side of the projector. The left camera was rotated to center the object more than the first case. The right camera was located a little closer to the projector than the one in the first case.

In Figure 6.11, the PRBA extracted from each image is overlapped to the log of their respective image. The left image has only one false match located in the lower right portion of the neck. The right image contains no false match errors. However, the location of the intersection points in the left and right image seem to be less centered than the one seen in case I.

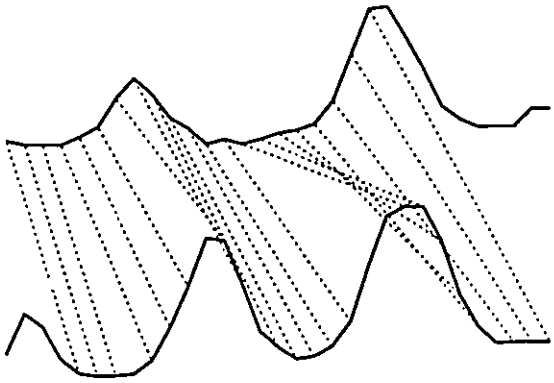
Total matches	Linear J	LMedS matches	Non-linear J	Average d per match
226	4.91412×10^6	209	208.021	0.49765

Table 6.2: Results of case II for the estimation of F.

The matching of both PRBAs yields 226 matches. Of these matches, 17 are found to be outliers by the LMedS method. The estimation of F yields an average distance between a point



(a) Matching using the proposed method without AR modeling



(b) Matching using Roy's method

Figure 6.8: Matching two segments.

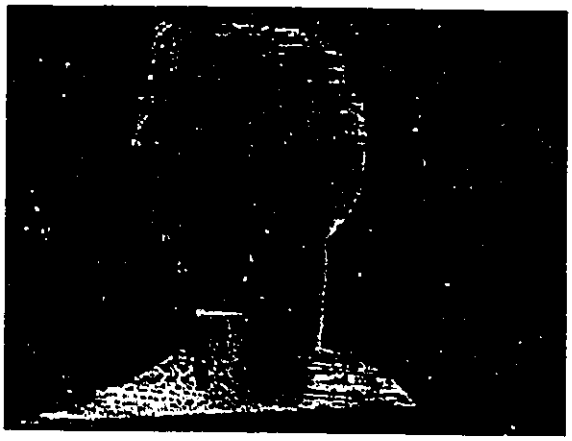
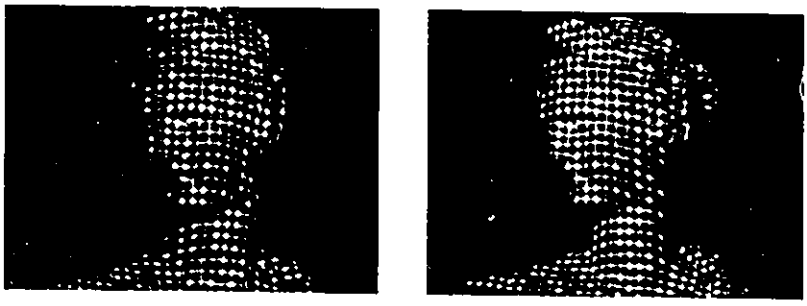


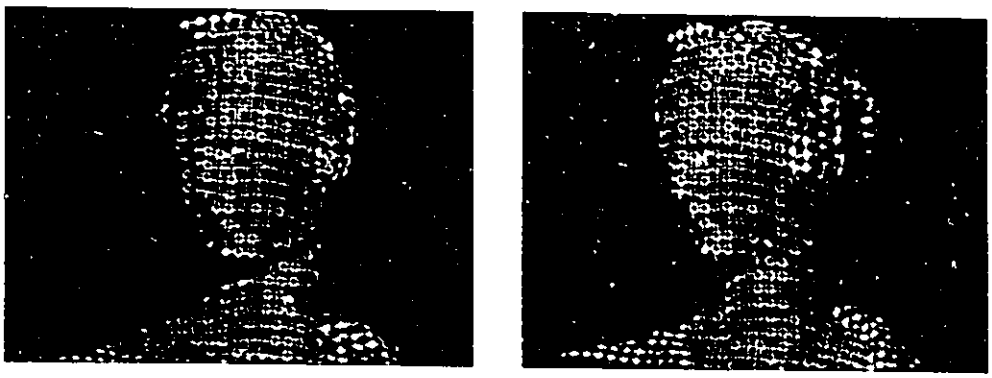
Figure 6.9: The disparity map for case I.



(a) Log of the left image.

(b) Log of the right image.

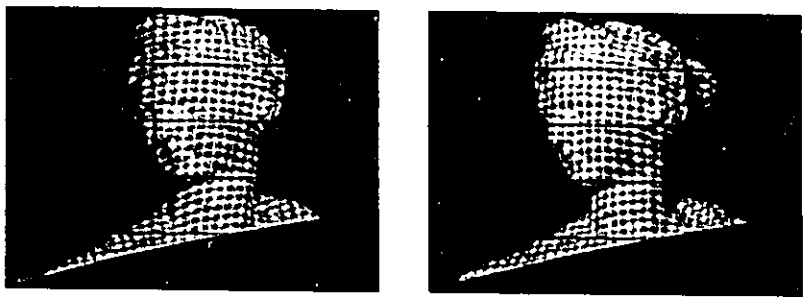
Figure 6.10: Left and right test images for case II.



(a) Left PRBA (b) Right PRBA

Figure 6.11: The PRBA extracted for the left and right image in case II.

and its epipolar line of 0.49765. This distance is mainly due to bad localization errors.



(a) The log of the left epipolar image. (b) The log of the right epipolar image.

Figure 6.12: Epipolar images for case II.

The epipolar images generated are shown in Figure 6.12. An average for d of 0.49765 does not affect the quality of the epipolarization process. The epipolar constraint can be assumed valid.

The disparity map (shown in Figure 6.13) shows the drawbacks of the epipolarization method used here (see Section 4.3.2). The epipolarization process takes all the epipolar lines that start on the left side. This might yield a situation like the one depicted in Figure 6.14.

In this situation, the epipolar lines are going apart from each other near the right end. This creates points for which there is no epipolar line going within a 1/2 pixel above or below

Figure 6.13: The disparity map for case II.

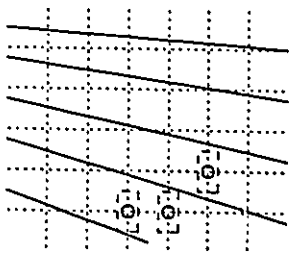
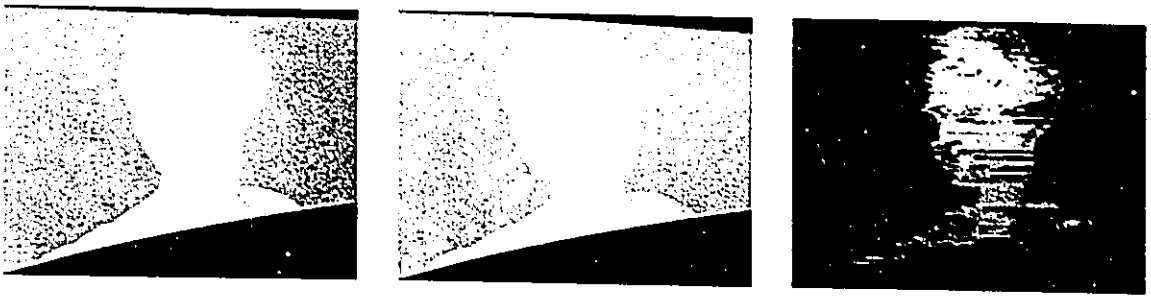


Figure 6.14: The effects of de-epipolarization.

them. These points are circled in the Figure 6.14. In the disparity image obtained for this case (Figure 6.13), the effect of these points is clearly seen.



(a) The left epipolar image with no PRBA.

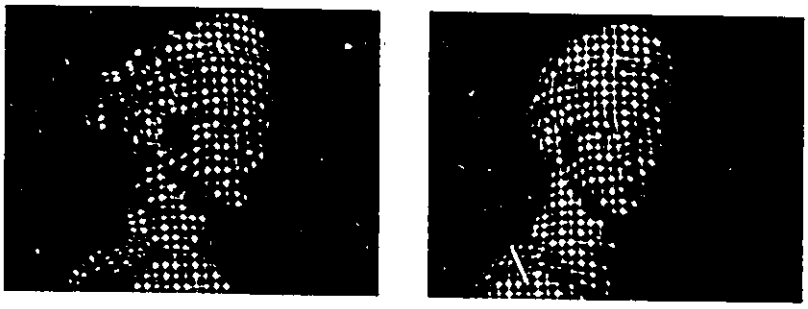
(b) The right epipolar image with no PRBA.

(c) The disparity computed.

Figure 6.15: The disparity of the object without a PRBA

The projected PRBA adds a texture to the object. This texture allows the stereo fusion algorithm to establish an unambiguous correspondence between the points of the two epipolar images. Figure 6.15 shows how the proposed DP algorithm behaves when the two images of the same object do not have a PRBA projected onto them. This clearly demonstrate the usefulness of the PRBA not only for the estimation of the F matrix but also for the stereo fusion algorithm itself.

6.3 Test Case III

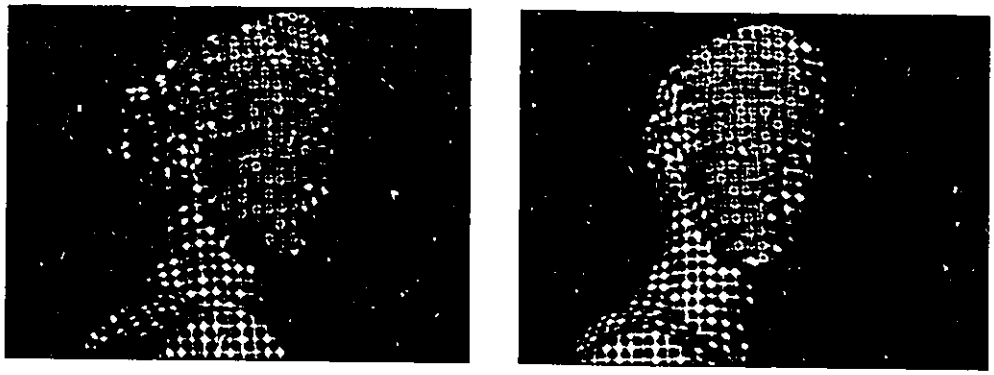


(a) Log of the left image.

(b) Log of the right image.

Figure 6.16: Left and right test images for case III.

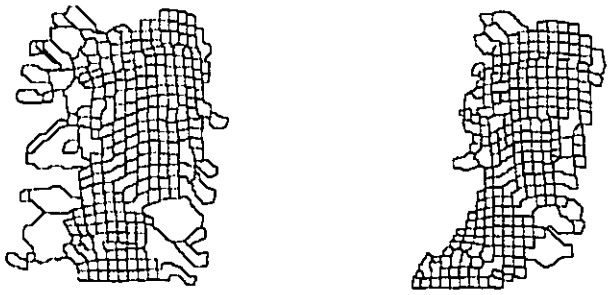
The left and right input images for the third test case are shown in Figure 6.16. For this case, the object was rotated and the cameras were put at a different height. The cameras are still on each side of the projector.



(a) Left PRBA (b) Right PRBA

Figure 6.17: The PRBA extracted for the left and right image of case III.

The two PRBAs extracted are shown in Figure 6.17. Very few matches (only 123) can be made from these two PRBAs. This can be explained by looking at the raw grids from which the PRBAs are generated. Figure 6.18 shows the raw grid extracted from the images.



(a) The grid of the left image (b) The grid of the right image

Figure 6.18: The raw grid extracted for the left and right image of case III.

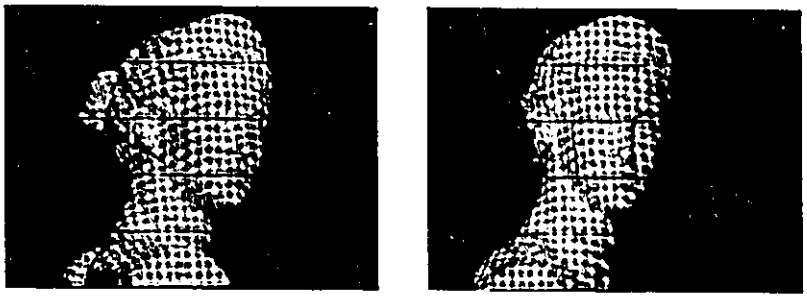
To eliminate potential errors, different steps are executed before a match to the PRBA is made. These steps eliminate connections and points according to the rules given in Section 3.4. In the

present case, they separate the grid into two regions along the chick of the statue. The PRBA is generated by looking at the point P who has the highest number of support (see Section 3.5). In the present images, this point is located over the face of the statue. This explains the small region in which the recognized points lie.

Total matches	Linear J	LMedS matches	Non-linear J	Average d per match
123	1.70597×10^6	100	132.22	0.661

Table 6.3: Results of case III for the estimation of F.

The few matches available affects the precision of the estimation process. The estimated F matrix yields an average distance d of 0.661. This seems a reasonable average distance. However, the effect of choosing nearby points when estimating F becomes apparent by looking at the epipolar images generated. The bucketing technique (see Section 4.2.3) is used to limit this effect but in the present case, its usefulness is limited by the concentration of the points over the same region (the face of the statue).



(a) The log of the left epipolar image.

(b) The log of the right epipolar image.

Figure 6.19: Epipolar images for case III.

The epipolar images are shown in Figure 6.19. The relatively high average distance d, combined with the limited region from which F is estimated, affects the validity of the epipolar constraint. From these epipolar images, it is apparent that the constraint is valid for the face of the statue. However, it is not valid near the shoulder of the statue.

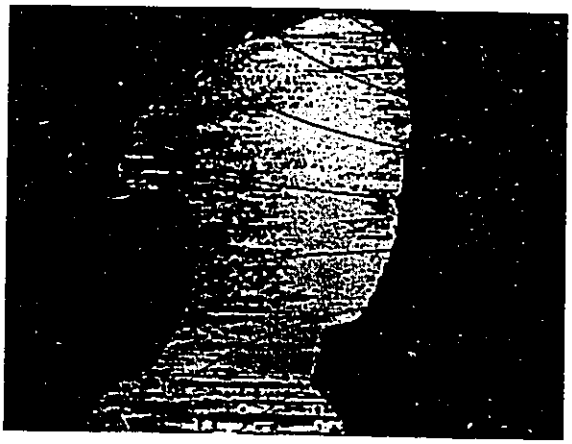
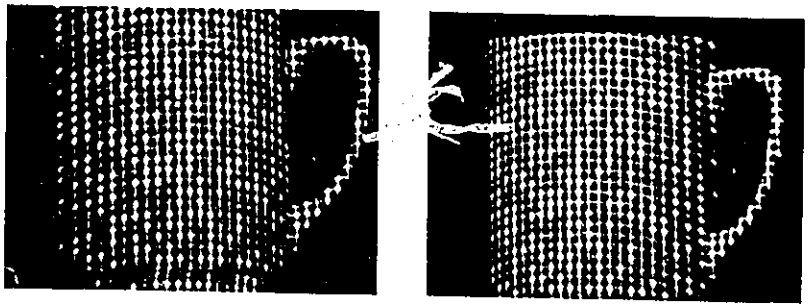


Figure 6.20: The disparity map for case III.

The disparity obtained is shown in Figure 6.20. The region for which the epipolar constraint is valid (the face) gives good results. But the region where it is not valid gives large occluded areas and gives bad disparity values.

6.4 Test Case IV

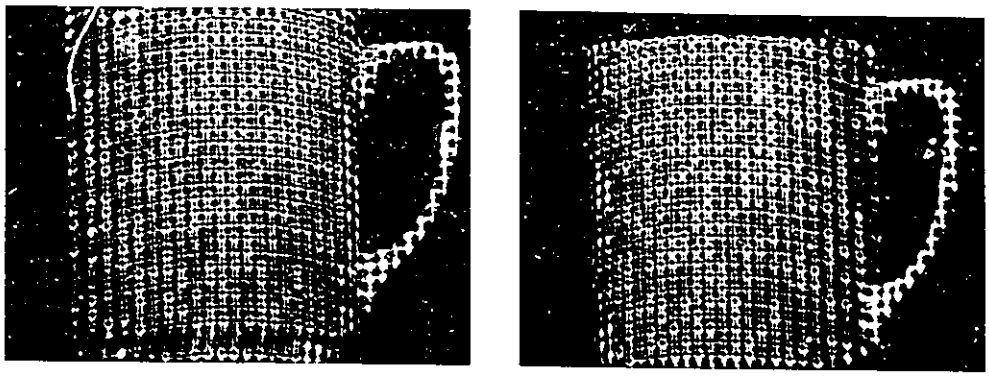


(a) Log of the left image. (b) Log of the right image.

Figure 6.21: Left and right test images for case IV.

The left and right input images for the fourth test case are shown in Figure 6.21. For this case, a cup replaces the original object. The location of the cameras is sensibly the same as in case III.

The cup object gives a high number of points for the PRBAs extracted (see Figure 6.22). At



(a) Left PRBA (b) Right PRBA

Figure 6.22: The PRBA extracted for the left and right image of case IV.

the base of the cup from the left image, an entire line of the grid was not perceived by the grid extraction process. Hence, almost all the lowest point in the PRBA are false matches. Some false matches also occurred near the handle of the cup in the left image. The right image doesn't contain any false match, however it has a very noticeable localization error on the top-right of the cup. This error is introduced at the grid extraction step, probably because of an over-smoothing of that region.

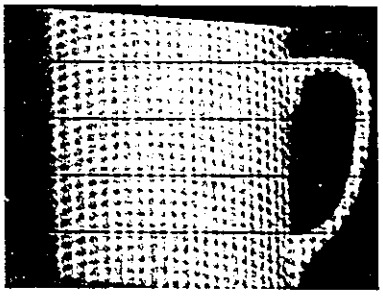
Total matches	Linear J	LMedS matches	Non-linear J	Average d per match
460	1.15265×10^7	417	161.42	0.1935

Table 6.4: Results of case IV for the estimation of F.

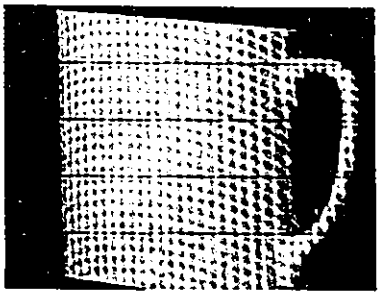
The high number of matches (460) allows for an estimation process which gives the lowest average distance d of all the test cases.

The epipolar images obtained are shown in Figure 6.23. For these images, the epipolar constraint is definitely valid.

Figure 6.24 shows the resulting disparity map. It does show one of the limitation of the current DP algorithm. The DP process is performed one epipolar line at a time. It does not take into account the results obtained for the other epipolar lines. This results in streaks (long



(a) The log of the left epipolar image.



(b) The log of the right epipolar image.

Figure 6.23: Epipolar images for case IV.

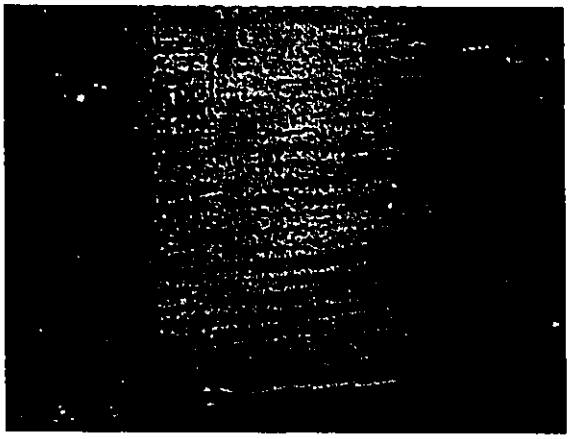


Figure 6.24: The disparity map for case IV.

horizontal lines) across the surface of the cup.

6.5 Conclusion

In this chapter, different results obtained for four test cases are described.

The first case shows the advantage of the DP used in this thesis over previous work [40]. It shows the benefits of AR modeling to improve upon the results obtained with DP.

The second case shows how the algorithm behavior in response to a small modification of the location of the cameras it also demonstrates the usefulness of the PRBA when a texture-less object is being matched.

The third case shows the effect of a drastic change in camera location and the effects of badly illuminated regions by the structured light (the chick of the statue).

The last case used the same drastic change in camera location as in the third case without having the bad illumination.

All the cases show that projecting the PRBA allows one to effectively estimate the epipolar geometry (the F matrix), and that mixing DP and AR modeling yields to good results.

Appendix A

The F matrix equation

A demonstration of how the F matrix equation is derived for Equation (4.8). From equation (4.7), the points m_1 and m_2 can be written as

$$m_1 = \begin{pmatrix} -f_1\, ku_1\, X + f_1\, ku_1\, \cot(\theta_1)\, Y + u_{01}\, Z \\ -f_1\, kv_1\, \sin(\theta_1)\, Y + v_{01}\, Z \\ Z \end{pmatrix}$$

$$m_2 = \begin{pmatrix} ((-ku_2\, r_{13} + ku_2\, \cot(\theta_2)\, r_{23})\, f_2 + u_{02}\, r_{33})\, Z + ((ku_2\, \cot(\theta_2)\, r_{21} - ku_2\, r_{11})\, f_2 + u_{02}\, r_{31})\, X \\ + ((ku_2\, \cot(\theta_2)\, r_{22} - ku_2\, r_{12})\, f_2 + u_{02}\, r_{32})\, Y + (-ku_2\, tx + ku_2\, \cot(\theta_2)\, ty)\, f_2 + u_{02}\, tz \\ (-f_2\, kv_2\, \sin(\theta_2)\, r_{23} + v_{02}\, r_{33})\, Z + (v_{02}\, r_{31} - f_2\, kv_2\, \sin(\theta_2)\, r_{21})\, X + (v_{02}\, r_{32} - f_2\, kv_2\, \sin(\theta_2)\, r_{22})\, Y \\ - f_2\, kv_2\, \sin(\theta_2)\, ty + v_{02}\, tz \\ r_{31}\, X + r_{32}\, Y + r_{33}\, Z + tz \end{pmatrix}$$

Therefore,

$$u_1 = \frac{-f_1\, ku_1\, X + f_1\, ku_1\, \cot(\theta_1)\, Y}{Z} + u_{01}$$

$$v_1 = \frac{-f_1\, kv_1\, \sin(\theta_1)\, Y}{Z} + v_{01}$$

and

$$u_2 = \frac{((-ku_2\, r_{13} + ku_2\, \cot(\theta_2)\, r_{23})\, f_2 + u_{02}\, r_{33})\, Z + ((ku_2\, \cot(\theta_2)\, r_{21} - ku_2\, r_{11})\, f_2 + u_{02}\, r_{31})\, X + ((ku_2\, \cot(\theta_2)\, r_{22} - ku_2\, r_{12})\, f_2 + u_{02}\, r_{32})\, Y + (-ku_2\, tx + ku_2\, \cot(\theta_2)\, ty)\, f_2 + u_{02}\, tz}{r_{31}\, X + r_{32}\, Y + r_{33}\, Z + tz}$$

$$v_2 = \frac{(-f_2\, kv_2\, \sin(\theta_2)\, r_{23} + v_{02}\, r_{33})\, Z + (v_{02}\, r_{31} - f_2\, kv_2\, \sin(\theta_2)\, r_{21})\, X + (v_{02}\, r_{32} - f_2\, kv_2\, \sin(\theta_2)\, r_{22})\, Y - f_2\, kv_2\, \sin(\theta_2)\, ty + v_{02}\, tz}{r_{31}\, X + r_{32}\, Y + r_{33}\, Z + tz}$$

The point m_2 can be written in function of point m_1, which gives

$$u_2 = \frac{\left(\left(\frac{f_2\, ku_2\, r_{11}}{f_1\, ku_1} - \frac{f_2\, ku_2\, \cot(\theta_2)\, r_{21}}{f_1\, ku_1} - \frac{u_{02}\, r_{31}}{f_1\, ku_1}\right) u_1 + \frac{f_2\, ku_2\, r_{11}\left(\frac{ku_1\, \cot(\theta_1)\,(v_1 - v_{01})}{kv_1\, \sin(\theta_1)} - u_{01}\right)}{f_1\, ku_1} + \frac{f_2\, ku_2\, r_{12}\,(v_1 - v_{01})}{f_1\, kv_1\, \sin(\theta_1)} - f_2\, ku_2\, r_{13} - \frac{f_2\, ku_2\, \cot(\theta_2)\, r_{21}\left(\frac{ku_1\, \cot(\theta_1)\,(v_1 - v_{01})}{kv_1\, \sin(\theta_1)} - u_{01}\right)}{f_1\, ku_1} - \frac{f_2\, ku_2\, \cot(\theta_2)\, r_{22}\,(v_1 - v_{01})}{f_1\, kv_1\, \sin(\theta_1)} + f_2\, ku_2\, \cot(\theta_2)\, r_{23} - \frac{u_{02}\, r_{31}\left(\frac{ku_1\, \cot(\theta_1)\,(v_1 - v_{01})}{kv_1\, \sin(\theta_1)} - u_{01}\right)}{f_1\, ku_1} - \frac{u_{02}\, r_{32}\,(v_1 - v_{01})}{f_1\, kv_1\, \sin(\theta_1)} + u_{02}\, r_{33}\right) Z + u_{02}\, tz - f_2\, ku_2\, tx + f_2\, ku_2\, \cot(\theta_2)\, ty}{\left(-\frac{r_{31}\, u_1}{f_1\, ku_1} - \frac{r_{31}\left(\frac{ku_1\, \cot(\theta_1)\,(v_1 - v_{01})}{kv_1\, \sin(\theta_1)} - u_{01}\right)}{f_1\, ku_1} - \frac{r_{32}\,(v_1 - v_{01})}{f_1\, kv_1\, \sin(\theta_1)} + r_{33}\right) Z + tz}$$

$$v_2 = \frac{\left(\left(\frac{f_2\, kv_2\, \sin(\theta_2)\, r_{21}}{f_1\, ku_1} - \frac{v_{02}\, r_{31}}{f_1\, ku_1}\right) u_1 + \frac{f_2\, kv_2\, \sin(\theta_2)\, r_{21}\left(\frac{ku_1\, \cot(\theta_1)\,(v_1 - v_{01})}{kv_1\, \sin(\theta_1)} - u_{01}\right)}{f_1\, ku_1} + \frac{f_2\, kv_2\, \sin(\theta_2)\, r_{22}\,(v_1 - v_{01})}{f_1\, kv_1\, \sin(\theta_1)} - f_2\, kv_2\, \sin(\theta_2)\, r_{23} - \frac{v_{02}\, r_{31}\left(\frac{ku_1\, \cot(\theta_1)\,(v_1 - v_{01})}{kv_1\, \sin(\theta_1)} - u_{01}\right)}{f_1\, ku_1} - \frac{v_{02}\, r_{32}\,(v_1 - v_{01})}{f_1\, kv_1\, \sin(\theta_1)} + v_{02}\, r_{33}\right) Z - f_2\, kv_2\, \sin(\theta_2)\, ty + v_{02}\, tz}{\left(-\frac{r_{31}\, u_1}{f_1\, ku_1} - \frac{r_{31}\left(\frac{ku_1\, \cot(\theta_1)\,(v_1 - v_{01})}{kv_1\, \sin(\theta_1)} - u_{01}\right)}{f_1\, ku_1} - \frac{r_{32}\,(v_1 - v_{01})}{f_1\, kv_1\, \sin(\theta_1)} + r_{33}\right) Z + tz}$$

Solving for Z in both equations,

$$\begin{aligned} Z = \frac{-kv_1\, \sin(\theta_1)\, f_1\, ku_1\, (-u_2\, tz + u_{02}\, tz - f_2\, ku_2\, tx + f_2\, ku_2\, \cot(\theta_2)\, ty)}{} \\ (u_2\, r_{31}\, kv_1\, \sin(\theta_1) + (ku_2\, r_{11}\, kv_1\, \sin(\theta_1) - ku_2\, \cot(\theta_2)\, r_{21}\, kv_1\, \sin(\theta_1))\, f_2 - u_{02}\, r_{31}\, kv_1\, \sin(\theta_1))\, u_1 \\ + ((r_{32}\, ku_1 + r_{31}\, ku_1\, \cot(\theta_1))\, u_2 + (ku_2\, r_{12}\, ku_1 - ku_2\, \cot(\theta_2)\, r_{22}\, ku_1 + ku_2\, r_{11}\, ku_1\, \cot(\theta_1) \\ - ku_2\, \cot(\theta_2)\, r_{21}\, ku_1\, \cot(\theta_1))\, f_2 - u_{02}\, r_{31}\, ku_1\, \cot(\theta_1) - u_{02}\, r_{32}\, ku_1)\, v_1 \\ + (-r_{31}\, ku_1\, \cot(\theta_1)\, v_{01} - r_{33}\, kv_1\, \sin(\theta_1)\, f_1\, ku_1 - r_{31}\, u_{01}\, kv_1\, \sin(\theta_1) - r_{32}\, ku_1\, v_{01})\, u_2 \\ + ((ku_2\, \cot(\theta_2)\, r_{23}\, kv_1\, \sin(\theta_1)\, ku_1 - ku_2\, r_{13}\, kv_1\, \sin(\theta_1)\, ku_1)\, f_2 + u_{02}\, r_{33}\, kv_1\, \sin(\theta_1)\, ku_1)\, f_1 \\ + (-ku_2\, r_{11}\, u_{01}\, kv_1\, \sin(\theta_1) - ku_2\, r_{11}\, ku_1\, \cot(\theta_1)\, v_{01} + ku_2\, \cot(\theta_2)\, r_{22}\, ku_1\, v_{01}) \\ + ku_2\, \cot(\theta_2)\, r_{21}\, u_{01}\, kv_1\, \sin(\theta_1) - ku_2\, r_{12}\, ku_1\, v_{01} + ku_2\, \cot(\theta_2)\, r_{21}\, ku_1\, \cot(\theta_1)\, v_{01})\, f_2 \\ + u_{02}\, r_{32}\, ku_1\, v_{01} + u_{02}\, r_{31}\, u_{01}\, kv_1\, \sin(\theta_1) + u_{02}\, r_{31}\, ku_1\, \cot(\theta_1)\, v_{01} \end{aligned}$$

$$
\begin{aligned}
Z \;=\; -\frac{v_2\, tz\, kv_1\, \sin(\theta_1)\, f_1\, ku_1 + (f_2\, kv_2\, \sin(\theta_2)\, ty\, kv_1\, \sin(\theta_1)\, ku_1 - v_{02}\, tz\, kv_1\, \sin(\theta_1)\, ku_1)\, f_1}{}\\
(-v_2\, r_{31}\, kv_1\, \sin(\theta_1) + v_{02}\, r_{31}\, kv_1\, \sin(\theta_1) - f_2\, kv_2\, \sin(\theta_2)\, r_{21}\, kv_1\, \sin(\theta_1))\, u_1\\
+ ((-r_{32}\, ku_1 - r_{31}\, ku_1\, \cot(\theta_1))\, v_2 + (-kv_2\, \sin(\theta_2)\, r_{22}\, ku_1\\
- kv_2\, \sin(\theta_2)\, r_{21}\, ku_1\, \cot(\theta_1))\, f_2 + v_{02}\, r_{32}\, ku_1 + v_{02}\, r_{31}\, ku_1\, \cot(\theta_1))\, v_1\\
+ (r_{31}\, u_{01}\, kv_1\, \sin(\theta_1) + r_{31}\, ku_1\, \cot(\theta_1)\, v_{01} + r_{32}\, ku_1\, v_{01} + r_{33}\, kv_1\, \sin(\theta_1)\, f_1\, ku_1)\, v_2\\
+ (f_2\, kv_2\, \sin(\theta_2)\, r_{23}\, kv_1\, \sin(\theta_1)\, ku_1 - v_{02}\, r_{33}\, kv_1\, \sin(\theta_1)\, ku_1)\, f_1\\
+ (kv_2\, \sin(\theta_2)\, r_{21}\, u_{01}\, kv_1\, \sin(\theta_1) + kv_2\, \sin(\theta_2)\, r_{22}\, ku_1\, v_{01} + kv_2\, \sin(\theta_2)\, r_{21}\, ku_1\, \cot(\theta_1)\, v_{01})\, f_2\\
- v_{02}\, r_{32}\, ku_1\, v_{01} - v_{02}\, r_{31}\, ku_1\, \cot(\theta_1)\, v_{01} - v_{02}\, r_{31}\, v_{01}\, kv_1\, \sin(\theta_1)
\end{aligned}
$$

And equating the right sides yields the following equation which corresponds to (4.8).

$$m_2^T A_2^{-T} T R A_1^{-1} m_1 = 0$$

where T is defined in (4.9).

Bibliography

[1] N. Ayache and B. Faverjon. Efficient registration of stereo images by matching graph descriptions of edge segments. *Int. J. Comput. Vision*, pages pages 107–131, 1987.

[2] H. H. Baker and T. O. Binford. Depth from edge and intensity based stereo. In *Proc. 7th Joint Conf. Artificial Intell.*, pages 631–636, Vancouver, Canada, Aug. 1981.

[3] S.T. Barnard and W.B. Thompson. Disparity analysis of images. *IEEE Trans. Pattern anal. Machine Intel.*, vol. 2(no. 4):pages 333–340, July 1980.

[4] Stephen T. Barnard and Martin A. Fischler. Computational stereo. *ACM Computing Surveys*, vol. 14(no. 4):pages 553–572, Dec. 1982.

[5] R. Bellman. *Dynamic Programming.* Princeton University Press, Princeton, N.J., 1957.

[6] K.L Boyer and A.C. Kak. Color-encoded structured light for rapid active ranging. *IEEE Trans. Pattern anal. Machine Intel.*, vol. 9(no. 1):pages 14–28, Jan 1987.

[7] Zen Chen, Shinn-Ying Ho, and Din-Chang Tseng. Polyhedral face reconstruction and modeling from a single image with structured light. *IEEE Trans. on Syst., Man and Cybern.*, vol. 23(no. 3):pages 864–872, May/June 1993.

[8] Steven D. Cochran and Gèrard Medioni. 3-d surface description from binocular stereo. *IEEE Trans. Pattern anal. Machine Intel.*, vol. 14(no. 10):pages 981–994, Oct. 1992.

[9] Umesh R. Dhond and J.K. Aggarwal. Structure from stereo — a review. *IEEE Trans. on Systems, Man and Cyber.*, vol. 19(no. 6):pages 1489–1510, Nov./Dec. 1989.

[10] D. B. Gennery. Object detection and measurement using stereo vision. In *Proc. ARPA Image Understanding Workshop*, pages 161–167, College Park, MD, Apr. 1980.

[11] Y. Grenier. Parametric time-frequency representation. In J.L. Lacoume and R. Stora Eds., editors, *Proc. Traitement du Signal.* Les Houches, Session XLV, 1985, Elsevier Science, 1987.

[12] M. J. Hannah. Bootstrap stereo. In *Proc. ARPA Image Understanding Workshop*, pages 201–208, College Park, MD, Apr. 1980.

[13] R.M. Haralick and L.G. Shapiro. *Computer and Robot Vision.* Addison Wesley Publishing Company, 1993.

[14] T.S. Huang and O.D. Faugeras. Some properties of the e matrix in two-view motion estimation. *IEEE Trans. Pattern anal. Machine Intel.*, vol. 11(no. 12):pages 1310–1312, Dec. 1989.

[15] R. A. Jarvis. A perspective on range finding techniques for computer vision. *IEEE Trans. Pattern anal. Machine Intel.*, vol. 5(no. 2):pages 122–139, March 1983.

[16] Steven M. Kay. *Modern Spectral Estimation: theory and application.* Signal processing series. Prenctice Hall, Engle /ood Cliffs, New Jersey 07632, 1989.

[17] Y.C. Kim and J.K. Aggarwal. Positioning 3-d objects using stereo images. *IEEE J. Robotics and Automation*, vol. 3(no. 4):pages 361–373, Aug. 1987.

[18] J. Kittler and J. Illingworth. On threshold selection using clustering criteria. *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC–15:pages 652–655, 1985.

[19] S. Kullback. *Information Theory and Statistics.* Wiley, New York, 1959.

[20] J. J. Le Moigne and A. M. Waxman. Structured light patterns for robot mobility. *IEEE J. Robotics Automat*, vol. 4(no. 5):pages 541–548, Oct 1988.

[21] H.C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, pages pages 133–135, 1981.

[22] Q.-T. Luong. Matrice fondamentale et calibration visuelle sur l'environnement: Vers une plus grande autonomie des systèmes robotiques. Orsay, Paris, France, December 1992. University of Paris XI. Dissertation.

[23] Q.T. Luong, R. Deriche, Olivier Faugeras, and T. Papadopoulo. On determining the fundamental matrix: Analysis of different methods and experimental results. Technical Report RR-1894, INRIA, April 1993.

[24] Q.T. Luong and O. Faugeras. Self-calibration of a stereo rig from unknown camera motions and point correspondences. Technical Report RR-2014, INRIA, july 1993.

[25] F.J. MacWilliams and N.J.A. Sloane. Pseudo-random sequences and arrays. *Proc. IEEE*, vol. 64(no. 12):pp 1715-1729, Dec. 1976.

[26] Henri Maître and Wei Luo. Using models to improve stereo reconstruction. *IEEE Trans. Pattern anal. Machine Intel.*, vol. 14(no. 2):pages 269-277, Feb. 1992.

[27] Henri Maître and Yifeng Wu. A dynamic programming algorithm for elastic registration of distorted pictures based on autoregressive model. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 37(2):pages 288-297, Feb 1989.

[28] D. Marr, G. Palm, and T. Poggio. Analysis of a cooperative stereo algorithm. *Biol. Cybern.*, vol. 28:pages 223-229, 1978.

[29] D. Marr and T. Poggio. Cooperative computation of stereo disparity. *Science*, vol. 194:pages 283-287, 1976.

[30] G. Medioni and R. Nevatia. Segment-based stereo matching. *Comput. Vision, Graphics, Image Processing*, vol. 31:pages 2-18, 1985.

[31] H. P. Moravec. Towards automatic visual obstacle avoidance. In *Proc. 5th Int. Joint Conf. Aritificial Intell.*, page 584, 1977.

[32] Yuichi Ohta and Takeo Kanade. Stereo by intra- and inter-scanline search. *IEEE Trans. Pattern anal. Machine Intel.*, vol. 7(no. 2):pages 139-154, March 1985.

[33] S.I. Olsen. Epipolar line estimation. pages 307-311, Santa Margherita Ligure, Italy, May 1992. Proc. Second European Conf. Comput. Vision.

[34] E.M. Petriu, N.Trif, S.K. Yeung, D. Ionescu, and P. Lavoie. Structured light triangulation using pseudo random binary array encoding. submitted to IEEE Transactions on Instrumentation and Measurements.

[35] B. Picinbono and M. Bouvet. Complex white noises and autoregressive signals. In *Proc. IEEE ICASSP 84*, pages 14.7.1–14.7.4, San Diego, March 1984.

[36] J.L. Posdemer and M.D. Altschuler. Surface measurement by space-encoded projected beam systems. *Comput. Graphics Image Processing*, 18:pages 1–17, 1982.

[37] W.H. Press, W.T. Vetterling, S.A. Teukolsky, and B.P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, second edition edition, 1992. pp 298.

[38] A. Rosenfeld, R.A. Hummel, and S.W. Zucker. Scene labeling by relaxation operation. *IEEE Trans. Syst. Man Cybern.*, vol 6:pages 420–423, June 1976.

[39] P.J. Rousseeuw and A.M. Leroy. *Robust Regression and Outliers Detection*. Joh n Wiley & Sons, New-York, 1976.

[40] S. Roy and J. Meunier. Stereoscopic analysis of multiple images. subnitted to the International Journal of Computer Vision, August 1993.

[41] N. Trif. Model-based visual recognition of 3-d objects using pseudo-random grid encoding. Master's thesis, University of Ottawa, May 1993.

[42] D.-C. Tseng and Z. Chen. Computing location and orientation of polyhedral surfaces using a laser-based vision system. *IEEE Trans. Robotics Automat.*, vol. 7:pages 842–848, Dec. 1991.

[43] P. Vuylsteke and A. Oosternlink. Range image acquisition with a single binary encoded light pattern. *IEEE Trans. Pattern anal. Machine Intel.*, vol. 12(no. 2):pages 148–164, Feb. 1990.

[44] E. J. Watson. *Mathematics of Computation*, vol. 16:pages 368–369, 1962.

[45] P.M. Will and K.S. Pennington. Grid coding: A preprocessing technique for robot and machine vision. In *Proc. 2nd Int. Joint Conf. Artificial Intell.*, pages 66–68, Sep. 1971.

[46] Z. Zhang, R. Deriche, Olivier Faugeras, and Q.T. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. Technical Report RR–2273, INRIA, May 1994.