

**SEGMENT-BASED STEREO MATCHING ALGORITHM WITH
RECTIFICATION FOR SINGLE-LENS BI-PRISM
STEREOVISION SYSTEM**

BAI YADING

NATIONAL UNIVERSITY OF SINGAPORE

2014

**SEGMENT-BASED STEREO MATCHING ALGORITHM WITH
RECTIFICATION FOR SINGLE-LENS BI-PRISM
STEREOVISION SYSTEM**

BAI YADING

(M.Sc., NATIONAL UNIVERSITY OF SINGAPORE)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

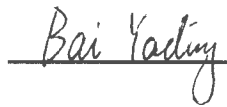
2014

DECLARATION

DECLARATION

I hereby declare that the thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.



Bai Yading

19 August 2014

ACKNOWLEDGMENTS

I would like to express the deepest appreciation to Associate Professor LIM KAH BIN, the supervisor of my Ph.D. study, for giving me such an interesting and fruitful project to improve and demonstrate my ability, and for his continuous supervision and valuable foresight and insight.

My gratitude also goes to Dr. Yong Xiao and Dr. Meijun Zhao, for their excellent early contribution on single-lens bi-prism stereovision system:

I would like to thank Mrs. Ooi, Ms. Tshin, Miss Hamidah and all the staff in Control and Mechatronics Laboratory of the Mechanical Engineering Department, for their kind support;

I consider it as an honor to work with WeiLoon Kee, Qing Wang, Jiayun Wu, Beibei Qian and other colleagues and friends in Control and Mechatronics Laboratory;

I owe my gratitude to my parents who give me great help and constant love through out all my student life.

TABLE OF CONTENTS

Contents

DECLARATION	I
ACKNOWLEDGMENTS	II
TABLE OF CONTENTS.....	III
SUMMARY	VI
LIST OF SYMBLES.....	VII
LIST OF TABLES	IX
LIST OF FIGURES	X
Chapter 1 Introduction	1
1.1 Stereovision	1
1.1.1 Stereo-correspondence.....	1
1.1.2 Rectification.....	4
1.1.3 Correspondence search algorithm	5
1.2 Motivation	5
1.3 Organization of the thesis.....	6
Chapter 2: Literature Review.....	8
2.1 Epipolar geometry	8
2.2 Stereo rectification	10
2.3 Stereo matching algorithm	13

TABLE OF CONTENTS

2.3.1 Global methods.....	14
2.3.2 Local methods.....	19
2.4 Image segmentation.....	21
2.4.1 Self-organizing map segmentation	22
2.4.2 Mean shift segmentation.....	24
2.4.3 Dense disparity feature	27
2.4.4 Image segmentation using level sets and active contour	27
2.5 Single-lens stereovision system	28
2.6 Summary	33
Chapter 3 Rectification of Single-lens Bi-prism Stereovision System.....	34
3.1 Background of stereovision rectification	36
3.1.1 Pinhole- camera model	36
3.1.2 Introduction of rectification using epipolar constraint	38
3.2 Ray-sketching approach to calculate the extrinsic parameters.....	41
3.2.1 Formation of virtual cameras.....	42
3.2.2 Determination of the extrinsic parameter using the ray-sketching method.....	44
3.3 Rectification algorithm.....	50
3.4 Experimental results	54
3.5 Summary	58
Chapter 4 Segment-based Stereo Matching Algorithm Using Belief Propagation.....	59

TABLE OF CONTENTS

4.1 Rectified image pair	61
4.2 Image segmentation.....	61
4.3 Disparity initialization using aggregation method	67
4.4 Disparity plane fitting.....	73
4.5 Refinement of the disparity plane	75
4.5.1 Refining disparity plane by outlier filtering	76
4.5.2 Refining disparity plane by merging connected segments with same disparity	80
4.6 Formulation of energy function.....	82
4.7 Belief propagation method	86
4.8 Depth recovery using disparity map.....	91
4.9 Summary	93
Chapter 5 Experiment Results and Analysis.....	94
5.1 Experiment setup.....	94
5.2 Experimental results and analysis	96
5.2.1 Experimental results based on the image pairs taken from Middlebury database.....	97
5.2.2 Experimental results using image pairs captured by single-lens bi-prism system	103
5.3 Summary	118
Chapter 6 Conclusion.....	120
List of Publications	126
Bibliography	127

SUMMARY

This thesis aims to develop a novel segment-based stereo-matching algorithm for 3-D depth recovery. The algorithm is to further improve the stereo correspondence results to achieve the said purpose. A novel segment-based stereo matching algorithm to extract the disparity information from the captured stereo image pair is proposed. A local method to obtain an initial disparity map is first employed and a segmentation algorithm (self-organizing map algorithm) is then applied to segment an image into regions of homogenous colors at the same time. Subsequently a plane fitting process is used to assign each segment a disparity plane. Finally, we create and optimize an energy function to refine the disparity values. To simplify the stereo correspondence search process, a rectification algorithm is developed. It involves the computation of the transformation matrix to transform the stereo image pair into the rectified stereo image pair. The algorithm developed is then tested on images captured by a single-lens bi-prism stereovision system developed by our research group. The results are compared with those determined by existing methods. To further demonstrate the effectiveness of our algorithm, additional rectified image pairs are used in our experimental study chosen from available standard database.

LIST OF SYMBLES

World coordinate system	(X_w, Y_w, Z_w)
Camera coordinate system	(X_c, Y_c, Z_c)
Disparity of the corresponding points located in left and right images	d
Depth of object in world coordinate system	z
Baseline, the distance between two camera optical centers:	λ
Effective real camera focal length	f
Effective virtual camera focal length	f'
Center of left image plane	$c_l(x_{ol}, y_{ol})$
Center of right image plane	$c_r(x_{ro}, y_{ro})$
Rotation matrix	R
Translation vector	T
Intrinsic parameters	M_{int}
Extrinsic parameters	M_{ext}
Fundamental matrix	F
Perspective projection matrix	P_p

LIST OF SYMBLES

Refractive index of the bi-prism glass	n
Epipole in the left image	e_l
Epipole in the right image	e_r
Matching cost of the stereo correspondence at point (x, y) with disparity d	$c(x, y, d)$
Point in world coordinate system	$P_w(X, Y, Z)$
Point in the left image	$p_l(x_l, y_l)$
Point in the right image	$p_r(x_r, y_r)$
Corner angle of the bi-prism	μ

LIST OF TABLES

Table 4.1 Performance of proposed initial disparity acquisition algorithm.....76

Table 5.1 Performance of different algorithms.....112

Table 5.2 Recovered depth value of the pixels chosen from “Robot Fighter” image.....125

Table 5.3 Parameters used in experiments of stereo image pair “robot fighter”125

Table 5.4 Performance of proposed algorithm with and without image rectification.....127

Table 5.5 Experimental results of stereo correspondence searching by different algorithms...130

LIST OF FIGURES

Figure 1.1 Searching of stereo correspondence and disparity.	2
Figure 1.2 Stereo image pair of the same scene captured by two cameras.....	3
Figure 1.3 Rectification of a stereo pair.....	4
Figure 2.1 Graph of epipolar geometry.....	8
Figure 2.2 Configuration of rectified image planes.....	11
Figure 2.3 Image structure after segmentation.	16
Figure 2.4 A randomly generated color palette.	24
Figure 2.5 Sketch map of mean shift	25
Figure 2.6 A single-lens stereovision system using a glass plate.	29
Figure 2.7 A single-lens stereovision system using three mirrors.....	30
Figure 2.8 A single-lens stereovision system using two mirrors.....	31
Figure 2.9 Single-lens stereovision system using prism.....	32
Figure 3.1 Single-lens Bi-prism stereovision system.	35
Figure 3.2 Pinhole camera model.	37
Figure 3.3 Epipolar geometry of two views.....	39
Figure 3.4 Image pair before and after rectification.	40
Figure 3.5 Formation of left and right virtual cameras.....	42
Figure 3.6 Relationship between left virtual camera and real camera.....	44
Figure 3.7 Sketch map of rectification algorithm.	51
Figure 3.8 “Book and card” image.	55
Figure 3.9 “Three objects” image a) left and right image; b) rectified left and right image	56
Figure 3.10 “Medicine” image a) left and right image; b) rectified left and right image.....	57

LIST OF FIGURES

Figure 4.1 Procedure of our segment-based stereo matching algorithm.	60
Figure 4.2 Process of the color palette updating.....	62
Figure 4.3 Segmentation results of Tsukuba:	65
Figure 4.4 Segmentation results of Art:	65
Figure 4.5 Segmentation result of Computer:.....	66
Figure 4.6 Aggregation windows.....	69
Figure 4.7 a) Reference image (Computer/Middlebury(2005)); b) initial disparity map.	71
Figure 4.8 a) Reference image (Arts/Middlebury(2005)); b) initial disparity map.	71
Figure 4.9 Flow chart of refinement of the disparity plane by Outlier filtering.	79
Figure 4.10 Structure of the segmented image.	82
Figure 4.11 Belief propagation Optimization.	87
Figure 4.12 Experimental results of Arts.....	90
Figure 5.1 Single-lens bi-prism stereovision system.	95
Figure 5.2 Experimental results of Tsukuba.....	97
Figure 5.3 Experimental results of Venus.....	98
Figure 5.4 Experimental results of Teddy.....	98
Figure 5.5 Experimental results of Cones.....	99
Figure 5.6 Experimental results of image “Books”.	102
Figure 5.7 Result of Image pair 1 captured by single-lens bi-prism system:.....	104
Figure 5.8 Result of Image pair 2 captured by single-lens bi-prism system:	105
Figure 5.9 Result of Image pair 3 captured by single-lens bi-prism system:	106
Figure 5.10 Result of Image pair 4 captured by single-lens bi-prism system:	107
Figure 5.11 Result of Image pair 5 captured by single-lens bi-prism system:	108

LIST OF FIGURES

Figure 5.12 Result of Image pair 6 captured by single-lens bi-prism system:	109
Figure 5.13 “Robot Fighter” image with 8 pixels chosen for the experiment.	112
Figure 5.14 Stereo image pair “Robot and Cup”.	114
Figure 6.1 Idea and non-ideal setups of single-lens stereovision system.	124
Figure 6.2 Schematic diagram of system setup using three single-lens stereovision system.	125

Chapter 1 Introduction

1.1 Stereovision

Stereovision is one of the most extensively researched areas in computer vision. It is important in 3-dimensional scene analysis, depth recovery, object recognition, etc. In stereovision, two or more images of the same scene are captured. Relevant information is then extracted and used to obtain the depth of the objects of interest in the scene. A complete depth map of the scene is obtained when the depths of all the pixels in the whole image are determined.

1.1.1 Stereo-correspondence

The basic problem in stereovision is searching the stereo correspondence which consists of determining the corresponding point of a point in one image (usually called the left image) in the other image (usually called the right image). Searching of the corresponding points in the two images of the same scene is important as they are essential in the determination of depth of objects in the scene. Figure 1.1 shows schematically a setup of a stereovision system.

In Figure 1.1, P is a point in the scene, the coordinates of which is $P(X, Y, Z)$ with respect to the pre-determined world coordinate system $O_w(X_w, Y_w, Z_w)$. The optical center of the left and right cameras are $O_L(X_L, Y_L, Z_L)$ and $O_R(X_R, Y_R, Z_R)$, respectively. λ , which is known as base-line distance, is the distance between O_L and O_R . It is bisected by O_w , note that the X-axes of O_w , O_L and O_R are aligned, and their Z-axes are all pointing in the same direction. Z_w is the depth that the stereovision system is trying to recover.

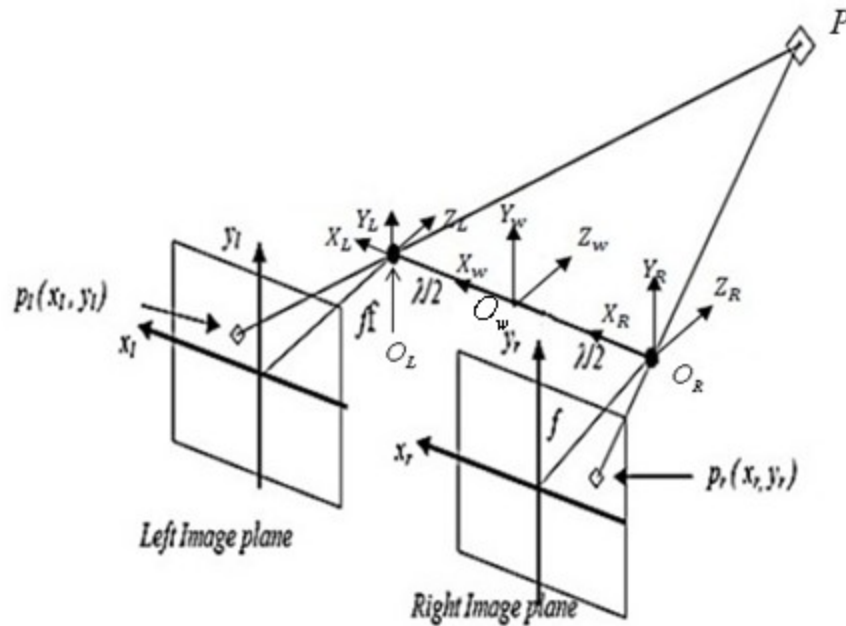


Figure 1.1 Searching of stereo correspondence and disparity.

The left (x_l, y_l) and right (x_r, y_r) image planes are the images of the scene captured by the left and right cameras, respectively. They are co-planar in Figure 1.1. $p_l(x_l, y_l)$ and $p_r(x_r, y_r)$ are the image points of P captured by the two said respective cameras. The two cameras are assumed to have the same focal length f .

With this setup, the depth of the point P is given by

$$Z = Z_w = \frac{f \lambda}{x_l - x_r} = \frac{f \lambda}{d} \quad (1.1)$$

where d is known as the disparity.

In Equation (1.1), f is the property of the camera, λ is the geometrical parameter. Thus, it is clear that the depth is highly dependent on the disparity d :

$$d = x_l - x_r \tag{1.2}$$

In the determination of d , x_l and x_r must be the x-coordinates of the same point in the scene, but might appear at different locations in the two images. In stereovision, $p_l(x_l, y_l)$ and $p_r(x_r, y_r)$ are known as correspondence points.

To illustrate this point, Figure 1.2 shows the two images of the same scene captured by two cameras. $P_l(x_l, y_l)$ and $P_r(x_r, y_r)$ are stereo correspondence points, whereas P_l' and P_r' are not.

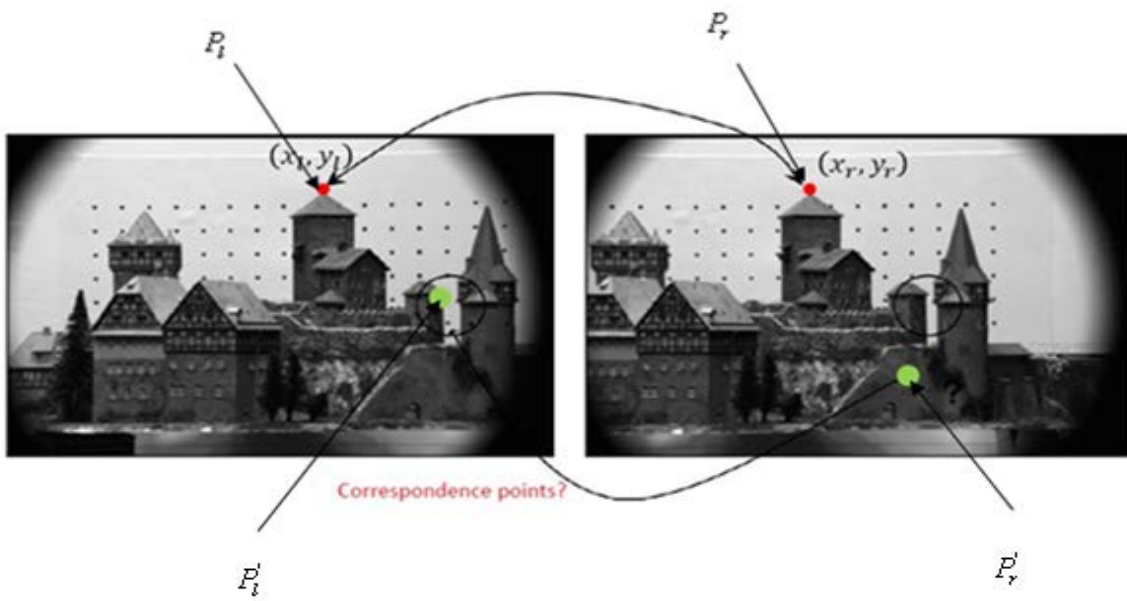


Figure 1.2 Stereo image pair of the same scene captured by two cameras.

1.1.2 Rectification

The two image planes in Figure 1.1 are coplanar (in Figure 1.3 the planes π'_1 and π'_2). It is a special and very convenient system setup in stereovision. In practice the two image planes in a stereovision system are usually not co-planar. They are usually at an angle to each other as shown in Figure 1.3 (planes π_1 and π_2). According to the epipolar geometry (which will be discussed in Section 2.1), if the two image planes are coplanar, the searching process of stereo correspondence points will be significantly simplified to a one dimensional search instead of the two dimensional search on the whole image.

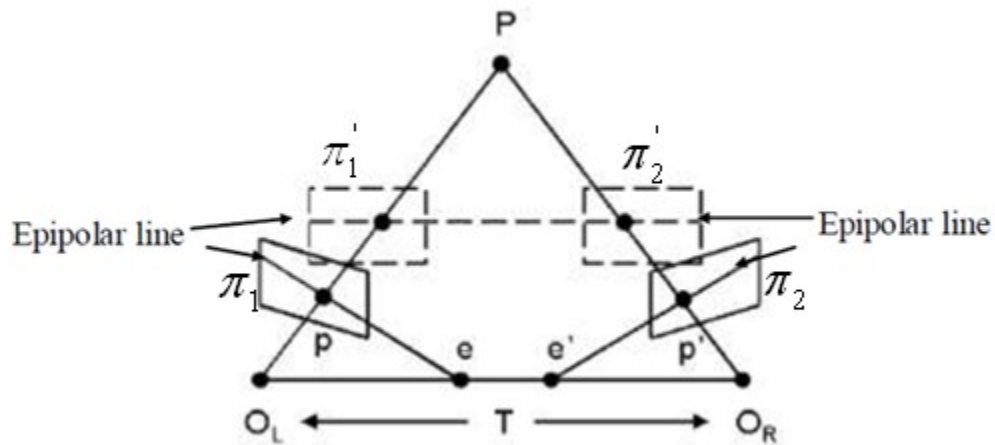


Figure 1.3 Rectification of a stereo pair.

1.1.3 Correspondence search algorithm

In this thesis, I propose a segment-based stereo matching algorithm using the belief propagation briefly described below.

In the proposed algorithm, a self-organization map segmentation method is employed to divide the reference image (chose one of the image from the image pair as the reference image) into segments. At the same time, it searches for stable points in the image using an initial disparity estimation method. A plane fitting process using the stable points is then applied to assign each single segment a disparity plane. A refinement of the disparity plane by filtering out outlier and merging connected segments with the same disparity plane is applied.

After the refinement, an energy function is created to evaluate the matching cost which will be optimized to aid in finding the best disparity map. A belief propagation method is used to complete the optimization process. The whole process shall be presented in Chapter 4.

1.2 Motivation

As mentioned above, searching the stereo correspondence points is an important issue and at the same time, a challenging one in stereovision. The accuracy of the results of this step affects, to a large extent, the result of 3-D depth recovery through the evaluation of disparity (Equation (1.2)). Admittedly there are many existing approaches in solving the problem of stereo correspondence search. In our research group, methods such as the calibration approach [65] and the geometrical approach [108] have been developed, to different degrees of success.

The main motivation of this thesis is to develop a novel approach to produce accurate results in stereo correspondence search. This helps to further expand the applicability of stereovision in areas that involve 3-D depth or scene recovery.

1.3 Organization of the thesis

In this thesis, new approaches and algorithms in stereovision are proposed to recover the depth of a scene in 3-D space. The thesis is organized into six chapters.

Chapter 1 introduces a general stereovision setup and discusses the main issues that affect the accuracy of 3-D depth recovery. The algorithm and approach that are employed are introduced.

A review on the theories and algorithms in stereovision is presented in Chapter 2. It includes epipolar geometry, epipolar constrain, rectification of stereo image pair, stereo matching algorithms, segmentation of color images and a single-lens stereovision systems.

The stereovision rectification algorithm proposed in this work is described in Chapter 3 which is based on a single-lens stereovision system. A ray-sketching approach is proposed to obtain the extrinsic parameters of the virtual cameras with respect to the real camera. The algorithm of computing the rectification transformation matrix is then proposed to rectify the stereo image pair captured using this system.

Chapter 4 presents a novel segment-based stereo matching algorithm using belief propagation algorithm. It consists of the following processes: color image segmentation, initial disparity map acquisition, plane fitting, disparity plane refinement and optimization of the energy function of disparity.

Chapter 5 gives the experimental results, which include the image segmentation results, final disparity map after applying the proposed algorithm. The discussion on the accuracy of the experiment result and comparison of the experimental results with other methods are also presented then in this chapter.

Last but not least, the conclusion and discussion on the future work are given in Chapter 6. A comprehensive list of reference is given after Chapter 6.

Chapter 2: Literature Review

This chapter introduces and reviews the relevant methods which are useful to handle the stereo correspondence search (stereo matching) problems.

2.1 Epipolar geometry

Epipolar geometry is an important concept in stereovision research. It is commonly exploited to facilitate the stereo correspondence search process. Epipolar geometry has been discussed by Trucco and Verri [1] and is briefly presented below.

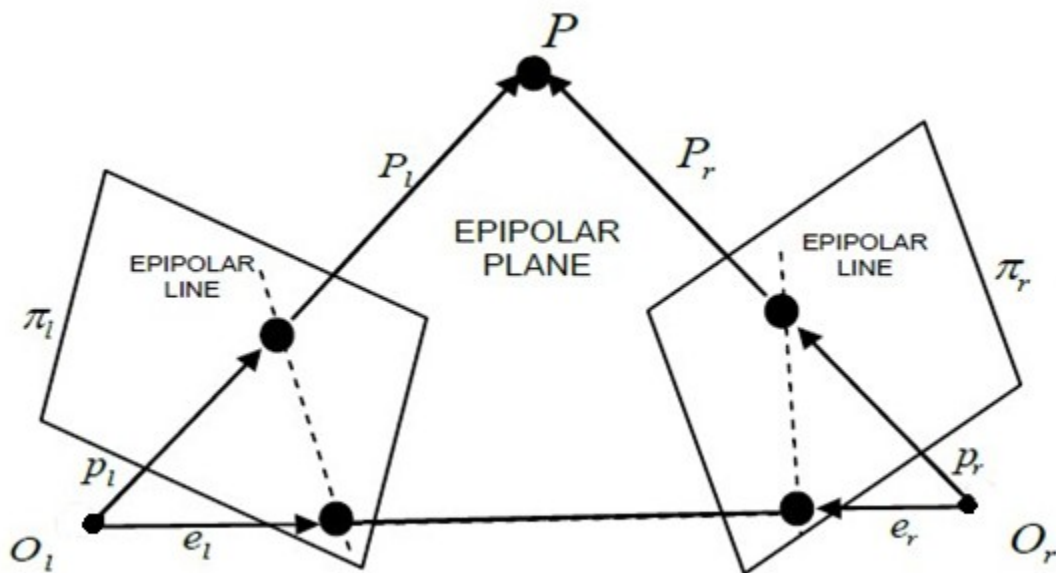


Figure 2.1 Graph of epipolar geometry.

In epipolar geometry, there are two pinhole cameras whose projection centers are O_l and O_r , respectively which are shown in Figure 2.1. The image planes π_l and π_r are their image planes respectively. The focal lengths are denoted by f_l and f_r . Normally, each camera identifies a 3-D reference frame fixed on its projection center and the z-axis is aligned with the optical axis. The vectors $P_l = [X_l, Y_l, Z_l]^T$ and $P_r = [X_r, Y_r, Z_r]^T$ refer to the same 3-D point P which is thought as a vector in the left and right camera reference frames respectively. The vectors $p_l = [x_l, y_l, z_l]^T$ and $p_r = [x_r, y_r, z_r]^T$ refer to the projections of P onto the left and right image plane respectively and they are expressed in the corresponding reference frame shown in Figure 2.1. The reference frames of the left and right camera are related by the extrinsic parameters:

- R , the rotation matrix;
- $T = O_r - O_l$, the translation vector.

The two parameters above enable us to define a rigid transformation in 3-D space. The relation between the vectors P_l and P_r is given by:

$$P_r = R(P_l - T) \quad (2.1)$$

The points at which the line through the center of projections (O_l and O_r) intersects the image planes in Figure 2.1 are called epipoles. They are denoted as e_l and e_r in Figure 2.1.

We can obtain the relation between a point in 3-D space and its projections is described by Equation (2.2) and (2.3) in vector form.

$$p_l = \frac{f_l}{Z_l} P_l \quad (2.2)$$

$$p_r = \frac{f_r}{Z_r} P_r \quad (2.3)$$

2.2 Stereo rectification

In the discussion on epipolar geometry, the two image planes are not coplanar. It is therefore very inconvenient to determine the epipolar line. In the stereo correspondence search, a special arrangement known as rectified configuration is used. In this configuration, the two image planes are co-planar as shown in Figure 2.2 (o_l and o_r are the optical centers of the left and right cameras, respectively). Given a point P in the real space, its projective point in left image plane is p_l , then the stereo correspondence point of p_l on the right image plane (p_r in Figure 2.2) must lie on the line which is a horizontal scan-line through p_l and extended to the right image (epipolar line).

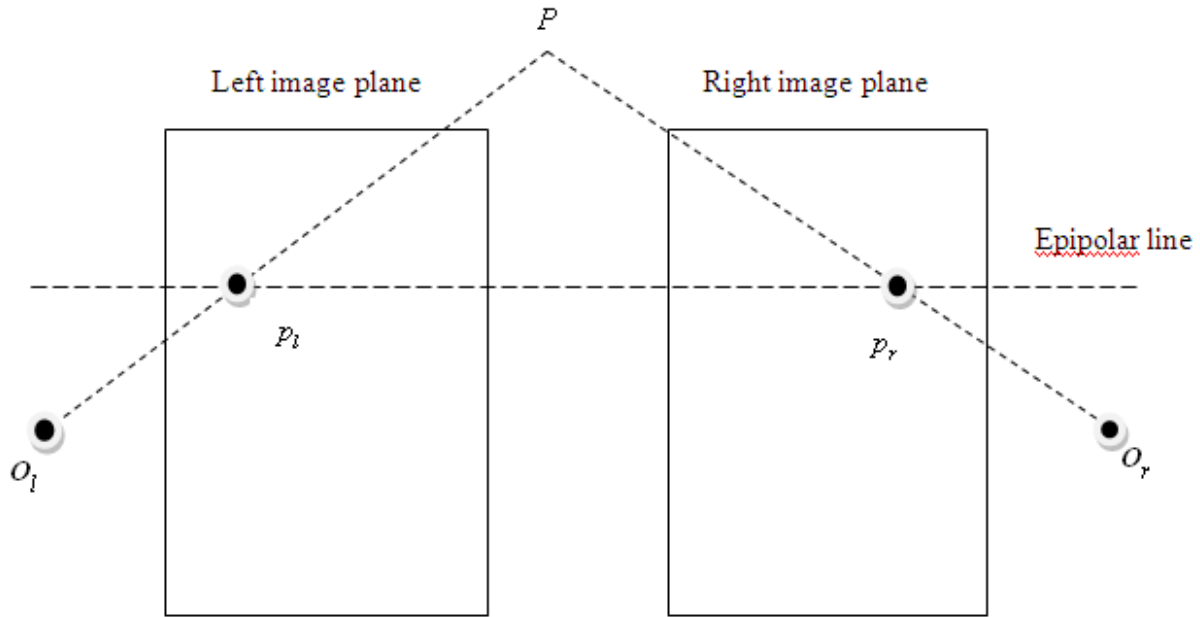


Figure 2.2 Configuration of rectified image planes.

Stereo rectification has been applied in photogrammetry for many years. The techniques originally were optical-based, but were later replaced by software methods that model the geometry of optical projection. In [2] an approach has been proposed by using the knowledge of known camera parameters. Similar techniques are demonstrated in [3]. The necessity of known calibration parameters is one of the disadvantages of these methods. Projective rectification has been introduced to overcome this disadvantage by using epipolar geometry with various constraints. In [4], a method to find the best transformation that preserves orthogonality around image centers has been given.

Recently, a stereo rectification method which takes geometric distortion into account and tries to minimize the effects of re-sampling has been given in [5]. Seitz et al. [6] propose a simple and efficient algorithm for generic two view stereo image rectification. Another available approach in [7] considered only the special case of partially aligned cameras. All these methods compute

projective rectifications in an indirect way, since an explicit estimation of fundamental matrix before rectification is needed. The computation of fundamental matrix has its own uncertainty. This indirect approach might obtain unpredictable rectifying results [8, 9]. Isgro and Trucco [10] propose a different procedure by obtaining the rectification transformations directly without computing the fundamental matrix with a disparity minimization based uniqueness criterion. However in some cases, the enforcement of minimizing x-axis disparity gives a distorted rectified image. A modification of this approach has been given in [11] using a proper shear transform. A method which approximates the calibrated case by enforcing the rectifying transformation to be collinear induced by plane at infinity has been presented by Fusiello et al. [12]. The advantage of their method is that there is no need of initial guess during the minimization process. A direct algorithm for stereo images rectification has been proposed by Zhang and Tsui [13] based on the estimation of homograph directly from geometric relationships.

In these algorithms, the basic rule of rectification is to transform the epipolar line into a horizontal scan-line. Given a point in one image, its corresponding point must lie on an epipolar line in the other image. This relationship is known as the epipolar constraint. If the two cameras are placed side by side on the same base line and have identical intrinsic parameters, then the acquired images are known as a rectified pair of stereo images. In these images, corresponding points must lie on a same horizontal scan line. When the two cameras are not arranged in this configuration, the image pairs can be ‘warped’ so that the corresponding points lie on the same scan-lines. This process is known as image rectification, and can be accomplished by applying 2-D projective transformations on each image. A stereo rectification method which takes geometric distortion into account and tries to minimize the effects of re-sampling has been proposed by D.

Lee and Kweon [14]. A simple and efficient algorithm for generic two view stereo image rectification has been presented in [15]. The approach of rectification which considers only the special case of partially aligned cameras is proposed by proposed by Agrawal et al. [16]. A different procedure which obtains the rectification transformations directly without computing the fundamental matrix with a disparity minimization based on the uniqueness criterion presented in [17].

In this thesis, a rectification algorithm using ray-sketching is proposed in Section 3.2.2 and we will discuss in detail about the differences and improvements of the algorithm.

2.3 Stereo matching algorithm

Stereo matching algorithms (also called stereo correspondence algorithms) continue to be an active research area as is proven by a large number of recent publications dedicated to this topic [18-23]. In stereovision system, two or more stereo images of the same scene are captured to extract the disparity information. To obtain the disparity from the given images, stereo correspondence search is an essential and important process. One point in the scene will project onto two image planes and the two projections lie in the image planes respectively are the stereo correspondence points. To determine the locations of stereo correspondence points which are necessary to compute disparity, many algorithms were proposed. In this section, we review some algorithms for stereo correspondence.

Normally, the algorithms are classified into two groups: global methods and local methods. Global methods can provide additional support for regions which are difficult to be matched locally but it is less sensitive to the occlusion region with uniform texture. Local methods are

very sensitive to the regions which are ambiguous. We review some global and local methods respectively in the following sections.

2.3.1 Global methods

A stereo matching method is called global method if there is a global objective function to be optimized. Normally, in global methods such as those presented in [24-28], some constraints are introduced to simplify disparity determination process [29]. Global methods first embody smoothness constraint and calculate the disparity map by minimizing an energy function. The energy function in global matching methods always consists of two terms, smoothness and data terms shown in Equation (2.4).

$$E_{all} = E_{data} + E_{smoothness} \quad (2.4)$$

E_{data} is used to calculate the matching cost when a certain disparity value d is set. $E_{smoothness}$ encodes the smoothness assumption made by the algorithms. In some algorithms [30, 31] a weight vector is added to obtain more accurate disparity map. Some of the global methods in the past years will be reviewed below:

1. Dynamic programming

Dynamic programming is an optimization approach that transforms a complex problem into a sequence of simpler problems; its essential characteristic is the multistage nature of the optimization procedure. More than the optimization techniques, dynamic programming provides a general framework for analyzing many problems discussed in [32]. Within this framework a variety of optimization techniques can be employed to solve particular aspects of a more general formulation. Usually, creativity is required before we can recognize that a

particular problem can be cast effectively as a dynamic program; and often sights are necessary to restructure the formulation so that it can be solved effectively. These approaches work by computing the minimum cost path through the matrix of all pair-wise matching costs between two corresponding scan-lines. Partial occlusion is handled explicitly by assigning a group of pixels in one image to a single pixel in the other image. Problems with dynamic programming include the selection of the right cost for occluded pixels [33-35] and the difficulty of enforcing scan-lines consistency. Another problem is that the dynamic programming approach requires additional ordering constraint. This constraint requires that the relative ordering of pixels on a scan-line remain the same between the two views in a stereo image pair, which may not be the case in scenes containing narrow foreground objects.

2. Cooperative Algorithms.

Cooperative algorithms such as Cooperative Optimization, inspired by computational models of human stereovision, were among the earliest methods proposed for disparity computation [36]. Such algorithms iteratively perform local computations, using nonlinear operations. In fact, for some of these algorithms, it is possible to explicitly state a global function that is being minimized.

In the cooperative optimization algorithm proposed by X. Huang [37], an image is decomposed into several segments and the segments are optimized individually by considering the influence of the neighboring segments. The energy function is optimized to obtain the best disparity plane for the final disparity map.

As shown in Figure 2.3, s_1, s_2, \dots, s_n are the segments. $E(x)$ denotes the total energy function of all segments, E_k is the energy function for the k^{th} segment, the cooperative optimization algorithm decomposes it into the sum of individual segments' energy function shown in Equation (2.5).

$$E(x) = \sum_{k=1}^n E_k \quad (2.5)$$

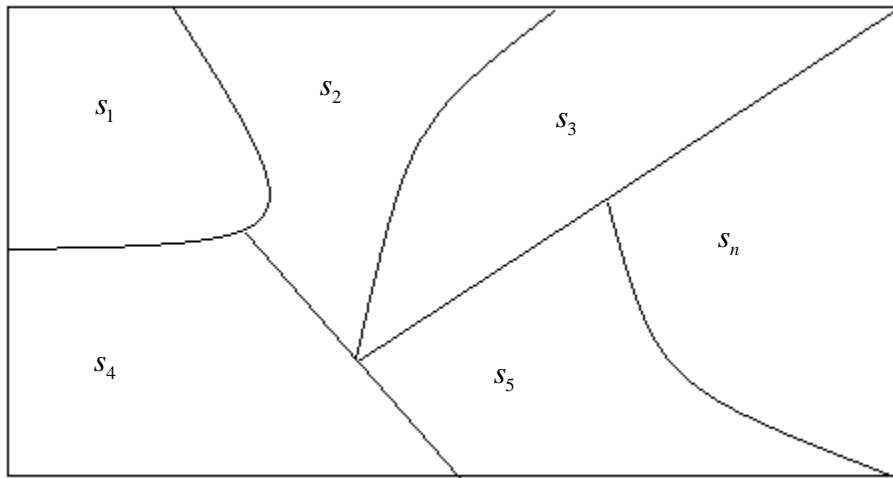


Figure 2.3 Image structure after segmentation.

The cooperative algorithm converts the disparity computation problem to a segment based optimization problem with multiple segments. However the result may not be correct due to the influence of the neighboring segments that are not yet considered.

To obtain the accurate depth map, an iterative process is executed in which the neighboring segments are considered. The energy function is modified into:

$$E_j = (1-k)E_j(x) + k_j \sum_{i \neq j} u_{ji} E_i(x) \quad (2.6)$$

where $E_i(x)$ is the energy function of i^{th} segment. k_j and u_{ji} are the corresponding weights.

Then the disparity value of j^{th} segment could be obtained using the method above.

3. Graph Cut algorithm

Kolmogorov and Zabih [38] and Hong and Chen [39] presented efficient graph cuts-based stereo algorithms to find a smooth disparity map that is consistent with the observed data. In their approaches, stereo correspondence problem is formulated as an energy minimization problem, which mainly includes: (i) a smoothness energy term that measures the disparity smoothness between neighboring pixel pairs; (ii) a data energy (E_d) that measures the disagreement between corresponding pixels based on the assumed disparities. A weighted graph is then constructed in which graph nodes represent image pixels; and graph label set (or terminals) relate the pixels to all possible disparities (or all discrete values in the disparity range interval) and graph edge weights correspond to the defined energy terms. Graph cuts technique is then used to approximate the optimal solution, which assigns the corresponding disparity (graph label) to each pixel (graph node).

4. Belief Propagation algorithm

Over the last few years, there have been exciting advances in the development of algorithms for solving early vision problems such as stereo and image restoration using Markov Random Field (MRF) models. While the MRF framework yields an optimization problem, good approximation techniques based on belief propagation [40] has been developed and

demonstrated for problems such as stereo and image restoration. The method is good both in the sense that local minima they find are minima over “large neighborhoods” and in the sense that they produce highly accurate results in practice. There are some differences between the belief propagation algorithm and graph cut algorithm. A comparison between the two different algorithms for the case of stereovision is described in [41].

The general framework in belief propagation for a problem is: Let P be the set of pixels in an image and L be a finite set of labels. The labels correspond to quantities that we want to estimate at each single pixel (for our matching problems, the label can be defined as the value of disparities or intensities). A labeling process f assigns a label $f_p \in L$ to each pixel $p \in P$. We assume that the labels should vary slowly almost everywhere but may change rapidly at some place such as pixels along the object boundaries. The quality of a labeling is given by an energy function shown below:

$$E(f) = \sum_{p \in P} C_p(f_p) + \sum_{p, q \in \text{edge}} A(f_p, f_q) \quad (2.7)$$

In Equation (2.7), p and q are the pixels along the edge. $C_p(f_p)$ is the cost of assigning label to pixel, and it is commonly called data cost and $A(f_p, f_q)$ measures the cost of assigning labels f_p and f_q to two neighboring pixels and it is normally called discontinuity cost. What we need is to find a label that minimizes this energy cost function. This framework when applied to our work can also be computational intensive.

2.3.2 Local methods

Local methods compute each pixel's disparity independently over a support region. The matching costs are aggregated over the region, and the disparity level with the minimal cost is selected as the output of the pixel. Some recent techniques such as adaptive weight [42] and segment support [43] can produce accurate disparity maps but with considerable computational time.

a. Non parametric local transform method

In the local method proposed by Zabih and Woodfill [42], two non-parametric local transforms are involved in the stereo correspondence search algorithm. The first, called rank transform, is a non-parameteric measure of local intensity. The second, called the census transform, is a non-parametric summary of local spatial structure.

Let P be a pixel, $I(P)$ is its intensity (usually an 8-bit integer,) and $N(P)$ denotes the set of pixels surrounding P . Define $\varepsilon(P, P')$ as 1 if $I(P') < I(P)$ and 0 otherwise. The non-parametric local transforms depend solely on the set of pixel comparisons, which is the set of ordered pairs $E(P) = \cup(P', \varepsilon(P, P'))$. The transform is called the rank transform, and is defined as the number of pixels in the local region (neighboring pixels of P) whose intensity is less than the intensity of the center pixel. Therefore, expressing it mathematically, rank transform is shown as $R(P) = \|\{P' \in N(P) | I(P') < I(P)\}\|$.

If $R(P_l)$ denotes the rank transform of a point P_l in the left image and $R(P_r)$ denotes the rank transform of a point P_r in the right image, the point in the right image which minimized the value of $R(P_r) - R(P_l)$ will be determined as the stereo correspondence point of P_l .

b. Feature matching method

The feature matching method proposed in [43] is a local stereo matching method which is based on the feature information in segment domain. For a stereo image pair, the pixels with the same feature information will be matched and could be seen as a stereo correspondence point. The method is insensitive to depth discontinuities and to regions of uniform texture. Normally the feature matching method will have two stages. 1) search in a sample texture for neighborhoods most similar to a context region; 2) merge a patch or a pixel with the synthesized output texture. Dynamic programming and graph cut have been used to optimize patch merging stage. A neighbor search in texture synthesis remarkably resembles image registration in computer vision. Rigid template matching is the simplest among all the feature matching methods due to the accurate capturing of the feature structure not only in the region with texture, but also in regions with depth discontinuities or occlusion.

2.4 Image segmentation

Segmentation is a process of dividing an image into objects or elements that are coherent under some criteria [44]. Segmentation techniques are often based on two image features: discontinuity and similarity. In color images, disparity discontinuities are found in image regions where there are abrupt color changes. Similarities are found in image regions where there is little color changes.

Most of the conventional methods of stereo matching deal directly with the pixels in the image one by one. The process involved is fastidious and computational intensive. Many of the methods begin by assigning disparity values using various methods of estimation directly to each pixel, and then optimization techniques are applied to obtain the best value for each pixel.

As mentioned in the Introduction, segment-based methods have attracted a great deal of attention due to their good performance. They are based on the assumption that the scene structure can be approximated by a set of non-overlapping planes in the disparity space and that each plane is coincident with at least one homogeneous color segment in the reference image. Segment-based stereo matching can simplify the problem by assigning the disparity for each single segment instead of doing so to each single pixel. In the former case, the problem of the determination of disparity will be reduced to dealing with segments (groups of pixels) rather than all the pixels in an image.

The segment-based methods have three important characteristics. Firstly, it reduces the ambiguity associated with un-textured regions. The drawback of the assumption is that depth discontinuities are prone to occur at different color boundaries. Secondly, the computational complexity is reduced due to much larger segments combining numerous areas into large block. Finally, the methods have the benefit of being noise tolerant by aggregating over those likelihood color pixels. The next sub-sections will present several methods of segmentation which are used for different purposes.

2.4.1 Self-organizing map segmentation

The self-organizing map is a color segmentation method which can successfully help to obtain a segmented image using the theory of Neural Network. The self-organizing map method [45] proposed a two- stage strategy including:

1. A fixed-size two dimensional feature map to capture the dominant colors of an image in an unsupervised way;
2. Combining a variable-sized one-dimensional feature map and color merging to control the number of color segments that are used in segmentation process.

Color segmentation is successfully performed by self-organizing maps (SOMs) and related networks in [46-48]. The model in [49] is based on a two-step Neural Network. In the first step, a SOM performs color reduction and then a simulated annealing step searches for the optimal clusters from SOM prototypes. This task involves a procedure of hierarchical prototype learning to generate different sizes of color prototypes from the sample of object colors. A color segmentation algorithm should be adaptive with respect to the final number of selected

colors/objects because non-adaptive color algorithms often produce poor color segmentation results. It is characterized by a variable number of nodes, which grows as new prototypes are needed. The new nodes are connected to two others under a triangle-shaped neighborhood. Instead of using the topological map to implement lateral plasticity control as a SOM does, the topological relations between nodes work as an inhibitory function of adaptive learning upon the prototype vectors.

The color quantization process reported in [50-53] can be characterized by the following two steps: (i) autonomous selection of colors from all colors present in the original image to form the color palette (Figure 2.4) and (ii) take each color in the original image to the nearest color in the palette. The final image must have only the selected colors and should be as similar as possible to the original one. In fact, color quantization and color segmentation are based on the same process of reducing the number of image colors. The main difference is that color quantization usually results in a pre-defined final number of colors and the larger the number of final colors, the better is the resultant image.

However, in color segmentation, each final color in the resultant image represents an object. Therefore only a few colors are desired, otherwise, many objects or subparts of objects will be detected. Hence, in color segmentation only a few distinct and dominant colors are desired.

This method is good for the reason that the user could decide the number of segments before operating the algorithm which could be used in the analysis of the effect of the disparity map obtained by different number of segments as well.

A self-organizing map segmentation algorithm is applied in this thesis to decompose reference image pair into segments with homogenous colors which will be discussed in Section 4.2.



Figure 2.4 A randomly generated color palette.

2.4.2 Mean shift segmentation

For common mean shift segmentation, the main task is to determine in which segment a pixel should be clustered into. Figure 2.5 shows the basic structure of the mean shift segmentation. If there are n sample points $(x_i, i = 1 \text{ to } n)$, the mean shift vector of one randomly point x is defined as:

$$M_h(x) = 1/n \sum_{x_i \in S_h} (x_i - x) \quad (2.8)$$

where S_h is the radius of a spherical space.

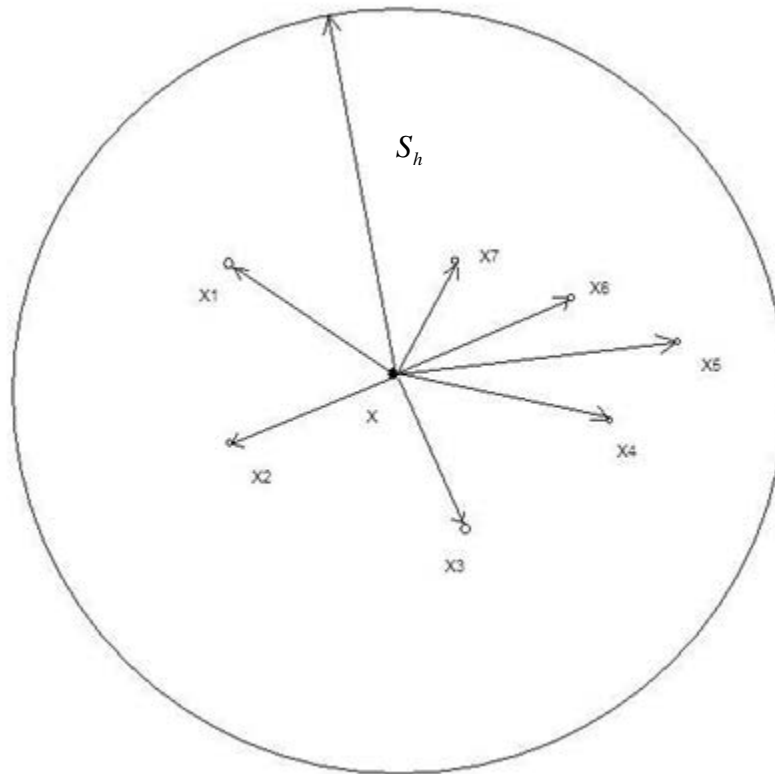


Figure 2.5 Sketch map of mean shift

We can see from Figure 2.5 that $(x_i - x)$ is the shift vector from x to x_i and the mean shift vector $M_h(x)$ is the average of all shift vectors in S_h ; if x_i is represented by a probability density function $f(x)$, then the mean shift vector always points toward the direction of maximum increase in the density.

The mean shift segmentation algorithm applied in stereo matching proposed in [54-58] is an iterative process, and hence the criterion of convergence must first be defined. In this process, when the criterion has been met, it simply means that a target point x_t can be clustered into the segment in question.

To determine which segment the target point x_t in the reference image should be clustered, we need firstly to set the kernel bandwidth H_s in the spatial domain and H_r in the range domain respectively to represent the segments we will obtain. Secondly we need to calculate the convergence point of x (which is the pixel we want to determine and to which segment it belongs). According to the mean shift segment theory, the mean shift vector is iteratively calculated until the vector is the same as the result computed at the last iteration. Let us denote $y_{t,i}$ as the mean shift vector at i^{th} iteration for the target point x_t , and $x_{t,i}$ is the point corresponding to $y_{t,i}$ at this iteration.

The detail process of the segmentation is as follow:

Algorithm 2.1: Algorithm for mean shift segmentation

1. Initial H_s and H_r
2. Select target pixel x_t
3. Initial $i=1$ and let $y_{t,1} = x_t$
Calculate $y_{t,i}$ until $y_{t,i} = y_{t,i-1}$ and record the information of $y_{t,i}$ and x_t
4. Choose next target pixel and repeat the steps 2 and 3.
5. Delineate the clusters $\{C_p\}_{p=1\dots m}$ by grouping together all $y_{t,i}$ which are closer than H_s in the spatial domain and closer than H_r in the range domain
6. For each x_t assign segment label $L=\{p|y_{t,i} \in C_p\}$

2.4.3 Dense disparity feature

In the method proposed in [59, 60], a dense disparity feature (DDF) is defined as a connected set of pixels in the reference image to which a common disparity value can be attributed. The main idea of the algorithm is to divide the original image into certain number of dense disparity features. The entire pixels in the same disparity feature share the same disparity value.

The target image is translated with respect to the reference image by a displacement corresponding to a disparity value and is subtracted from the reference image, producing a difference image. Match features correspond to pixel regions in the difference images that satisfy a matching criterion. Then a series of binary morphological operations are used to connect pixels and form smoothed dense disparity features which correspond to uniform disparities.

This is a new approach to facilitate stereo correspondence searching discussed in [61, 62]. The method extracts dense disparity features by shifting the stereo images relative to each other and applying a series of binary operations on the resulted difference image. The algorithm can run either for selected disparity values or for all possible disparity levels within a predefined range.

2.4.4 Image segmentation using level sets and active contour

Level sets method is firstly proposed by Osher and Sethian in [115] for evolving curves and surface. It has been employed in many applications, because it allows for automatic change of topology, such as merging and breaking; moreover the calculations are made on a fixed rectangular grid. In [116], Caselles et al. have proposed the geodesic active contour model for image segmentation which is an elaborated method for boundary-based image segmentation application.

In the segmentation method, four steps of operation are taken.

1. An off-line step is performed that creates multicomponent probabilistic texture descriptors for the given set of texture patterns, where the multi-dimensional feature data is derived using a set of filtering operators. There are three sub-steps in this step: a) The texture features are captured using a predefined set of filter operators; b) these features are modeled for each texture pattern using continuous probability density function; c) Validating these features and refers to the estimation of some reliability measurements for the different filters operators that have been used to obtain them
2. Apply the same operators and derive an observation set that refers to the same multidimensional feature space to construct the texture descriptor
3. Segmentation step is performed using a unified model that integrates a boundary and a region based module and refers to the optimization of a curve based objective function.
4. Use the geodesic active region model that aims to find the best minimal length geodesic curve that consists of image pixels with high boundary probabilities and creates regions that refer to an “optimal” grouping which is seen as a segment.

This kind of segmentation is a feature-based segmentation method that is also called geodesic active region. The boundary and region information are cooperating in a coupled active contour model and a level set approach is used to implement the image segmentation.

2.5 Single-lens stereovision system

Many different single-lens stereovision techniques have been developed because of the significant potential advantages of this technique over conventional two or multiple camera stereovision systems. These techniques can be classified into two groups. The techniques of the

first group use some optical devices, such as mirrors to achieve the stereovision effect; the techniques of the second group exploit some known cues from the vision system itself such as known camera movement or from its environment such as known illumination conditions. Both categories rely on triangulation knowledge to explain the generation of their stereovision effect.

The system created and presented in [80] is a single-lens stereovision system which can obtain stereo images as shown in Figure 2.6. In this system a glass plate is positioned in front of the camera so that its rotation will cause the optical axes of the camera slightly shifted because of its refraction. Hence stereo image pairs can be captured but with small disparities.

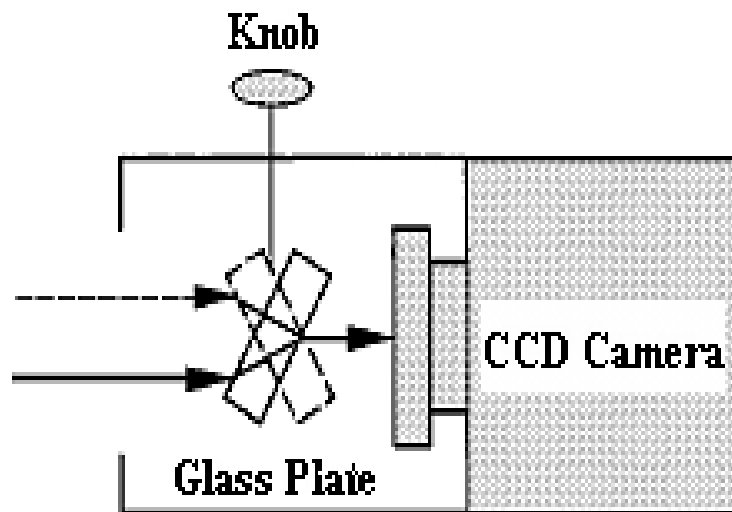


Figure 2.6 A single-lens stereovision system using a glass plate.

Another system describe in [81] is a single-lens stereo-camera system which employs three mirrors as shown in Figure 2.7. Two mirrors are positioned at a 45° relative to the optical axis of the camera, and a third mirror is positioned in front of the camera lens and can be rotated to be

parallel to either of the fixed mirrors in sequence and two different images are thus obtained from one static scene.

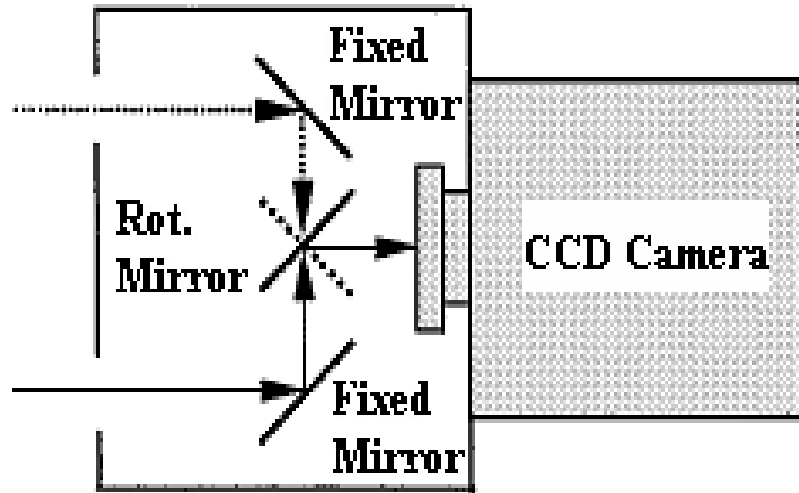


Figure 2.7 A single-lens stereovision system using three mirrors.

The systems described above require the camera to take two separate shots to obtain stereo image pair. Their applications are probably limited to static scene or slow changing environment only. In [82], the system is described as another mirror-based single-lens stereovision system as shown in Figure 2.8 which can overcome this problem. The acquired images are reflected by the mirrors and transformation processes of these images are needed before carrying out the correspondence and depth measurement as in a normal two camera stereovision system.

Another system which has movable mirror components to control its view scope is introduced by Inaba et al. [83]. Nene and Nayar [84] performed further analysis of this kind of mirror based stereovision system. In their study a single camera is positioned to point towards not only planar mirrors, but also hyperboloidal, ellipsoidal and paraboloidal mirrors. By using non-planar

reflecting surfaces, such as hyperboloids and paraboloids, a wide field of view (FOV) can be achieved.

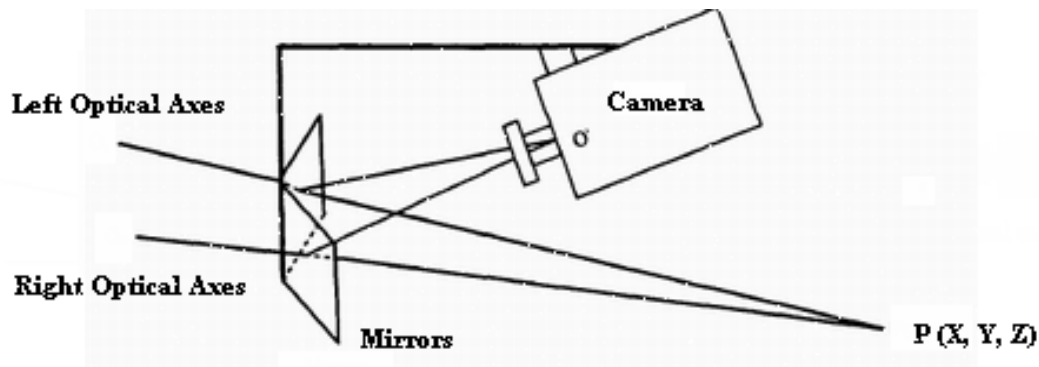


Figure 2.8 A single-lens stereovision system using two mirrors.

In [85], the author proposed a system such that an implementation by positioning the mirror beside the object. According to the theory proposed by Ng [86], he further refined the concepts of stereovision from a single perspective of a mirror symmetric scene and concluded that a mirror symmetric scene is equivalent to observing scene with two cameras and all traditional analysis tools of binocular stereovision can be applied. Later, Lim and Xiao [64-68] proposed a similar system (See Figure 2.9) and extended the study to include the use of multi-face prism. They also proposed the idea of virtual cameras for the analysis of their system.

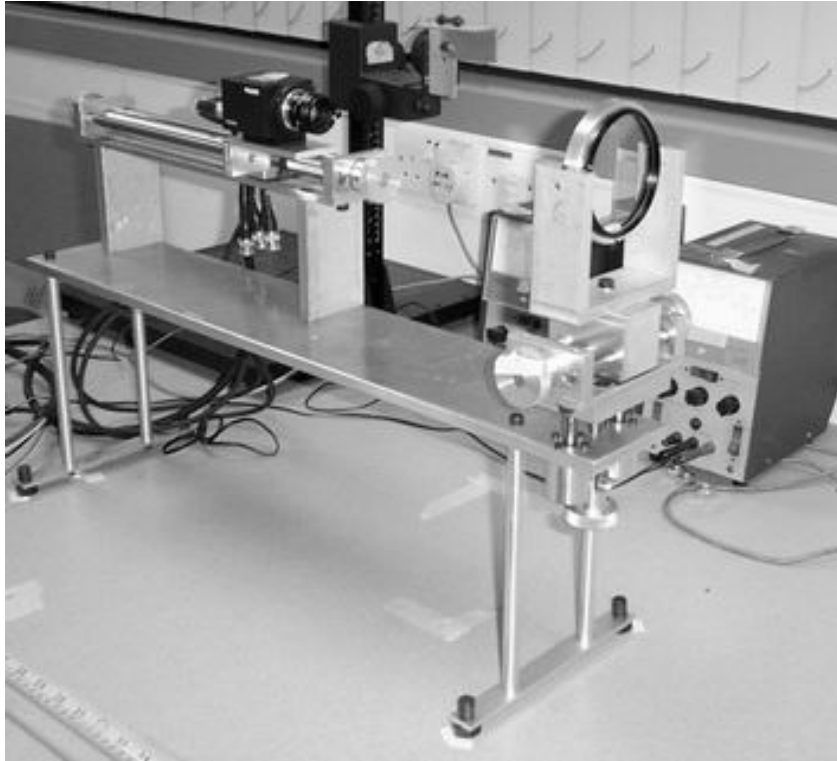


Figure 2.9 Single-lens stereovision system using prism.

In this thesis, a similar Single-lens Bi-prism stereovision system (discussed in detail in Section 5.1) will be used to capture stereo image pairs and the image pairs are used to test the accuracy of our proposed algorithms. A significant advantage of this prism based virtual stereovision system is that two stereo images will be captured simultaneously with one camera. The number of the system parameters is less than that of conventional system. In addition, the synchronization problem in image capturing is eliminated automatically. The parameters obtained using the ray-sketching method (discussed in Section 3.2.2) can also be used to handle the rectification problem (discussed in Chapter 3) which significantly simplifies the stereo matching algorithm proposed in this thesis.

2.6 Summary

In this chapter, we have first reviewed the theory of epipolar geometry and epipolar constraint which are very important both in rectification and stereo correspondence searching. Secondly a few of stereo matching algorithm using global and local methods were reviewed. Some of image segmentation algorithms were then reviewed. The development and the effects of the rectification technique were discussed after which the single-lens stereovision systems were described at the end of this chapter. Based on the information gained from the review, we proposed a segmented based stereo matching algorithm (Chapter 4) with rectification (Chapter 3) for a single-lens bi-prism stereovision system. In the next chapter, we will describe a novel rectification algorithm for the system.

Chapter 3 Rectification of Single-lens Bi-prism Stereovision System

In stereovision, setting up a system to capture two or more different views of the same scene is the basis of all the subsequent processes. Conventional systems use two or more cameras to achieve this goal which have inherent problems such as taking images at different times may lead to the different illumination conditions and the number of system parameters is large which increases the difficulty of the rectification, and etc.

In this thesis, we will use a single-lens bi-prism stereovision system which only employs one camera. This system is able to capture two different views of the same scene to form one stereo image pair, which can be taken as two images captured by two identical and symmetrically positioned virtual cameras. There are some advantages using this system, one of which is the smaller number of system parameters compared to a binocular stereovision system. In addition, there is no necessity to have special hardware to synchronize the image capturing. Another advantage is that as there is only one camera used in this system, the system setup is simple and is also more compact.

The image captured in this system is divided into two “left” and “right” sub-images. The two images can be considered to have been captured by two virtual cameras (shown in Figure 3.5) consequently. However, due to the refraction of the prism, the two virtual cameras are slanted. The system setup is shown in Figure 3.1 and distances between system components vary according to the requirements of the experiment.

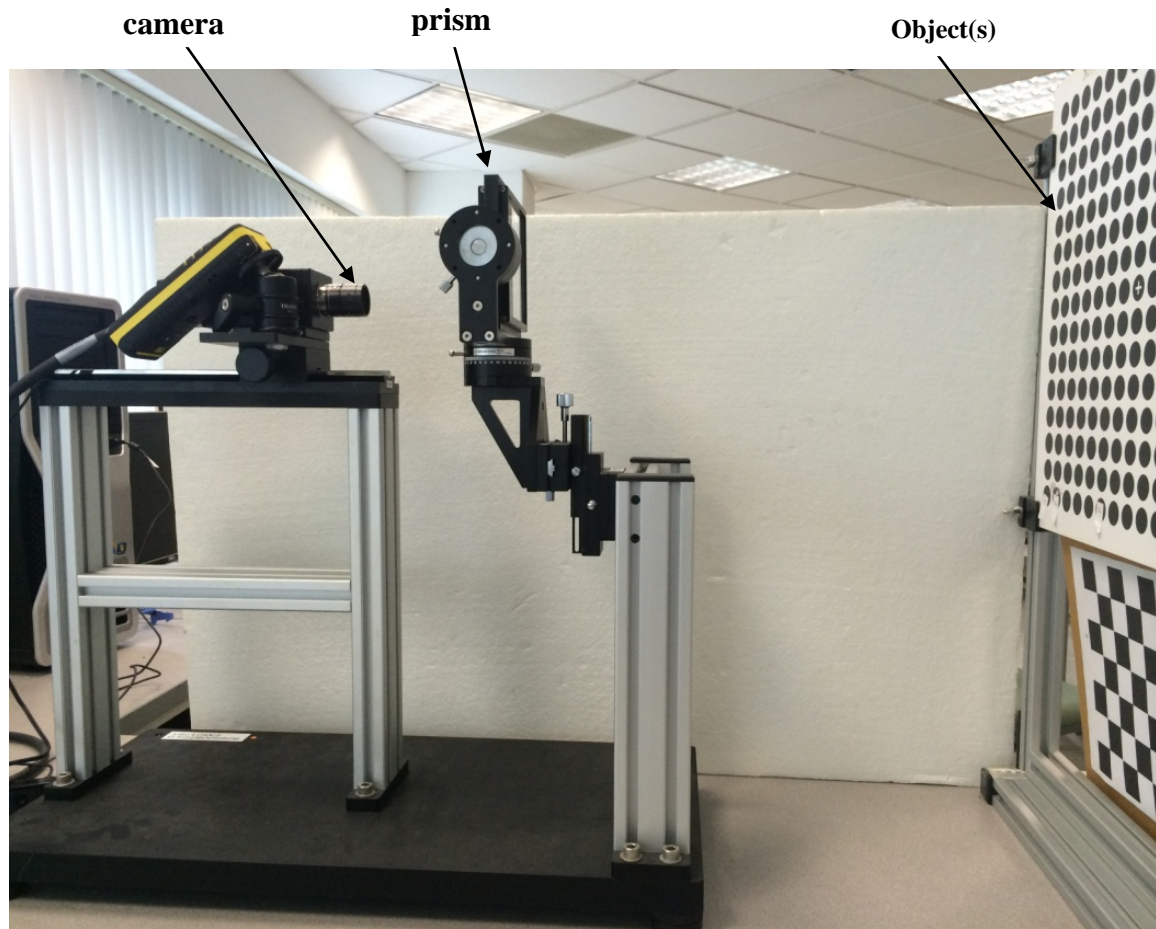


Figure 3.1 Single-lens Bi-prism stereovision system.

In stereovision, if the system setup can be arranged such that the two captured image planes are co-planar, the epipolar constraint can then be implemented to simplify the correspondence searching process because the search will be reduced to a one dimensional search. The process to transform the slanted image planes into co-planar image planes is called rectification. A rectification algorithm is proposed in this chapter and its contents are structured as follow:

- The background of stereovision rectification is given in Section 3.1;
- A ray sketching method is proposed to determine the extrinsic and intrinsic parameters of the system (Section 3.2);
- A rectification algorithm is proposed;
- Some experimental results on rectification are shown in Section 3.3;
- A summary is given in Section 3.4.

3.1 Background of stereovision rectification

In this section, we will introduce some basic concepts in stereovision which includes pinhole-camera model and the use of epipolar geometry in rectification. These concepts will be applied in the following sections.

3.1.1 Pinhole- camera model

The pinhole camera model is widely used in computer vision research. In Figure 3.2, the pinhole camera is modeled by its optical center C and the image plane with its center O . The world coordinates are represented by $O_w \equiv (X_w, Y_w, Z_w)$ and the 2-D image plane coordinates are represented by $O \equiv (x_{im}, y_{im})$. The candle is the 3-D object. The top of the flame is the point $P(X, Y, Z)$, which is projected onto the image point $P'(x_p, y_p)$ given by the intersection of the

image plane with the line containing C and P . The line orthogonal to the image plane passing through the point C intersects the image plane at point O .

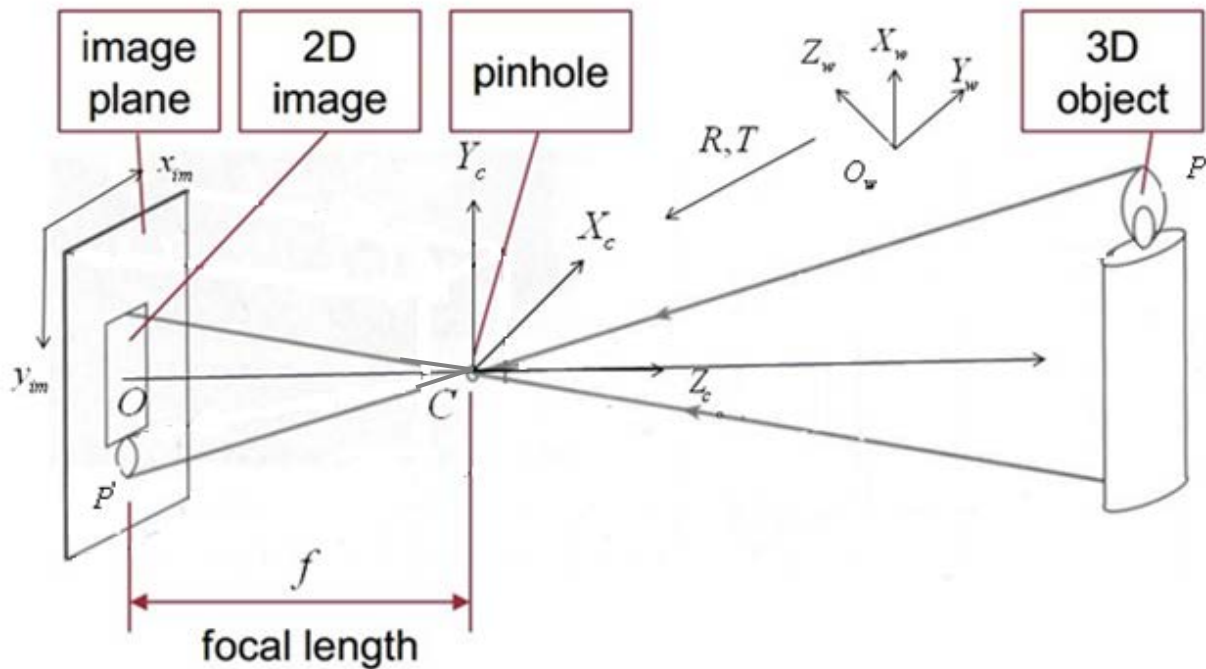


Figure 3.2 Pinhole camera model.

In Figure 3.2, there is a linear transformation in homogenous coordinates which transforms the 3-D world coordinates into 2-D image coordinates. The coordinates of P with respect to the world reference frame are $[X_w, Y_w, Z_w]^T$ and its projection in the image plane is P' whose coordinates are $[x_{im}, y_{im}]^T$ with respect to the image coordinate frame. If we make

$\tilde{P}^i = [x_{im}, y_{im}, 1]^T$ and $\tilde{P}_w = [X_w, Y_w, Z_w, 1]^T$ the respective homogeneous coordinates of P^i and P , the perspective transformation is given by:

$$\varepsilon \tilde{P}^i = C_{R,T} \tilde{P}_w \quad (3.1)$$

where ε is a scale factor. The term $C_{R,T}$ is the perspective projection matrix which is composed of two terms,

$$C_{R,T} = M_{\text{int}} [R, T] \quad (3.2)$$

The matrix M_{int} is a matrix containing intrinsic parameters of the camera with the form as follow:

$$M_{\text{int}} = \begin{bmatrix} \frac{f}{s_x} & 0 & o_x \\ 0 & \frac{f}{s_y} & o_y \\ 0 & 0 & 1 \end{bmatrix} \quad (3.3)$$

where s_x and s_y are the effective size of pixel (in millimeter) in the horizontal and vertical direction, f is the focal length, R is the rotation matrix and T is the translation vector which represents the rigid transformation between the camera reference frame centered at C and the world reference frame O_w .

3.1.2 Introduction of rectification using epipolar constraint

In this section, we will describe the fundamental principles of rectification. According to the epipolar geometry, epipolar lines all intersect at epipoles. If the two image planes of the cameras

are coplanar, the correspondence search is simplified to one dimension - a horizontal line parallel to the baseline between the cameras (Figure 3.3).

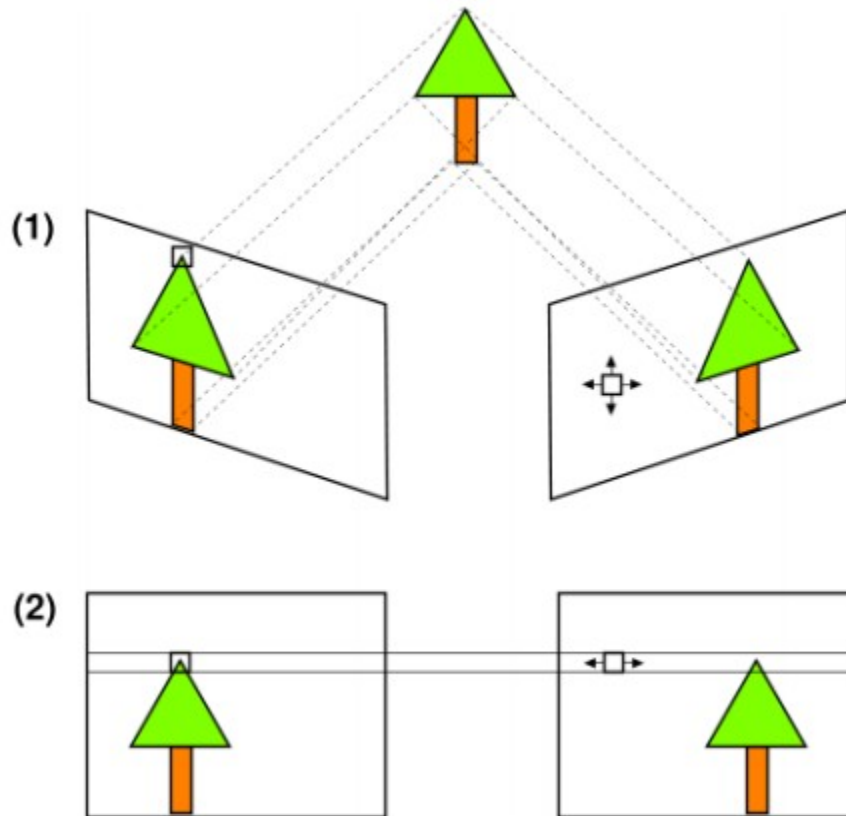


Figure 3.3 Epipolar geometry of two views.

We can describe the fundamental principle of rectification using Figure 3.4. It shows the geometry of the rectified stereovision setup.

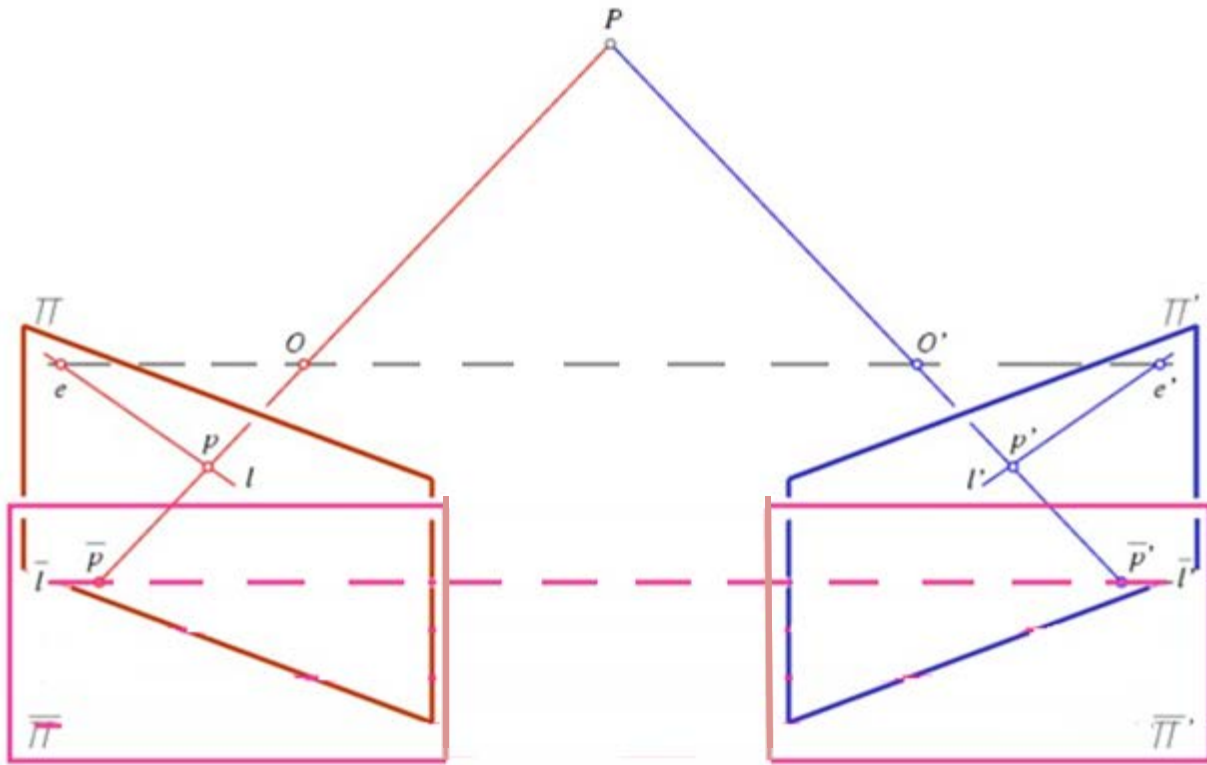


Figure 3.4 Image pair before and after rectification.

In Figure 3.4 image planes Π and Π' are the slanted image planes before rectification and planes $\bar{\Pi}$ and $\bar{\Pi}'$ are the rectified image planes after rectification. Points p and p' are the correspondence points in the slanted image planes and \bar{p} and \bar{p}' are corresponding image points in rectified left and right images. Lines l and l' are the epipolar lines of p' and p respectively and line \bar{l} and \bar{l}' are the epipolar lines of \bar{p}' and \bar{p} respectively.

According to epipolar geometry, the following property can be used in our rectification process.

- If \bar{p} and \bar{p}' are corresponding image points in the rectified left and right images,

$$\bar{p}^T F \bar{p} = 0 \quad (3.4)$$

where F is the fundamental matrix which is expressed in detail by Trucco [1]

- The epipolar line corresponding to \bar{p} is

$$\bar{l} = F \bar{p} \quad (3.5)$$

- The epipolar line corresponding to \bar{p}'

$$\bar{l} = F^T \bar{p}' \quad (3.6)$$

According to the above theory, the rectification process can be achieved when mapping the epipoles to infinity (according to Hartley's algorithm [59]). This concept will be applied in our rectification algorithm (section 3.3).

3.2 Ray-sketching approach to calculate the extrinsic parameters

To proceed with the rectification of the stereo image pair captured by the single-lens stereovision system, some extrinsic and intrinsic parameters of the system must be obtained first. Therefore, in this section, we present a ray-sketching approach to determine the extrinsic parameters.

We shall first present the concept of virtual camera in our single-lens prism based stereovision system [65]

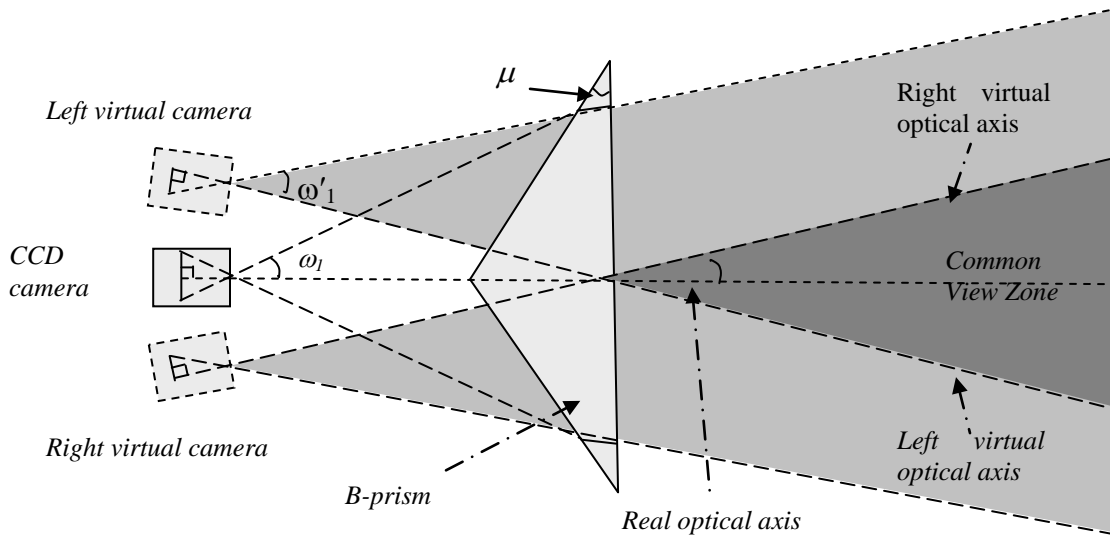


Figure 3.5 Formation of left and right virtual cameras by single-lens bi-prism stereovision system.

3.2.1 Formation of virtual cameras

When a bi-prism is placed in front of a camera, the image plane of this camera will capture two images of the same scene simultaneously (Figure 3.5). The two images can be taken as the images captured by the two virtual cameras which are generated due to the presence of the bi-prism.

In Figure 3.6, a ray-sketching diagram is given which shows the formation of the virtual cameras. We consider the plane which passes through the real camera optical center and intersects the real camera image plane perpendicularly, i.e. $X_w - Z_w$ plane in Figure 3.6. As the point p_c and p_b are chosen as the center point and boundary point of intersection line of the two planes we mentioned above, the Field of View of the left virtual camera is between the two boundary lines defined as below:

1. The optical axis of the virtual camera which can be determined by back-extending the refracted ray of the ray which is along the real camera optical axis;
2. The line which is determined by back-extending the refracted ray of the ray along the real camera FOV boundary line.

The intersection of the two lines above is taken as the virtual camera optical center O_L . In the formation of virtual cameras, we can use the ray-sketching method to obtain the position of the two virtual cameras, orientations and focal lengths which will be used in our rectification algorithm which is shown in Section 3.3.

To determine the above parameters, we shall make the following assumptions:

- i. The properties of the cameras, such as pixel size and focal length are constant;
- ii. The projection of the bi-prism apex line on the camera image plane shall bisect the image plane vertically;
- iii. The prism is symmetrical with respect to its apex line;
- iv. The back plane of the prism is parallel to the image plane of the real camera.

Based on the assumptions above, two virtual cameras can be assumed to be having identical properties and symmetrically positioned with respect to the real camera optical axis.

In the following sections, only the left virtual camera is presented in Figure 3.6, because the right virtual camera formation can be inferred in a similar way. The left virtual camera image plane corresponds to the right half of the real image plane and the right virtual camera image plane corresponds to the left half of the real image plane.

3.2.2 Determination of the extrinsic parameter using the ray-sketching method

In this section, we will propose a ray-sketching method to obtain the extrinsic parameter of the single-lens bi-prism stereovision system. We have discussed and report the use of the extrinsic parameters of this system in [110]. In the following section we will describe how to obtain these parameters and how to use them in our rectification algorithm.

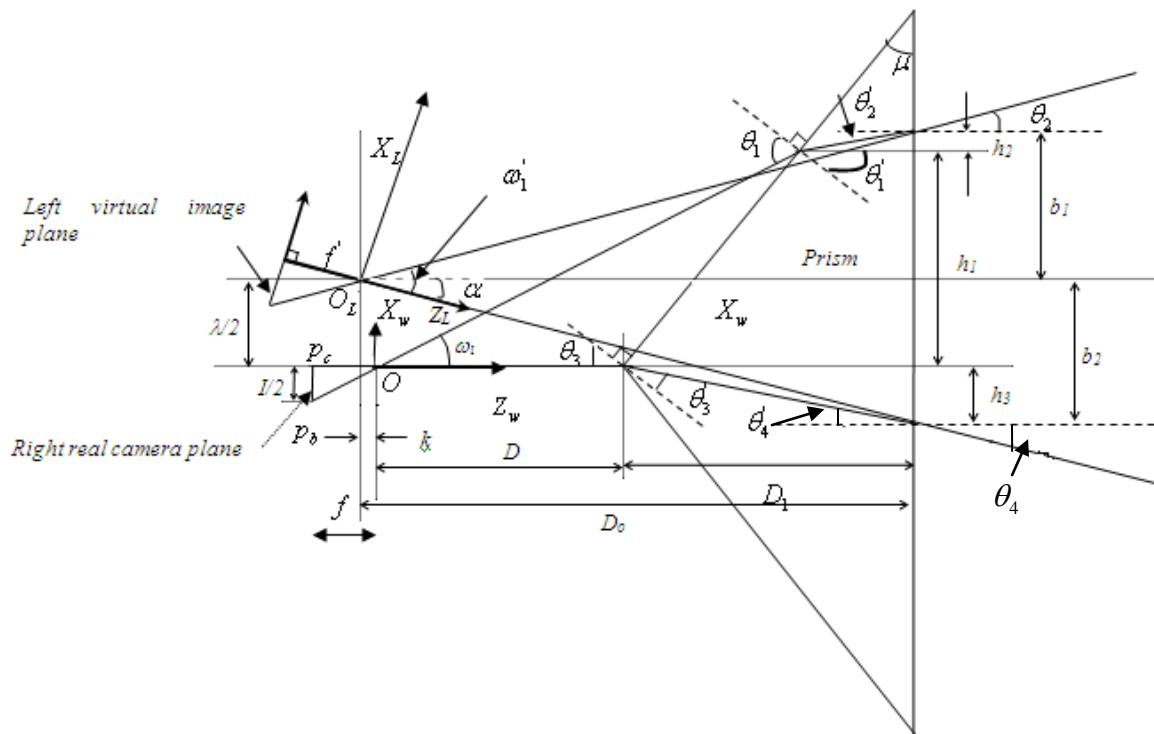


Figure 3.6 Relationship between left virtual camera and real camera in single-lens stereovision system using bi-prism.

Figure 3.6 shows the top view of the system setup schematically, in which the world coordinate system is represented by (X_w, Y_w, Z_w) and it is fixed on the real camera optical center O ; the Left Virtual Camera Coordinate System ((LVCCS: X_L, Y_L, Z_L)) is fixed on the left virtual camera optical center O_L ; the Left Virtual Image Coordinate System ((LVICS: x_L, y_L, z_L)) is fixed on the center of left virtual image plane. The real camera optical axis Z_w can be considered as the demarcation line that bisects the setup into two left and right halves. In Figure 3.6, the part below Z_w is the right half of the real camera image plane. The z-axis of LVCCS, Z_L , is along the back-extending ray of the refracted ray of the ray which is along the real camera optical axis.

Table 3.1 Parameters of single lens bi-prism stereovision system

Parameters	Definition
μ	the corner angle of the prism
α	the rotation angle of the left virtual camera with respect to the real camera. It is equal to the rotation angle of the right virtual camera with respect to the real camera
f	the focal length of the camera and f' is the focal length of the virtual cameras
D	the distance of the real camera optical center from the apex of prism
D_1	the distance between the bi-prism apex and its back plane
D_0	the distance of the real camera optical center from the back plane
n	the refractive index of the prism glass material
k	the distance between the optical center of left image plane (O_L) and the center of real image plane (O) in Z_w -axis
I	the camera image plane width

Among the parameters above, μ , n , D and D_1 are assumed to be known and I can be read from camera's technical manual.

In Figure 3.6, $\theta_1, \theta_2, \theta_3, \theta_4, \theta_1', \theta_2', \theta_3'$ and θ_4' are formed by the refracted ray of the prism. From the law of refraction, we can obtain:

$$\frac{\sin \theta_1}{\sin \theta_1'} = \frac{\sin \theta_2}{\sin \theta_2'} = \frac{\sin \theta_3}{\sin \theta_3'} = \frac{\sin \theta_4}{\sin \theta_4'} = n \quad (3.7)$$

We can obtain the following equations based on the geometrical relationship in Figure 3.6:

$$\begin{cases} \theta_3 = \mu \\ \theta_4 = \mu - \theta_3' \\ \theta_1 = \mu + \omega_1 \\ \theta_2' = \theta_1' - \mu \end{cases} \quad (3.8)$$

From Equation (3.7) and (3.8), we obtain the following angles:

$$\alpha = \theta_4 = \arcsin\left(n \sin\left(\mu - \arcsin \frac{\sin \mu}{n}\right)\right) \quad (3.9)$$

$$\theta_2 = \arcsin\left(n \sin\left(\arcsin \frac{\sin(\mu + \omega_1)}{n} - \mu\right)\right) \quad (3.10)$$

In Figure 3.6, we can see that

$$\tan \omega_1 = I/2f \quad (3.13)$$

$$\omega_1' = \theta_2 + \theta_4 \quad (3.12)$$

$$\begin{cases} b_1 = D_0 \tan \theta_2 \\ b_2 = D_0 \tan \theta_4 \end{cases} \quad (3.13)$$

$$\left\{ \begin{array}{l} h_1 = \frac{D}{\cot \omega_1 - \tan \mu} \\ h_2 = (D_1 - h_1 \tan \mu) \tan \theta_2' \\ h_3 = D_1 \tan \theta_4' \end{array} \right. \quad (3.14)$$

We also note the geometrical relationship between b_1, b_2, h_1, h_2 and h_3

$$b_1 + b_2 = h_1 + h_2 + h_3 \quad (3.15)$$

Thus, from Equations (3.13), (3.14) and (3.15),

$$D_0 = \frac{h_1 + h_2 + h_3}{\tan \theta_2 + \tan \theta_4} \quad (3.16)$$

We can obtain:

$$k = D_0 - D - D_1 = \frac{\left[\frac{D}{\cot \omega_1 - \tan \mu} (1 - \tan \mu \tan \theta_2') + t(\tan \theta_2' + \tan \theta_4') \right]}{\tan \theta_2 + \tan \theta_4} - D - D_0 \quad (3.17)$$

$$\lambda = 2(b_2 - h_3) = \frac{2 \tan \theta_4}{\tan \theta_2 + \tan \theta_4} \left[\frac{D}{\cot \omega_1 - \tan \mu} (1 - \tan \mu \tan \theta_2') + t(\tan \theta_2' + \tan \theta_4') \right] - 2t \tan \theta_4' \quad (3.18)$$

The focal length of the virtual camera (f') and the focal length of real camera (f) can be expressed as:

$$\left\{ \begin{array}{l} f = \frac{I}{2 \tan \omega_1} \\ f' = \frac{I}{2 \tan \omega_1'} \end{array} \right. \quad (3.19)$$

where

$$\omega_1' = \theta_2 + \theta_4 \quad (3.20)$$

For small angles, we have

$$\tan \theta = \sin \theta = \theta \quad (3.21)$$

Now, we substitute Equations (3.9), (3.10) (3.21) into (3.20):

$$\omega_1' = (n\mu - \mu) + (\mu + \omega_1 - n\mu) = \omega_1 \quad (3.22)$$

Therefore, from Equation (3.19)

$$f' = f \quad (3.23)$$

Therefore the focal length of the virtual cameras is equal to that of the real camera. The rotational angle α can be obtained from Equation (3.9) and R_L is given by Equation (3.24)

Then we get the rotation angle α and the rotation matrix can be expressed as follow:

$$R_L = \begin{bmatrix} \cos \alpha & 0 & \sin \alpha \\ 0 & 1 & 0 \\ -\sin \alpha & 0 & \cos \alpha \end{bmatrix} \quad (3.24)$$

The translation vector is:

$$T_L = \begin{bmatrix} OC_{Lx} \\ OC_{Ly} \\ OC_{Lz} \end{bmatrix} \quad (3.25)$$

where $OC_L(OC_{Lx}, OC_{Ly}, OC_{Lz})$ are the coordinates of the optical center of the left virtual camera with respect to the world coordinate system.

Camera intrinsic parameter of the left virtual camera is shown in Equation (3.26)

$$M_{L\text{int}} = \begin{bmatrix} -f_x & 0 & o_x \\ 0 & -f_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \quad (3.26)$$

where:

- f_x :focal length of the virtual camera in the x-direction;
- f_y :focal length of the virtual camera in the y-direction;
- o_x :coordinates of the virtual camera image center in the x-direction of left virtual camera image plane;
- o_y :coordinates of the virtual camera image center in the y-direction of left virtual camera image plane.

From [1], the projection Matrix is given as:

$$P_L = M_{L\text{int}}[R_L | T_L] \quad (3.27)$$

In the same way,

$$P_R = M_{R_{\text{int}}}[R_R|T_R] \quad (3.38)$$

where $M_{L_{\text{int}}}$ and $M_{R_{\text{int}}}$ are the intrinsic matrices of the left and right virtual camera respectively, R_L and R_R are the rotation matrices of the left and right virtual camera, respectively; and T_L and T_R are the translation vectors of the left and right virtual camera, respectively. The development described in this section will be important in our rectification algorithm.

Comparing with the previous work proposed by K.B. Lim, D.L. Wang and W.L. Kee [114] which choose the three rays in a random plane which passes through the real camera optical center and not perpendicular to the image plane, our method is focusing on the plane which passes through the real camera optical center and intersects the real camera image plane perpendicularly, i.e. $X_w - Z_w$ plane which is shown in Figure 3.6, the rays we studied in our algorithm has no shift in Y_w direction when it refracted by the bi-prism which significantly reduce the computation cost comparing with planes studied in [114] because we can recover the position of the virtual camera only use the two rays described in Section 3.2 instead of the three rays using in [114].

3.3 Rectification algorithm

In this section we will describe the process of a rectification algorithm for the image pair captured by single-lens bi-prism stereovision system.

The goal of rectification is to use the intrinsic parameters of cameras and extrinsic parameters of the system (R and T) to compute the image transformation that makes the conjugated epipolar line collinear and parallel to the horizontal line. The intrinsic and extrinsic parameters have been obtained in Section 3.2. In this section, we shall develop the rectification transformation matrix which projects the image points on both the left and right image planes of the virtual cameras to

two coplanar planes as shown in Figure 3.7. This transformation makes the epipolar lines parallel to a horizontal scan-line and move the epipole of the new image planes of the virtual cameras to infinity (shown in Figure 3.7).

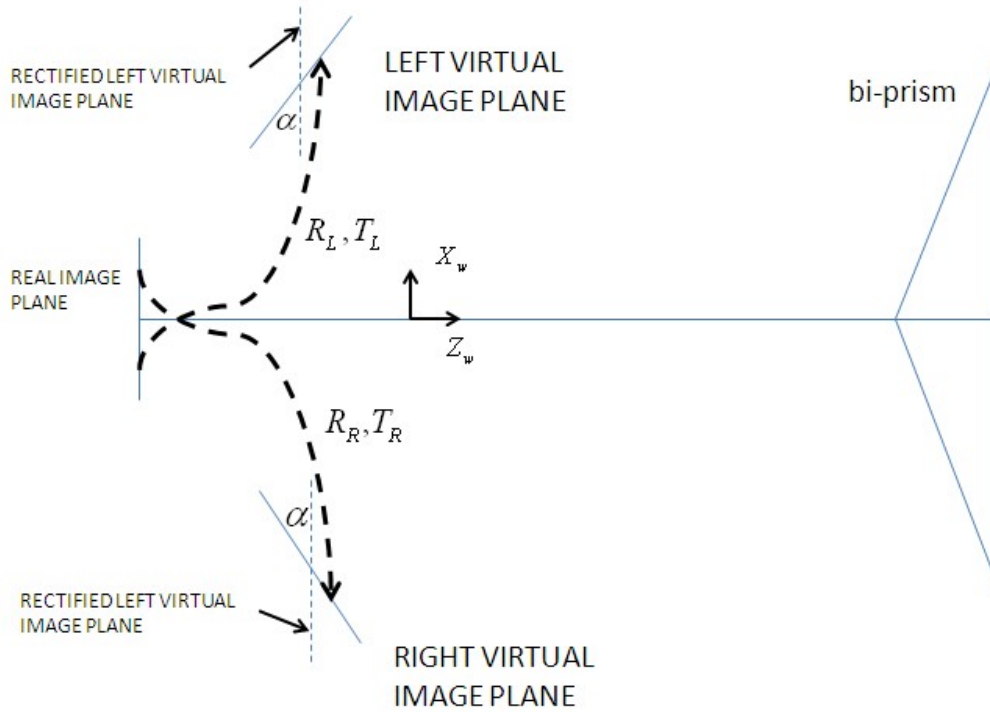


Figure 3.7 Sketch map of rectification algorithm.

To carry out the transformation, we build a new virtual camera coordinates as

$$R_{rect} = \begin{bmatrix} u_1^T \\ u_2^T \\ u_3^T \end{bmatrix} \quad (3.29)$$

Based on our system setup (shown in Figure 3.6) we define

u_1 as the new X-axis of the virtual camera coordinates which is chosen to be parallel to baseline;

$$u_1 = \frac{T}{\|T\|} = \frac{OC_{Lx} - OC_{Rx}}{\|OC_{Lx} - OC_{Rx}\|} \quad (3.30)$$

u_2 as the new Y-axis of the virtual camera coordinates which is chosen to be the same as the original Y_w in world coordinate frame;

u_3 as the new Z-axis of the virtual camera which equals to the cross product of u_1 and u_2 .

Thus, the final rectification transformation matrices from virtual camera planes to rectified virtual camera planes are shown in Equation (3.31) and (3.32):

$$R_l = R_L R_{rect} \quad (3.31)$$

$$R_r = R_R R_{rect} \quad (3.32)$$

The process of the rectification algorithm is shown as follow:

Algorithm 3.1: Algorithm of rectification:

- 1) Calculate the rotation matrix R_L and translation vector T_L using Equation (3.24) and (3.25)
- 2) In the world coordinate frame, the relationship between camera point $p_r(x, y, -f)$ on the right half of real camera and the left virtual camera point p_{lv} on the left virtual camera is:

$$p_{lv} = R_L(p_r - T_L)$$

- 3) Build the matrix R_{rect} as in Equation (3.29)
- 4) Set $R_l = R_L R_{rect}$ and $R_r = R_R R_{rect}$
- 5) For each point p_{lv} in the left virtual camera, we compute

$$R_l p_{lv} = [x', y', z']$$

where $[x', y', z']$ are the coordinates of rectified point in world coordinate frame

- 6) Compute coordinates of the rectified point with respect to the left camera coordinate frame as:

$$p_l' = \frac{f}{z'} [x', y', z'] = \frac{f}{z'} R_l p_{lv}$$

- 7) Compute coordinates of each rectified point with respect to the rectified image plane

$$p_{rect} = M_{Lint} p_l'$$

- 8) Repeat the previous step for points in the left half plane of the real image plane to get the coordinates of the points in the rectified right virtual image plane.

3.4 Experimental results

Our system is set up with a prism placed in front of a CCD camera which is shown in Figure 3.1. It captures two views of the same scene at the same time. In our experiment, we are using the proposed algorithm on three stereo image pairs and the final results are shown in Figure 3.8 to 3.10. The three images we have chosen are named as “book and card”, “three objects”, and “medicine” which are captured using our stereovision system with different prism angles ($\mu = 6.4^\circ, 10^\circ, 21.6^\circ$). In each figure, a) is the left and right image taken directly by our system and b) is the rectified image pair obtained by our rectification algorithm. Three random points are chosen in the left image and their epipolar lines are shown in the right image (Figure 3.8(a), 3.9(a) and 3.10(a)). We randomly choose four points in the left rectified image and their epipolar lines are displayed in the right rectified image respectively in Figure 3.8(b), 3.9(b) and 3.10(b). The parameters we have used to capture these image pairs are shown as follow (see Figure 3.6).

- The focal length of two virtual camera: $f' = 25mm$;
- Refraction index of material of bi-prism: $n = 1.43$;
- The distance between optical center of camera and apex of the prism $D = 200mm$
- The angle of bi-prism: $\mu = 6.4^\circ, 10^\circ, 21.6^\circ$ (for Figure 3.8-3.10 respectively);
- The distance between the optical center of the camera to the back plane of the prism $D_0 = 205.6mm, 208.8mm, 219.8mm$ (for Figure 3.8-3.10 respectively).

The system is configured under the following conditions:

1. The projection of the bi-prism apex line on camera image plane shall bisect the image plane vertically;

2. The prism is placed symmetrical with respect to its apex line;
3. The back plane of the prism is parallel to the image plane of the real camera;
4. Images of object and back ground are captured by the single-lens stereovision system using bi-prism;
5. Apply rectification algorithm proposed in this chapter to obtain the rectified image pair.

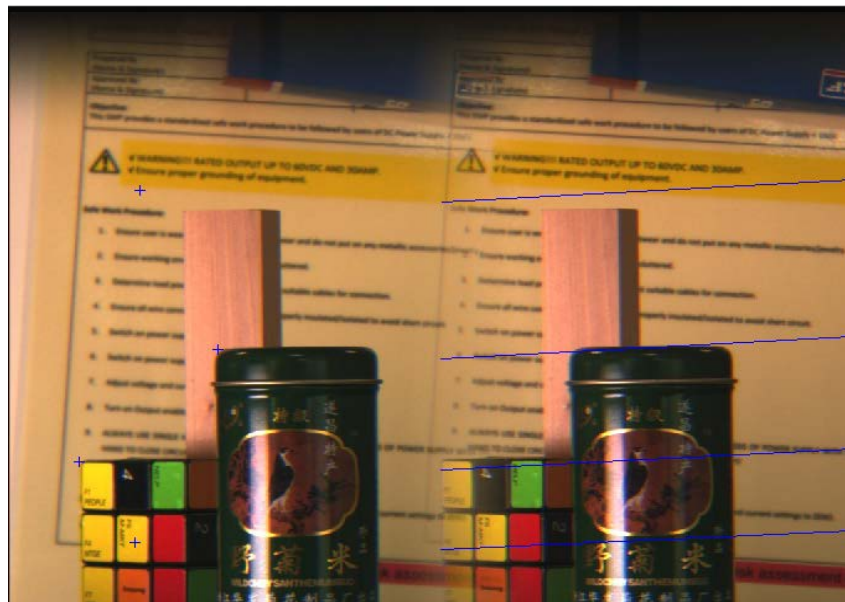


(a)

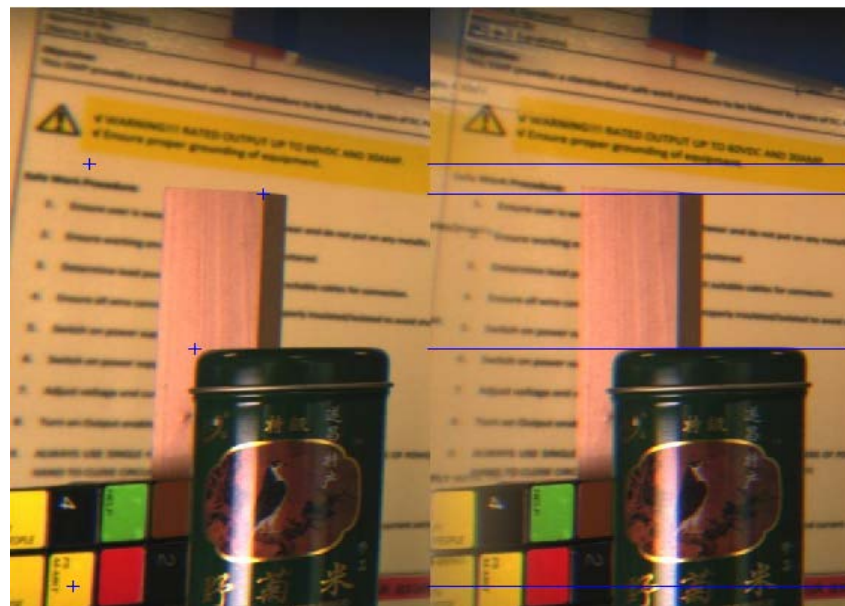


(b)

Figure 3.8 “Book and card” image a) left and right image; b) rectified left and right image (epipolar lines shown in the right images).



(a)



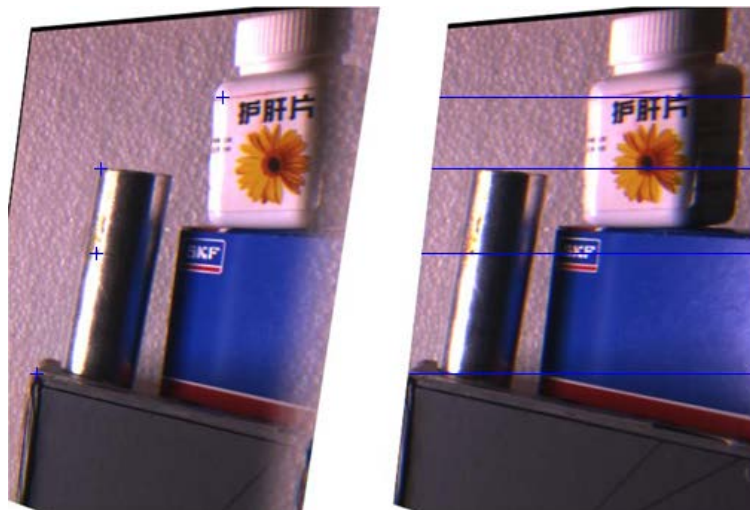
(b)

Figure 3.9 “Three objects” image a) left and right image; b) rectified left and right image

(epipolar lines shown in the right images).



(a)



(b)

Figure 3.10 “Medicine” image a) left and right image; b) rectified left and right image (epipolar lines shown in the right images).

From the experimental result shown in Figure 3.8 -3.10, we can find that after implementing the proposed rectified algorithm, the points selected in the rectified left image are rectified which is proved by the fact that their epipolar lines in the rectified right image appear as horizontal scan-lines (shown in Figure 3.8(b), 3.9(b) 3.10(b)).

More detailed experimental results and analysis of the performance of our proposed rectification algorithm are shown and discussed in Section 5.3

3.5 Summary

In this chapter, a rectification algorithm was proposed. A ray sketching method is used to obtain the system parameters which are used in the rectification algorithm. The goal of the stereo rectification is to transform the slanted virtual camera image planes to coplanar image planes which reduced the searching cost of the stereo correspondence. The proposed method is easy to operate because the number of parameters that need to be computed is small. More detailed experimental results will be shown in Section 5.3 which shows the accuracy and advantages of the proposed method. In addition, the results of the proposed method will be compared with the one using conventional method. In the next chapter, a novel segment-based stereo matching algorithm will be proposed.

Chapter 4 Segment-based Stereo Matching Algorithm Using Belief

Propagation

In this chapter, a novel segment-based Stereo Matching algorithm using belief propagation is proposed. This stereo matching algorithm is used to determine the disparity map and depth of the scene of interest from the image pair captured by a single-lens bi-prism stereovision system.

The flow chart of our proposed algorithm is shown in Figure 4.1. Beginning with a rectified image pair, we first apply a self-organizing map segmentation algorithm (Section 4.2) to segment the left image (reference image) into regions of same colors; at the same time a local stereo matching method (Aggregation method) is proposed (Section 4.3) to obtain the initial disparity map which is used as the initial data for the subsequent steps.

Then, we employ a disparity plane fitting method to assign a disparity plane for each segment (Section 4.4). The refinements to the disparity planes are then carried out by filtering outliers and merging neighboring segments that share similar disparity plane (Section 4.5). Next, we create an energy function to measure the matching cost for all segments (Section 4.6) and use a belief propagation method to optimize the energy function to obtain the accurate disparity planes (Section 4.7). The disparity map is obtained finally and the depths of the points in the scene are recovered (Section 4.8).

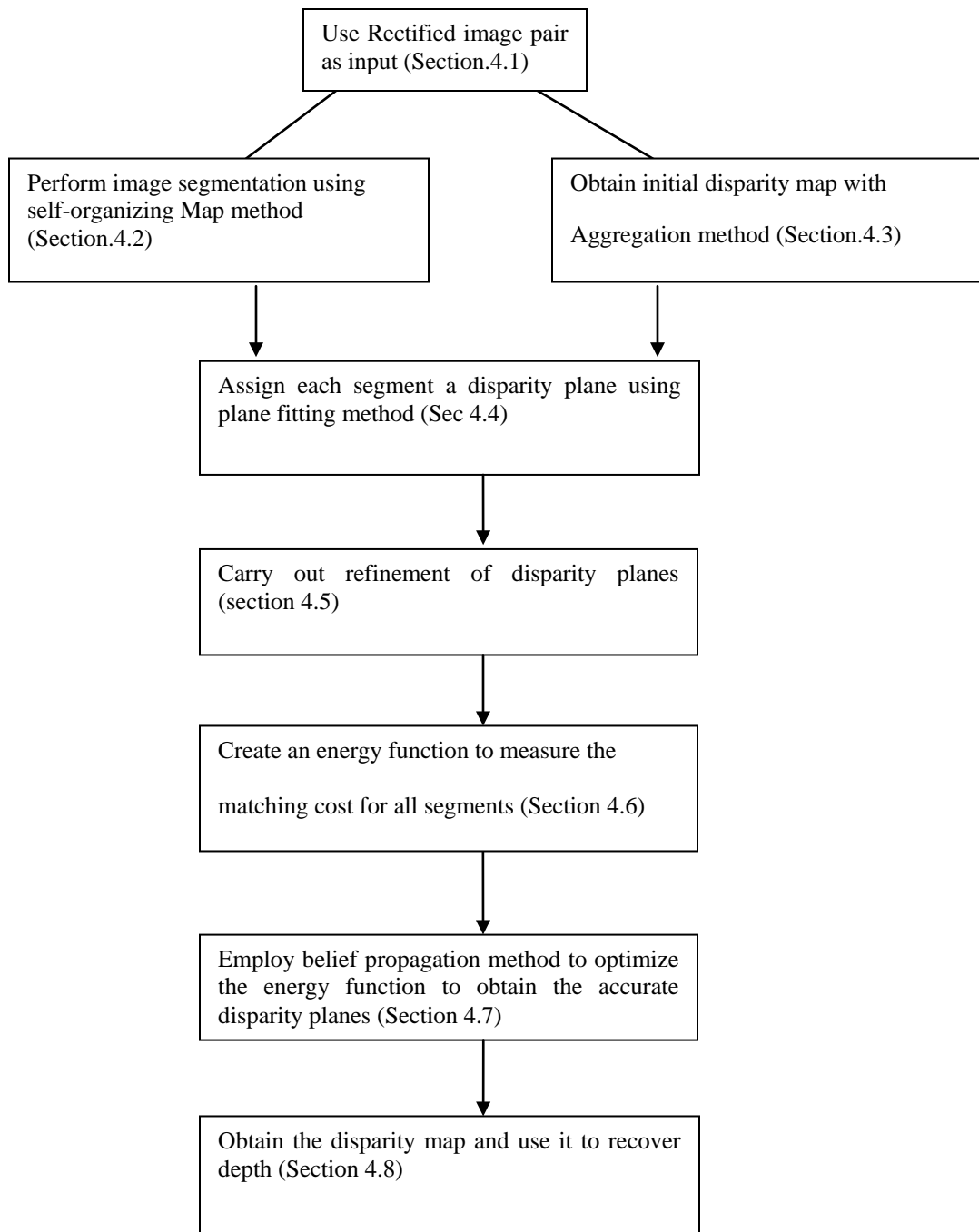


Figure 4.1 Procedure of our segment-based stereo matching algorithm.

4.1 Rectified image pair

In stereovision, the two image planes are usually inclined at an angle to each other, as we have discussed in Section 1.1.2. However, if the two image planes can be transformed to be coplanar the searching of stereo correspondence is significantly simplified and the computational cost of stereo matching algorithms is reduced as a result. To benefit from this property, in this chapter, the image pairs used are:

1. The stereo image pair captured by single-lens bi-prism system is rectified using the rectification algorithm (presented in Chapter 3)
2. The stereo image pair chosen from Middlebury Stereo database (the standard rectified image pair)[70]

4.2 Image segmentation

Image segmentation is the process of segmenting an image into objects or elements that are coherent under some criterion which is discussed by Cheng and Jiang [39]. Color segmentation is normally used to segment the color image into regions of homogenous color. The segmentation results will be used for the subsequent steps in our work. In this thesis, a self-organizing map algorithm is used to segment the reference image (left image of the stereo image pair) and the segmentation results will be used in the subsequent steps of our algorithm.

In general, segment-based stereo matching algorithms will have the two assumptions below:

1. Disparity values vary smoothly in segmented regions.
2. Disparity discontinuities only occur on region boundaries.

Based on the two assumptions, we can segment the reference image and assign a disparity to each segment instead of each pixel which requires less effort in computing disparity. It is one of the advantages of our proposed algorithm.

Self-organizing map [26] is a segmentation algorithm based on Neural Network Theory which is also called self-organizing map segmentation algorithm (SOMSA) in many works [46-48].

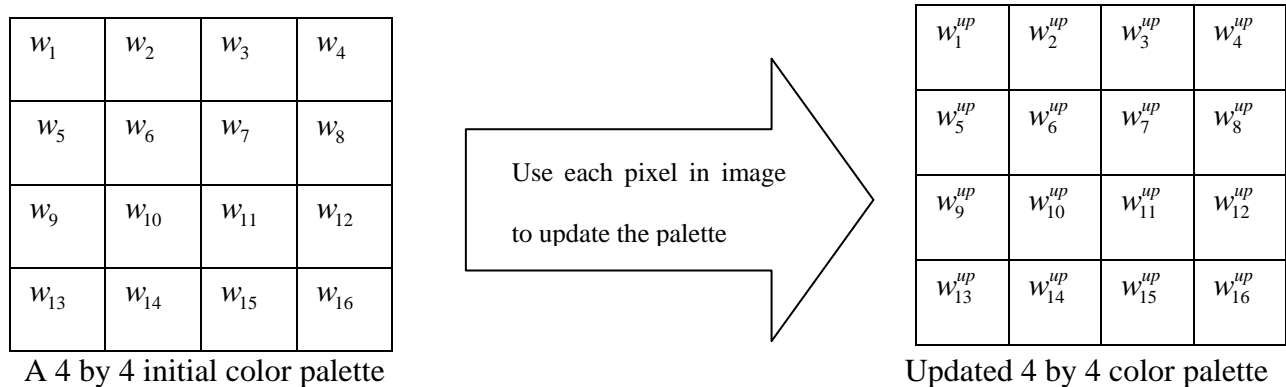


Figure 4.2 Process of the color palette updating.

In SOMSA, The color segmentation process using self-organizing map has a 3-D vector (red, blue, and green values) to represent the color of every pixel in an image, according to the RGB

color standard. An initial color palette is first generated randomly shown in Figure 2.4 (in this example, an 8 by 8 initial colors is generated) and the colors in the palette are represented as a weight vector $w_i = [w_{ir}, w_{ig}, w_{ib}]^T$. Subsequently the color of each pixel, denoted as $x = [r, g, b]^T$, will be chosen as an input to update the color palette.

In detail, the color segmentation process using SOMSA involves 3 steps:

1) Randomly generate a color palette in which the colors are represented by its weight vector

$$w = [w_r, w_g, w_b]^T \text{ (an example of a 4 by 4 color palette is shown in Figure 4.2);}$$

2) Choose every pixel in the image as input to update the color palette;

3) Map each pixel in the original image to the nearest color in the updated palette.

The reference image (left image of the stereo image pair) is decomposed into segments with homogenous colors which are in the updated palette. Hence, in SOM color segmentation, only a few distinct and dominant colors are desired.

The detailed process of SOMSA is shown as follows:

Algorithm 4.1: Algorithm for image segmentation:

1. Initialization of the parameters: Initialize the neighborhood variance, $\sigma = \sigma_0$; learning rate $\eta = \eta_0$, at time iteration, $t = 0$ and the weight vector for each node as

$$w_i = [\text{rand}(0,0.2), \text{rand}(0,0.2), \text{rand}(0,0.2)]^T \text{ where } i = 1 \text{ to } N$$

2. Present a randomly chosen image pixel, $x(t) = [r(t), g(t), b(t)]$ as input data to the network
3. Calculate Euclidian distance between the input vector $x(t)$ and weight vector $w(t)$

$$\|x(t) - w_i(t)\| = \sqrt{[r(t) - w_{ir}(t)]^2 + [g(t) - w_{ig}(t)]^2 + [b(t) - w_{ib}(t)]^2} \quad (4.1)$$

The weight vector with the smallest Euclidian distance with the input vector is chosen as the best matching unit (BMU).

4. Update the weight vector of the BMU and its neighbors using

$$w_i(t+1) = w_i(t) + \eta(t)H(t)(x(t) - w_i(t))$$

where $H_i = \exp\left(-\frac{\text{dist}}{2\sigma^2(t)}\right)$ and dist is the distance between the weight vectors of BMU and the current

node computed as in Equation 4.1.

5. Update leaning rate: Let $t = t + 1$ and repeat step 2 to 4 and update η and σ as follow

$$\eta(t) = \eta_0 \exp(-t) \text{ and } \sigma(t) = \sigma_0 \exp(-t)$$

6. Repeat step 2-5 until the following convergence criterion is satisfied,

$$e = \frac{1}{N} \sum_{i=0}^{N-1} \|w_i(t) - w_i(t+1)\|^2 \leq e_{\min} = 10^{-4}$$

7. Update the color palette (eg. Figure 2.4) using the updated weight vector.
8. Forming the new segmented image: In the image, replace each original pixel color by its nearest color in the color palette.
9. Clustering pixels: Pixels with the same color will be group into respective segments. We will then have an image with regions segmented according to colors.

Using the method shown above, a color based segmented image is obtained based on the similarity of colors. Figure 4.3-4.5 shows the reference image and the segmentation result using SOM segmentation algorithm.

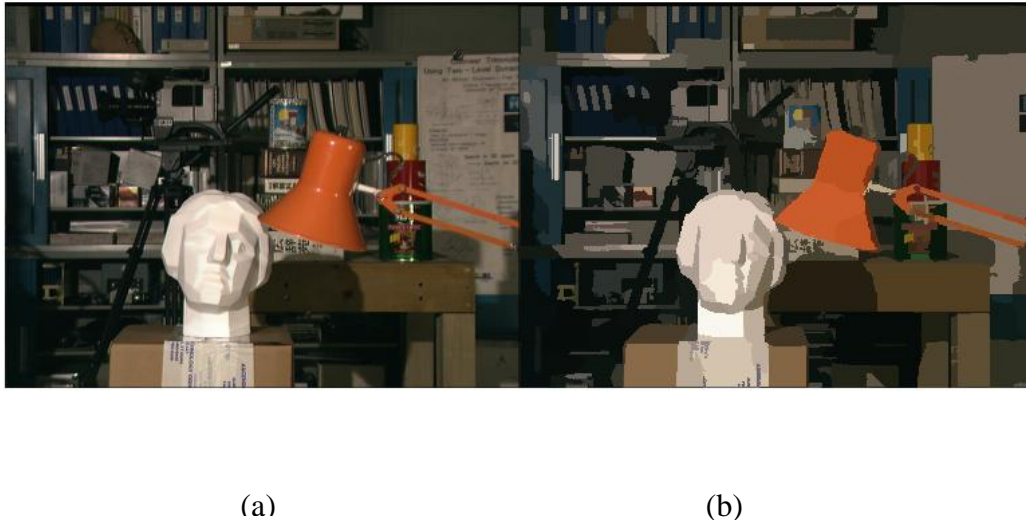


Figure 4.3 Segmentation results of Tsukuba: (a) Reference image (Tsukuba/Middlebury); (b) Segmentation result.

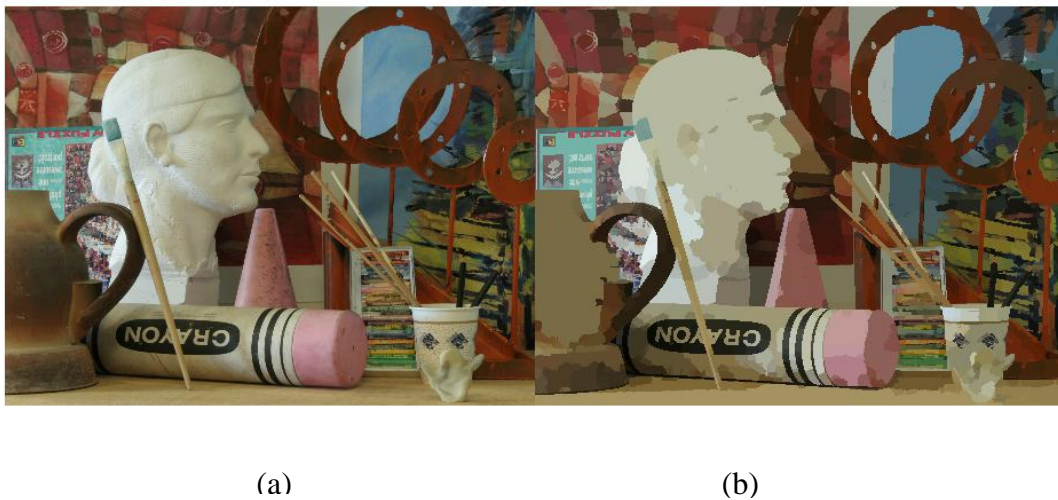


Figure 4.4 Segmentation results of Art: (a) the reference image (Art/Middlebury (2005)); (b) Segmentation result.



(a)

(b)

Figure 4.5 Segmentation result of Computer: (a) Reference image (Computer/Middlebury (2005)); (b) Segmentation result.

Figure 4.3(b), 4.4(b), and 4.5(b) show the segmentation results using self-organizing map segmentation algorithm. We note that the pixels with similar color are clustered into a same segment. In addition, the segment boundaries are clear, which shows that the segmentation based on color is successfully accomplished by the proposed method described above.

We can also note that the segmentation result is better in 4.4(b) than in 4.5(b). The reason is that the color differences between the segments are more distinctive as shown in 4.4 (b) than that in 4.5(b). This is due to the fact that the distances between the weight vectors of the segments are larger in 4.4(b) than that in 4.5(b). The large distance between two weight vectors of two connected segments implies that the color between them is very distinct. As a result, the boundaries between segments are clearer in 4.4(b).

The segmentation results will be used to fit the segments into disparity planes in Section 4.4.

After this segmentation process, a segmented reference image (left image) is obtained, which be used in Section 4.4. In the next section a local matching method is proposed to determine the initial disparity of pixels in the image.

4.3 Disparity initialization using aggregation method

The purpose of this section is to carry out stereo correspondence search using a local method called the aggregation method. The purpose is to determine the initial disparity maps. It uses local information of pixels to obtain an initial disparity map which contains disparity values of all the pixels in the image. Note that the image pair used has been rectified. As we have already discussed in Section 2.1 on epipolar geometry, given a point in the reference image (the left image), in a rectified image pair, its correspondence point must exist on the same horizontal scan line in the right image. Therefore we only need to search all points on this line in the right image to determine the stereo correspondence point. The disparity is calculated by taking the difference in the x coordinates of the two corresponding points.

We shall describe the method for determining the disparity plane in detail below:

1. Given a rectified image pair, the left image is taken as the reference image. The disparity value of every pixel in the left image is evaluated, at the end of which a left disparity map denoted as D_L is obtained. Note that D_L contains the disparity of every pixel in the left image. Figure 4.7 (b) shows an example of an initial disparity map.
2. Then, the right image is next taken as the reference image, and the disparity of every pixel in the right image is evaluated. A disparity map denoted as D_R is then obtained.

The two disparity maps D_L and D_R will be used in the Refinement of the disparity planes, which will be presented in Section 4.5.

The procedure in obtaining the disparity map is presented below. We shall use the left image as the reference image for illustration.

Given $P_l(x_l, y_l)$ a point in left image, $P_r(x_l + d, y_l)$ is a point on the same scan-line (epipolar line) with $P_l(x_l, y_l)$. P_r is one of the potential corresponding points with a disparity d , according to epipolar geometry.

To determine the stereo correspondence of the point P_l , we define a function which measures the total matching cost if the disparity of the two correspondence points is d :

$$c_{all}(x, y, d) = c(x, y, d) + c_{rank}(x, y, d) \quad (4.2)$$

Equation (4.2) consists of two terms:

- 1) The first term $c(x, y, d)$ is the matching cost of points $P_l(x, y)$ and $P_r(x + d, y)$ with a disparity of d . It is used to measure the similarity of the two chosen points. It is the sum of square difference (SSD) in intensity between the points in the aggregation window centered at P_l and those in the aggregation window centered at P_r (Figure 4.6). We compute this term using Equation (4.3)

$$c(x, y, d) = \frac{1}{9} \left(\sum_{i=-1}^1 \sum_{j=-1}^1 \left| I^2(x+i, y+j) - I^2(x+i+d, y+j) \right| \right) \quad (4.3)$$

where $I(x, y)$ is the intensity of point (x, y)

We normalize $c(x, y, d)$ using Equation (4.4)

$$c'(x, y, d) = \frac{c(x, y, d) - c_{\min}(x, y, d)}{c_{\max}(x, y, d) - c_{\min}(x, y, d)} \quad (4.4)$$

where $c_{\min}(x, y, d)$ and $c_{\max}(x, y, d)$ are the minimum and maximum value of $c(x, y, d)$ respectively.

Note that in a rectified image pair, stereo correspondence points are on the same scan-line, so d is in the range from 0 to the maximum number of pixels on the epipolar line (which is denoted as m). So for pixel $P_l(x, y)$, $c(x, y, d)$ will be computed m times and we record $c_{\min}(x, y, d)$ and $c_{\max}(x, y, d)$ among these m values. Then $c'(x, y, d)$ is obtained.

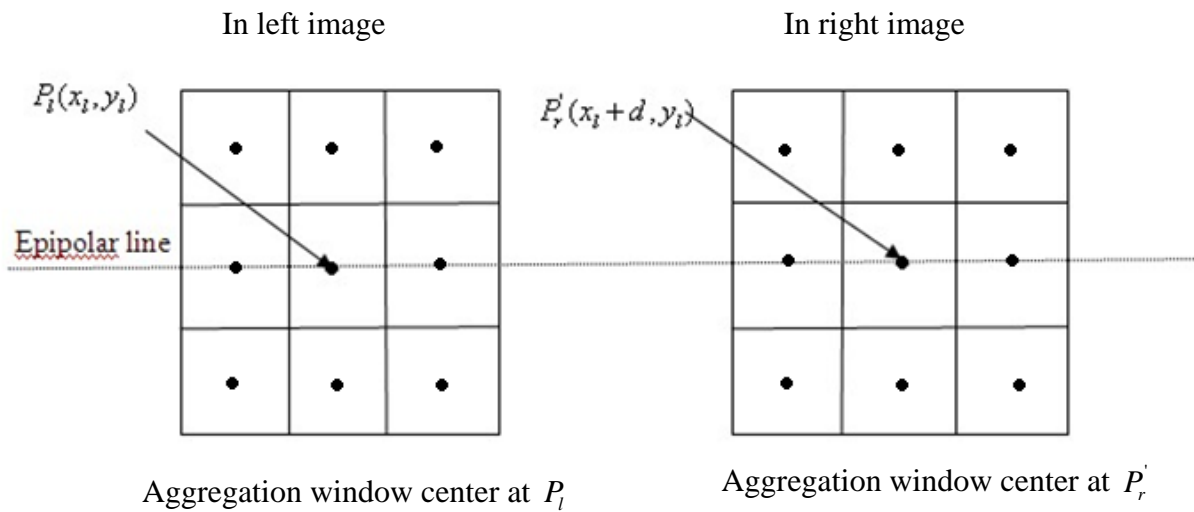


Figure 4.6 Aggregation windows (3 by 3) of the point P_l and potential correspondence point P_r' .

- 2) The second term $c_{rank}(x, y, d)$ is a measure of similarity between points P_l and P_r' based on the intensity distribution of their neighboring points. As presented in [42], if the center of the aggregation window is point $P(x, y)$, the number of points in the window whose intensity is less than the intensity of the center point P is denoted as $n(P(x, y))$, then the term c_{rank} is defined as Equation (4.5)

$$c_{rank}(x, y, d) = \left| n(P_l(x, y)) - n(P_r(x + d, y)) \right| \quad (4.5)$$

We note that c_{rank} is a function of d which can be used to compare the similarity of the two stereo correspondence points and the range of $c_{rank}(x, y, d)$ is from 0 to 8,

We normalize c_{rank} using Equation (4.6),

$$c'_{rank}(x, y, d) = \frac{c_{rank}(x, y, d) - 0}{8 - 0} = \frac{c_{rank}}{8} \quad (4.6)$$

We rewrite Equation (4.2) with the normalized terms (Equation (4.4) and (4.6)):

$$c_{all}(x, y, d) = c'(x, y, d) + c'_{rank}(x, y, d) \quad (4.7)$$

The combination of the two terms in Equation (4.7) strengthens the correctness in the measurement of the similarity between corresponding points and the accuracy of the disparity map obtained by method is higher than the traditional aggregation windows algorithms proposed in [42, 43, 62] (shown in Table 4.1).

c_{all} will be computed between P_l and each point on the epipolar line in the right image. The disparity that minimizes c_{all} is chosen as the initial disparity of P_l . We perform the same steps for every pixel in the left image and then the disparity map could be obtained using Equation (4.8).

$$D_L(x, y, d) = \arg \min c_{all} \quad (4.8)$$

The same procedure is carried out when choosing the right image as reference and we will obtain the disparity map D_R . Figure 4.7 and Figure 4.8 show the initial disparity plane we have obtained using the algorithm proposed in this section.



Figure 4.7 a) Reference image (Computer/Middlebury(2005)); b) initial disparity map.



Figure 4.8 a) Reference image (Arts/Middlebury(2005)); b) initial disparity map.

We also compare the initial disparity map obtained by our method with three different approaches proposed in [42, 43, 62] (which are the Line segment method, Non-parametric transform method and Census transformation method, respectively) using the standard images “Tsukuba, Teddy, Venus, and Cones” from Middlebury Database [70].

The Percentage of Incorrect Matching Pixels, computed by Equation (4.9) in the whole image is used to evaluate the accuracy of the four algorithms. Table 4.1 shows the results.

$$B = (n_{ic} / N) \times 100\% \quad (4.9)$$

where n_{ic} is the number of incorrect matching pixels (comparing with the ground-truth map supplied by [70]) and N is the total number of pixels in the whole reference image.

The pixel is identified as incorrect matching pixel if the condition expressed in Equation (4.10) is met:

$$|d'(x, y) - d_0(x, y)| > 1 \text{ pixel} \quad (4.10)$$

where $d'(x, y)$ is the disparity value of pixel (x, y) obtained by the algorithms and $d_0(x, y)$ is the accurate disparity of pixel (x, y) supplied by [70].

The accuracy of the disparity map obtained by four local methods including ours is shown in Table 4.1. The experimental results show that our method yields a more accurate disparity map comparing to other traditional local methods.

The good performances of our method on different image pairs also prove that our method can be used to obtain initial disparity map on different type of image (images with texture less region and occluded regions) accurately.

Table 4.1 Performance of proposed initial disparity acquisition algorithm

Methods	Images	Tsukuba	Teddy	Venus	Cones	Average
		<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>
Line segment[43]	% of the Incorrect matching pixels	25.41	17.62	16.64	21.36	20.25
Sum of Hamming Distance[42]		31.72	26.53	24.62	15.78	24.66
Normalized Census transformation[62]		21.37	14.21	11.78	9.21	19.34
Our method		17.23	17.12	10.73	6.34	12.86

The initial disparity map obtained in this section will be used in the subsequent process of the segment-based stereo matching algorithm (Sections 4.4 and 4.5). A disparity plane fitting method is introduced in the next section.

4.4 Disparity plane fitting

In this section we shall present the method to assign a disparity plane to each segment in the segmented reference image. Our method is based on the work presented by Tao et al. [27]. We shall use the initial disparity obtained in Section 4.3 to determine the disparity plane of each

segment in the image. All the pixels inside a segment will have the same disparity of this disparity plane which is modeled by an equation of a plane, given by:

$$d^k(x, y) = ax + by + c \quad (4.11)$$

The disparity plane is described by three coefficients a , b , and c . Here, (x, y) are the coordinates of pixel in the k^{th} segment in the image. $d^k(x, y)$ denotes the disparity plane of the k^{th} segment. As mentioned in Section 4.1, a segmented region should have a constant disparity value. Therefore, with the appropriate values of the coefficients a , b and c in Equation (4.11), the disparity of any pixel in the region is a constant. Note that if there are n segments in the reference image, there are n disparity planes in the reference image. This section presents how the three parameters of Equation (4.11) are determined.

If there are m pixels in the k^{th} segment in the image, then the matrices can be formed as:

$$A = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \dots & \dots & \dots \\ x_m & y_m & 1 \end{bmatrix} \quad (4.12)$$

$$B = \begin{bmatrix} d_1^k \\ d_2^k \\ \dots \\ d_m^k \end{bmatrix} \quad (4.13)$$

$$A[a, b, c]^T = B \quad (4.14)$$

where (x_i, y_i) are the coordinates of the i^{th} point inside segment k , m is the number of pixels in segment k and d_i^k is the corresponding disparity value at (x_i, y_i) that has been obtained in Section 4.3.

We use Least Squares Error method to solve for the plane parameters $(a, b$ and $c)$ by rewriting the matrix Equation (4.14) as:

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = (A^T A)^+ A^T B \quad (4.15)$$

where matrix $(A^T A)^+$ is the pseudo-inverse of $A^T A$ which can be computed using method presented in [74].

After applying the above mentioned procedure for all the segments of the segmented image determined in Section 4.2, each segment is now assigned a disparity plane with the appropriate values of a, b and c obtained.

At this juncture, we would have obtained the equations representing respective disparity planes of all the segments. The following section aims to refine the disparity planes or disparity values by filtering the outlier, and merging the neighboring segments with similar disparity value.

4.5 Refinement of the disparity plane

In this section we shall describe our method which is used to refine the disparity plane obtained in the last section. The plane fitting process assigns each segment a disparity plane defined by its parameters $[a, b, c]$. However, the result will be affected by the outliers inside the segments that should be filtered away. In addition, the connected segments with the same disparity plane or

disparity value should be merged. These two actions serve to refine the disparity plane obtained in Section 4.4.

4.5.1 Refining disparity plane by outlier filtering

After having obtained the left and right disparity maps (D_L and D_R) in Section 4.3, they are used to cluster the pixels in the image into

- Stable pixels.
- Unstable pixels.

The pixels are classified as Stable Pixels when they satisfy the criterion based on their similarity in disparity values (in D_L and D_R) are given by:

$$|d_l(x_l) - d_r(x_l - d_l(x_l))| < 1 \text{ pixel} \quad (4.16)$$

where $d_l(x_l)$ and $d_r(x_l - d_l(x_l))$ are the disparities of the pixel $P(x_l, y)$ and $P'(x_r, y)$ respectively. This criterion can be used to filter out the occluded pixel and areas with low texture details (unstable pixels) where disparity estimation tends to be unreliable.

The process is described in detail as follow:

If we assume that $P'(x_r, y)$ is a pixel in the right image, and it is the corresponding pixel of $P(x_l, y)$ which is in the left image. The disparity is:

$$d_l(x_l) = x_l - x_r \quad (4.17)$$

Then, we shall perform a cross-checking. By considering the pixel $P'(x_r, y)$ in the right image and its corresponding pixel in the left image $P''(x'_l, y)$

where

$$d_r(x_r) = d_r(x_l - d_l(x_l)) = x_r - x_l' \quad (4.18)$$

Now, we can use the results of Equations (4.16) (4.17) and (4.18) to determine whether the pixel x_l is stable. If it is so, it will be labeled as stable; otherwise, it will be labeled as unstable.

In the k^{th} segment of the segmented image, the number of stable pixels is denoted as n_{stable}^k and the number of unstable pixels is denoted as $n_{unstable}^k$.

The set of stable pixels is denoted as $S' = \{(x_i, y_i) \in S \mid |d_l(x_i, y_i) - d_r((x_i - d_l(x_i), y_i))| < 1\}$ (where S is the set of all pixels in segment k)

Then we update S' as follow:

1. If $\frac{n_{unstable}^k}{n} \leq \varepsilon$; where $n_{unstable}^k$ denotes the number of unstable pixels in segment k , n is the total number of pixels in segment k , and ε is a pre-defined threshold. The segment k is labeled as a stable segment. The stable segments are denoted as SEG_{stable} and if $\frac{n_{unstable}^k}{n} > \varepsilon$, it implies that there is insufficient information to provide reliable plane estimation. The segment k is labeled as an unstable segment. Unstable segments will not be used to update the disparity plane. Then S' is updated into $S'' = \{(x_i, y_i) \in S' \mid (x_i, y_i) \in SEG_{stable}\}$

2. If the initial disparity of pixel (x_i, y_i) in the k^{th} segment is d_i^k (obtained in Section 4.3), then the pixels in S'' that satisfy Equation (4.19) will remain in the set of stable pixels.

$$\left| d_i^k - (ax_i + by_i + c) \right| \leq t_{out} \quad (4.19)$$

where t_{out} is a threshold of outlier (normally set value around 1), d_i^k is the initial disparity of pixel (x_i, y_i) , $[a, b, c]^T$ is the disparity plane parameter obtain in Section 4.4.

Then, S'' is updated into $S''' = \{(x_i, y_i) \in S'' \mid \left| d_i^k - (ax_i + by_i + c) \right| \leq t_{out} \}$

Note that the pixels in S''' are the stable pixels after filter out the outliers and the plane fitting procedure given in Section 4.4 is executed using the stable pixel we obtain in this section. Thus the disparity plane is refined and the procedure is shown in Figure 4.9.

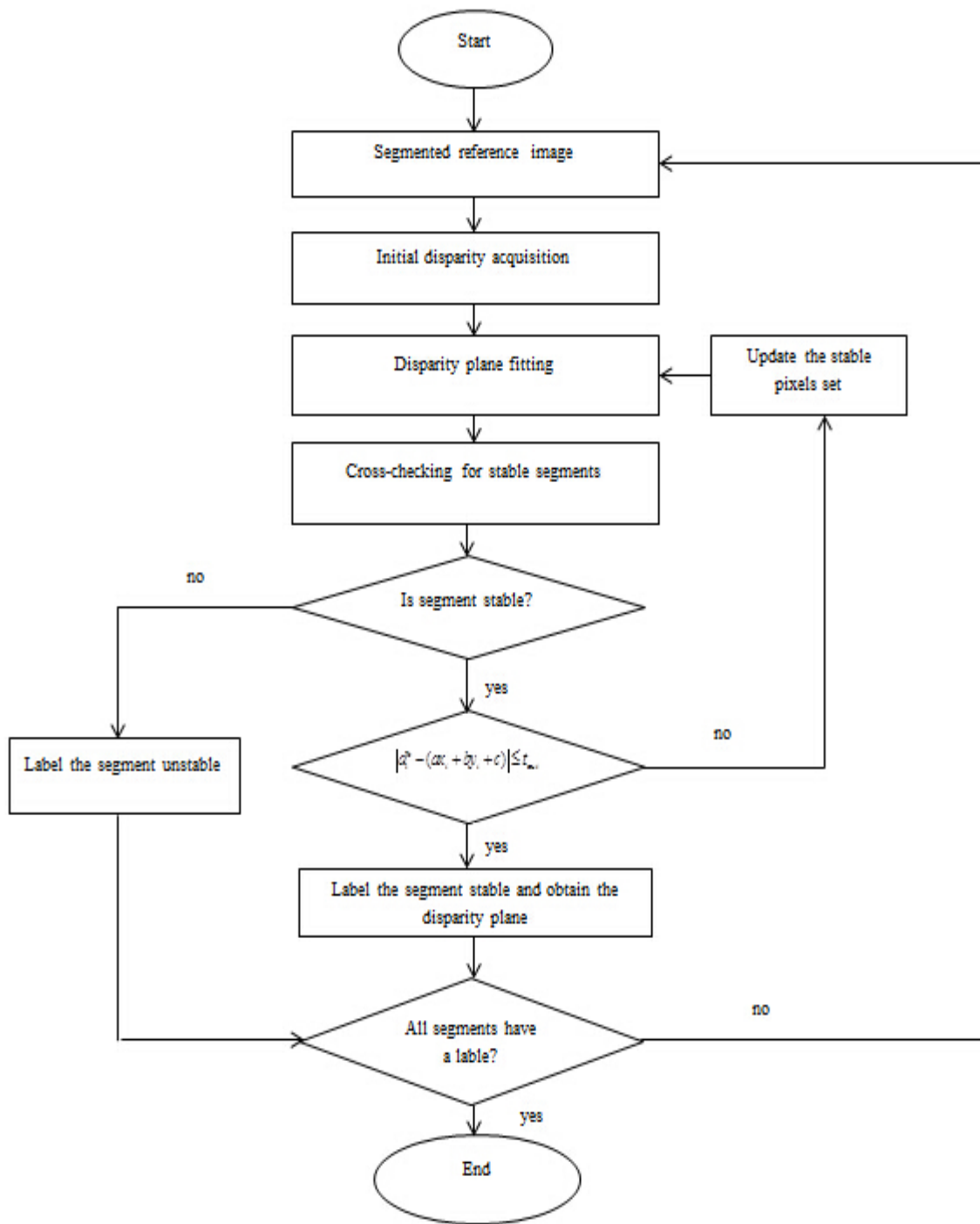


Figure 4.9 Flow chart of refinement of the disparity plane by Outlier filtering.

4.5.2 Refining disparity plane by merging connected segments with same disparity

After plane fitting and outlier filtering, we would have obtained a segmented image where all the segments have been assigned to a particular disparity plane. However some of the segments are assigned “bad” disparity planes due to the occluded pixels and texture-less segments in which there are not enough stable pixels to obtain the disparity planes parameters. Therefore we shall refine the disparity planes in this section in two steps: 1) merge the connected segments that share the same disparity plane and 2) assign a disparity plane to those unstable segments (see Section 4.5.1).

In our method, we denote $C = \{d_i \mid i = 1 \text{ to } n\}$, where n is the number of segments} as the set of all disparity planes we have obtained in Section 4.4, and we substitute each d_i into one chosen segment to re-compute the matching cost of all pixels inside this segment. The disparity plane which minimizes this cost as the refined disparity plane for this segment is then chosen. We repeat the same procedures for all segments in the image to finish the refinement.

The refinement of the disparity plane should be operated through the steps below:

Algorithm 4.2: Algorithm for refining the disparity plane:

1. Forming disparity plane set: Collect all the disparity planes obtained from plane fitting and save them in a cluster which we called possible disparity plane set, $C = \{ d_i \mid i = 1 \text{ to } n \text{ where } n \text{ is the number of segments} \}$
2. Let $I = \{ S_i \}, i = 1 \text{ to } n$, where n is the number of segments. I represents the whole image and S denotes the segments inside I
3. Let $S_i = \{ P_k^i \}, k = 1 \text{ to } m_i$ (where m_i is the number of the pixels in S_i , P_k^i denotes the k^{th} pixel in S_i)
4. Updating disparity plane for segments:

For $j = 1 \text{ to } n$

For $i = 1 \text{ to } n$, calculate the matching cost using:

$$E_i = \sum_{k=1}^{m_j} c(x_k^j, y_k^j, d_i)$$

$$E_j = \min \{ E_i \}$$

$$d_j = \arg \min (E_j)$$

End

End

5. Recovering disparity map:

$D = \{ d_j \}$ for $j = 1 \text{ to } n$, where n is the number of segments.

After the steps above, we would have obtained the newly updated disparity plane for each segment, the connected segments which have the same disparity will be merged into one segment and the unstable segments are assigned a more accurate disparity plane.

4.6 Formulation of energy function

To obtain a more accurate disparity map, we formulate an energy function of disparity planes. This energy function is used to measure the energy cost of all the disparity planes in the image. According to the segmentation result we have obtained in Section 4.2, the image is segmented into segments with homogenous colors (image structure is shown in Figure 4.10). According to the discussions in Section 4.4, each segment is assigned a disparity plane.

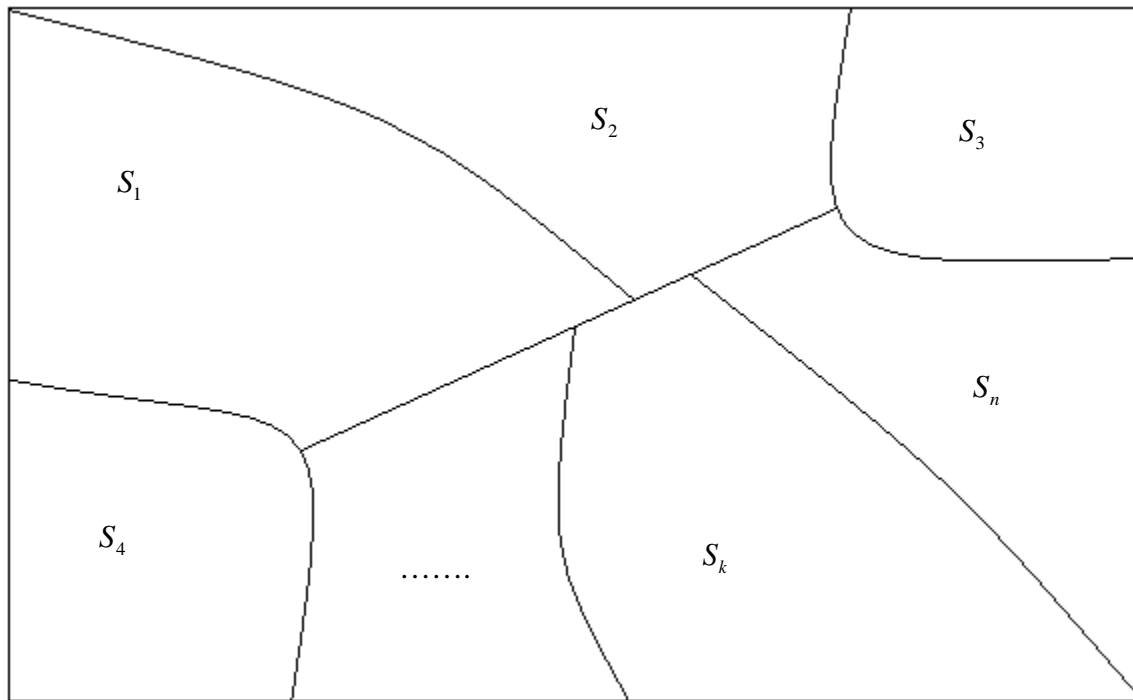


Figure 4.10 Structure of the segmented image.

Therefore, the total energy function is the sum of all the energy functions of all the segments which is expressed as

$$E_{all} = \sum_{k=1}^n E_k \quad (4.20)$$

Then the energy function for the k^{th} segment will be formulated as:

$$E_k = E_{k_data} + E_{k_smoothness} + E_{k_unstable} \quad (4.21)$$

We shall describe the three energy terms separately as follows;

$$E_{k_data} = w_d \sum_{(x,y) \in S_k} c(x, y, d_k) \quad (4.22)$$

1. E_{k_data} represents the cost due to the differences in intensity between two corresponding pixels in a stereo image pair, It embodies the data constraint which measures the similarity of the two corresponding points each with a certain disparity value. Let us consider a point $P_l(x, y)$ in the k^{th} segment in the left image, we evaluate the difference $c(x, y, d_k)$ (see Equation (4.3) in intensity between P_l and the corresponding point $P_r(x + d_k, y)$ where d_k is the disparity of the k^{th} segment (S_k shown in Figure 4.9). We repeat this procedure with all the pixels in the segment, and sum up the results. The final summation of all these values is the term E_{k_data} . w_d is a predefined weight factor which controls the strength of the data term in energy function.
2. $E_{k_smoothness}$ measures the difference in disparity values between two neighboring pixels. This term represents the penalty to those connected pixels with disparity discontinuity. The term embodies the smoothness constraint and encourages coherence in regions of similar color. It is expressed as :

$$E_{k_smoothness} = \sum_{(x,y \in k)} p_{sm} \quad (4.23)$$

where

$$P_{sm} = \begin{cases} \sum_{(x',y') \in N(x,y)} \frac{\lambda}{\sqrt{(x-x')^2 + (y-y')^2}} \exp(\delta[I(x,y) - I(x',y')]^2), & \text{if } (d(x,y) - d(x',y') > 1) \\ 0, & \text{otherwise} \end{cases}$$

In P_{sm} , $d(x,y)$ and $d(x',y')$ are the disparity value of (x,y) and (x',y') respectively, and (x',y') is a neighboring pixel of (x,y) ; $I(x,y)$ and $I(x',y')$ are the intensity of the two pixels, λ is a predefined constant, and δ is a constant defined by Equation (4.24), where $\langle \bullet \rangle$ means the expectation of the disparities of all the pixels in the current support region presented in [89].

$$\delta = (2 \langle I(x,y) - I(x',y') \rangle)^{-1} \quad (4.24)$$

Note that the smoothness term involves penalty to the disparity discontinuity in pixel domain. In fact, based on the assumption that we have made in Section 4.2, the pixels inside a segment should have the same disparity value. Thus this term would mainly involve penalty to the pixels which are in the close neighborhood of the segment boundaries whose neighboring pixels may lie in other connected segments. This term is then affected not only by the disparity of the segment k but also by the disparities of its neighboring segments.

$$3. \quad E_{k_unstable} = w_{unstable} \left[\frac{\sum_{(x,y) \in S_k} P_{un}}{N_k} \right] \quad (4.25)$$

$E_{k_unstable}$ takes care of the unstable pixels (the occluded pixel and the texture-less pixel describe in Section 4.5.1) inside a segment. The smaller the number of unstable pixels, the smaller will be the value of this energy term. Let P_{un} be the unstable index and N_k total number of the pixels in segment k ; when the pixel is unstable $P_{un} = 1$ otherwise $P_{un} = 0$. Obviously, the term $\sum_{(x,y) \in S_k} P_{un}$ is the total number of unstable pixel in the segment k (see Section 4.5.1). $w_{unstable}$ is a parameter that controls the strength of this unstable term.

Combining the three energy terms, the final expression of the Energy Function is:

$$E_{all} = \sum_{k=1}^n \left(w_d \sum_{(x,y) \in S_k} c(x, y, d_k) + \sum_{(x,y) \in S_k} p_{sm} + w_{unstable} \left[\sum_{(x,y) \in S_k} P_{un}(x, y) / N_k \right] \right) \quad (4.26)$$

With the energy function formulated, we aim to obtain the optimal disparity map $D_{op} = \{d_{k_op}\}$ ($k = 1$ to n) that minimizes E_{all} . In the next section the belief propagation method is used to achieve this goal.

4.7 Belief propagation method

The purpose of this section is to obtain an accurate disparity map $D_{op} = \{d_{k_op}\}$, $k = 1$ to n (n is the number of segments in the whole image) using the belief propagation (BP) optimization method. In this method we use Markov network to optimize the energy function (Equation (2.7)) by modifying the disparity values of all the segments. Figure 4.11 shows a Markov network. The nodes X_s , ($s = 1$ to n) where n is the total number of the nodes (number of segments determined in Section 4.2) and x_s , ($s = 1$ to n) are the states of node X_s . In our work, the value of x_s is the disparity of the s^{th} segment (denoted as d_s)

$$E(x, y) = \sum_s \phi_s(x_s, y_s) + \sum_s \sum_{t \in N(s)} \psi_{s,t}(x_s, x_t) \quad (4.27)$$

In Equation (4.27), y_s denotes the state of observed nodes Y_s connected with each hidden node X_s , $\phi_s(\bullet)$ is the local evidence for node s which is self-updated with time, and $\psi_{s,t}(\bullet)$ is the compatibility matrix between node s and t which is updated by the connected nodes with time.

An optimal state x_{s_op} ($s = 1$ to n) which minimize energy function (expressed in Equation (4.27)) is obtained using belief propagation method.

The method is proposed in [47] and we briefly describe it as follow:

In Figure 4.1, by denoting $X = \{x_s\}$ and $Y = \{y_s\}$, the posterior $P(X / Y)$ can be factorized as:

$$P(X | Y) \propto \prod_s \phi_s(x_s, y_s) \prod_s \prod_{t \in N(s)} \psi_{st}(x_s, x_t) \quad (4.28)$$

From Equation (4.27) and (4.28), we note that to find the optimal states to minimize Equation (4.27) is the same as finding optimal states to maximize Equation (4.28) because of the negative

logarithmic property reported in [25]. Thus, the problem of minimizing the energy function (Equation 4.27) is modified into the maximizing the posterior probability $P(X / Y)$ which is the theoretical basis of belief propagation.

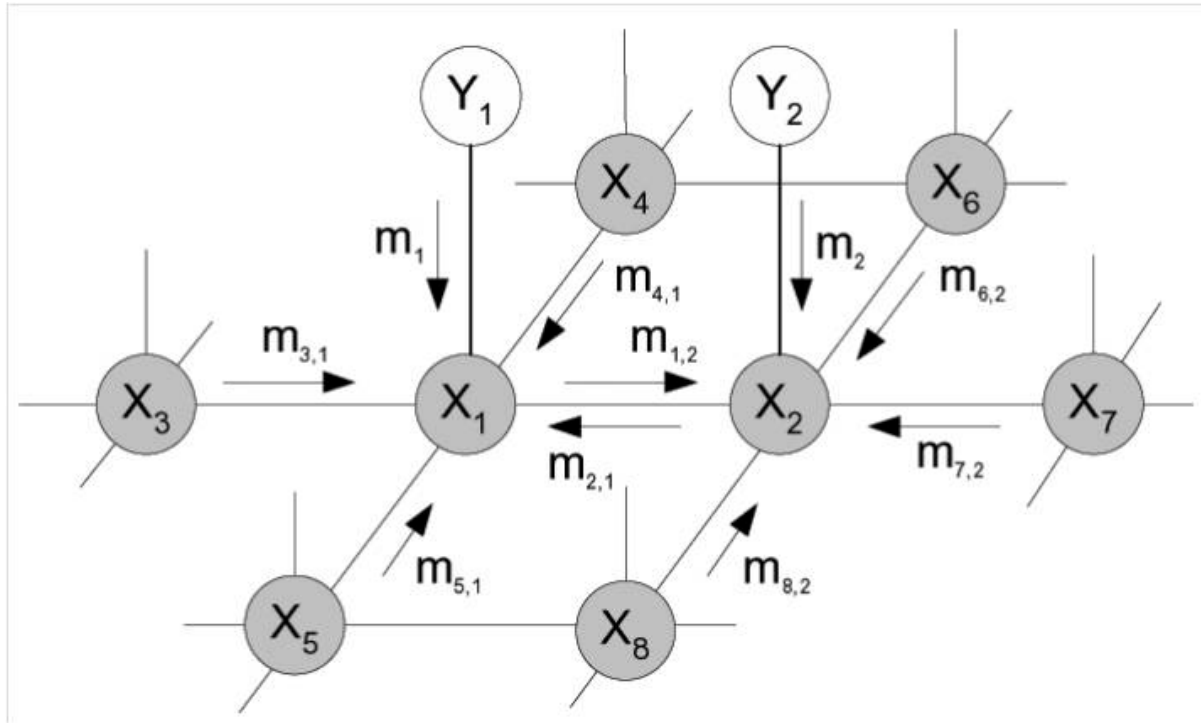


Figure 4.11 Belief propagation Optimization.

In the Markov Network shown in Figure 4.11, the belief propagation proceeds as follow:

$m_{st}(x_t)$ denotes the message that node x_s sends to node x_t ; $m_s(x_s, y_s)$ represents the message that node y_s sends to node x_s and $b_s(x_s)$ is the belief of node x_s . If the number of discrete states of x_s is L , $m_{st}(x_t)$, $b_s(x_s)$ and $m_s(x_s, y_s)$ are all vectors with L elements. Note that $L = (d_{\max} - d_{\min} + 1)$ (where d_{\max} and d_{\min} are the maximum and minimum disparity among all the segments in image and both values are truncated to the nearest integers). The task is to find the

optimal state x_s , (where $s=1$ to n) to maximize the posterior probability shown in Equation (4.28), the algorithm is shown below:

Algorithm 4.3: Algorithm for belief propagation:

1. Initializing messages: Initialize all messages $m_{st}(x_t)$ as uniform distributions
2. Update message $m_{st}(x_t)$ iteratively for $i=1$ to T

$$m_{st}^{i+1}(x_t) = k \max_{x_s} \psi_{st}(x_s, x_t) m_s^i(x_s, y_s) \prod_{x_k \in N(x_s) \setminus x_t} m_{ks}^i(x_s) \quad (4.29)$$

where $m_s^i(x_s, y_s) = \phi_s^i(x_s, y_s)$ and k is the normalization constant

3. Compute

$$b_s(x_s) = k m_s(x_s) \prod_{x_k \in N(x_s)} m_{ks} x_s \quad (4.30)$$

4. The optimal state x_s is computed as

$$x_s = \arg \max_{x_k} b(x_k) \quad (4.31)$$

In our proposed algorithm, we are calculating the disparity for each segment and the segments are taken as the nodes in Markov network; the energy cost in segment s which is denoted as E_s is the belief of node s ; the disparity d_{s_op} is the optimal disparity which minimizes the energy function E_s (denoted as x_s in Markov Network) and the disparity map $D_{op} = \{d_{s_op}\}$ ($s=1$ to

n) (denoted as the set of $\{x_s\}_{s=1 \text{ to } n}$) is the optimized disparity map. In the energy function, the data term E_{s_data} and the unstable term $E_{s_unstable}$ can be viewed as the local evidence ($m_s(x_s, y_s)$ in belief propagation method) which are self-updated and are not affected by the disparities of neighboring segments. On the other hand, the smoothness term can be used to determine the compatibility matrix ($\psi_{s,t}(x_s, x_t)$) which is updated using the disparities of neighbor segments. Therefore we define

$$\psi_{s,t}(x_s, x_t) = \exp(-E_{s_smoothness}) \quad (4.32)$$

$$\phi(x_s, y_s) = \exp(-(E_{s_data} + E_{s_unstable})) \quad (4.33)$$

From Equation (4.28), (4.32) and (4.33) we obtain:

$$P(X|Y) \propto \prod_s \phi_s(x_s, y_s) \prod_s \prod_{s \in N(s)} \psi_{st}(x_s, x_t) = \exp(-E_{s_all}) \quad (4.34)$$

From Equation 3.34, we note that to minimize our energy function is equal to maximize the posterior in Markov network and we could use belief propagation method to obtain accurate disparity map $D_{op} = \{d_{s_op}\} = \{x_{s_op}\}$.

At this juncture, we have obtained the disparity for each segment in the image and the final disparity map is then obtained. Figure 4.12 shows the disparity map obtained by initial disparity acquisition method (Section 4.3) and final disparity map using our proposed algorithm.

Figure 4.12 shows the experimental result using the standard rectified image “Arts”. Figure 4.12 (a) is the reference image; (b) is the accurate disparity map supplied by Middlebury database [70] (c) is the initial disparity map obtained using the method proposed in Section 4.3; and (d) is the final disparity map obtained using our proposed algorithm. We can observe that our BP

algorithm can produce a disparity map which is very close to that provided by the Middlebury database [70]. In addition, by comparing the initial disparity map shown in Figure 4.12(c) with the final disparity map shown in Figure 4.12 (d), we can see the significant improvement of the accuracy of the disparity map using BP. More experimental results and analysis of the performance will be presented and discussed in Section 5.2.



Figure 4.12 Experimental results of Arts. (a) reference image; (b) disparity map provided in Middlebury database; (c) initial disparity map; (d) disparity map obtained by the proposed algorithm.

4.8 Depth recovery using disparity map

In this section we use the intrinsic and extrinsic parameters obtained in Chapter 3 and the optimized disparity map obtained in Chapter 4 to recover the 3-D depths of all the pixels in the captured scene. The structure of a rectified stereovision system is shown in Figure 4.13.

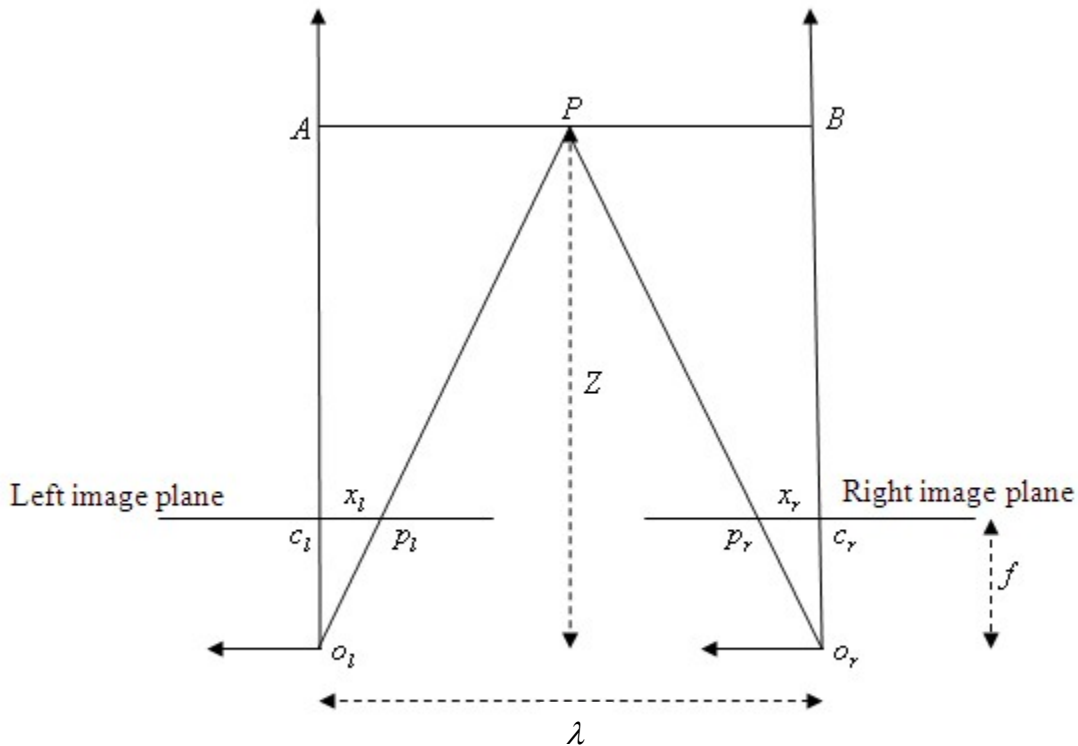


Figure 4.13 Depth recovery based on disparity.

In Figure 4.13 P is a point in 3-D scene, the left and right image planes are coplanar. c_l and c_r are the centers of the left and right image planes, respectively; o_l and o_r are the optical centers of the left and right cameras, respectively, p_l and p_r are the projective points of point P on the left and right image planes respectively. x_l is the x coordinate in the left image plane respect to c_l , x_r is the x coordinate in the right image plane with respect to c_r . λ is the distance between the two optical centers (baseline).

According to the geometrical relationship in Figure 4.13, we obtain:

$$\begin{cases} \frac{x_l}{f} = \frac{|PA|}{Z} \\ \frac{-x_r}{f} = \frac{|PB|}{Z} \end{cases} \quad (4.35)$$

from which we obtain the following relationship.

$$\lambda = |PA| + |PB| \quad (4.36)$$

$$\frac{x_l - x_r}{f} = \frac{\lambda}{Z} \quad (4.37)$$

Note that $d = x_l - x_r$ then,

$$Z = \frac{\lambda f}{d} \quad (4.38)$$

In Equation (4.38), f is the focal length of the camera which can be obtained from its technical specification. λ is the baseline which can be calculated using Equation (3.18) in Chapter 3, d can be computed using the stereo matching algorithm proposed in this chapter. As all the parameters for calculating depth are known, the depth of point P can be recovered accordingly. Using the same method, the depths for all the points in 3-D scene can be recovered. The results are shown in Chapter 5, Section 5.2

4.9 Summary

In this chapter, a segment-based stereo matching algorithm was proposed. The algorithm consists of the following steps: color based segmentation, initial disparity acquisition, disparity plane fitting, disparity plane refining, and belief propagation based optimization of the energy function to obtain an accurate disparity map. In color based segmentation, a self-organizing map segmentation algorithm was applied to decompose the reference image into segments with homogenous colors. A local stereo matching algorithm was then proposed to obtain the initial disparities for pixels inside the image. The initial disparities were used as the initial data for the remaining steps. A plane fitting process was executed subsequently on the segmented image to assign each segment a disparity plane which can be used to compute disparity for each pixel in that segment. A refinement of the disparity plane was done by filtering outliers and merging the connected segments that have a same disparity plane. In order to obtain a more accurate disparity map, an energy function which considers likelihood, smoothness and penalty of occlusion is formulated and the belief propagation optimization method was applied to obtain the optimal disparity map which minimizes the energy function. Finally we obtain an accurate segment-based disparity map. The experimental results demonstrate the accuracy of our proposed segment based stereo matching algorithm. Additional experimental results and the associated discussion will be presented in the next chapter.

Chapter 5 Experiment Results and Analysis

In this chapter, experimental results on the performance of our proposed algorithms are shown and the advantages and disadvantages of the algorithms are discussed. The majority of the stereo image pairs used in our experimental studies are captured by the single-lens bi-prism stereovision system in our laboratory which is shown in Figure 5.1. To test our algorithms and the associated methods, we have also used the standard stereo image pairs taken from a standard stereo image source, namely, the Middlebury database [70] available on internet. In this database the sample image pairs are carefully prepared, and their properties, such as the disparity values between the corresponding pixels are provided. This information is known as Ground Truth Data.

5.1 Experiment setup

Figure 5.1 shows the setup of our experiments. The system components include the followings:

- The camera used is a PULNIX TM-6CN CCD camera;
- The camera is mounted on two rails on a horizontal plane, and the rails are perpendicular to each other. The rails and the associated hardware are custom made. The camera is allowed to move in the two perpendicular directions;
- The laser sensor located next to the CCD camera is used to measure the physical distance between the camera image plane and the objects;
- Bi-prisms with angles ($\mu = 6.4^\circ, 10^\circ, 21.6^\circ$) are used in the experiments;

- The image pairs in the experiments will be processed by the procedure presented in Chapter 3 and Chapter 4. The left image will be taken as the reference image. The algorithm is coded and executed in Matlab.

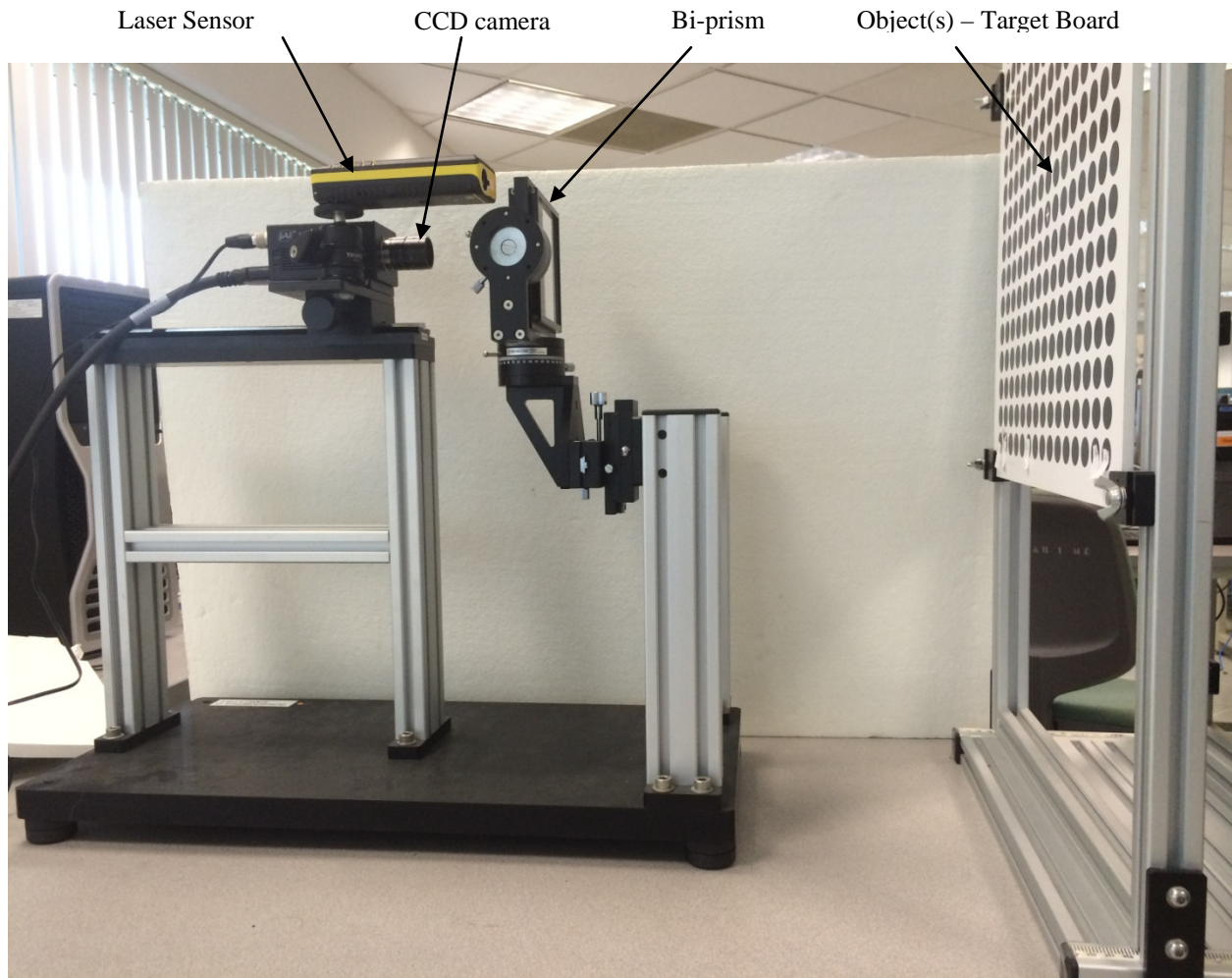


Figure 5.1 Single-lens bi-prism stereovision system.

5.2 Experimental results and analysis

The experimental results are shown in three parts. In Section 5.2.1, the experiments were carried out using stereo image pairs chosen from Middlebury database [70]. Images in this standard database are used to evaluate the stereo matching algorithms presented in this thesis. This part is to show the accuracy of the depth obtained by our proposed segment-based stereo matching algorithm.

The second part of our experimental study is presented in Section 5.2.2. Here, we test our proposed stereo matching algorithm using image pairs captured by the single-lens bi-prism stereovision system set up in our laboratory. This is to test the effectiveness of the algorithm when applied to non-rectified image pairs, which are encountered in most practical applications.

Section 5.2.3 presents the last part of our experimental studies, in which we evaluate the performance of our proposed rectification algorithm by comparing the depth recovery result with other algorithms.

5.2.1 Experimental results based on the image pairs taken from Middlebury database

The purpose of conducting this series of experiments is to evaluate the accuracy of our algorithm using standard stereo image pairs. In this section, we first evaluate the performance of the proposed segment-based stereo algorithm using standard image pair chosen from Middlebury database [70]. In our experiment, we use the image pairs “Tsukuba” ,“Venus”, “Teddy”, “Cones”, “Art” and “Computer”, the values of the following parameters used in the experiments are fixed: $w_{data} = 0.6$, $\varepsilon = 0.9$, $w_{unstable} = 0.2$ and $\lambda = 50$ (that has been shown to be a versatile setting for a wide variety of images in [104])

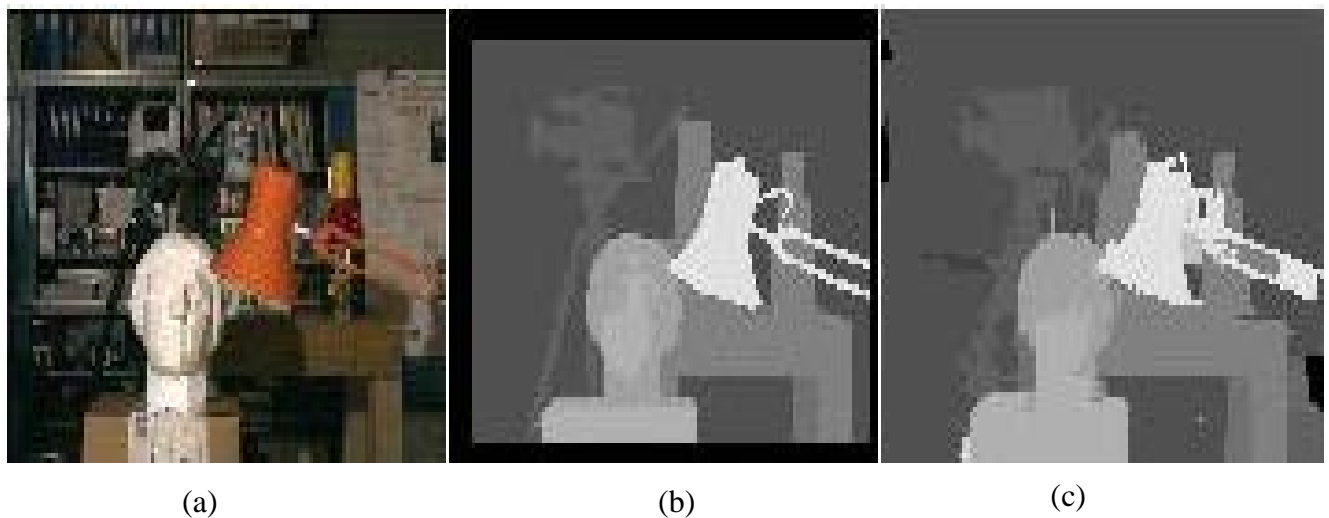


Figure 5.2 Experimental results of Tsukuba. (a) Reference image; (b) Disparity map provided in Ground-Truth Database [70]; (c) Disparity map obtained by our proposed algorithm.

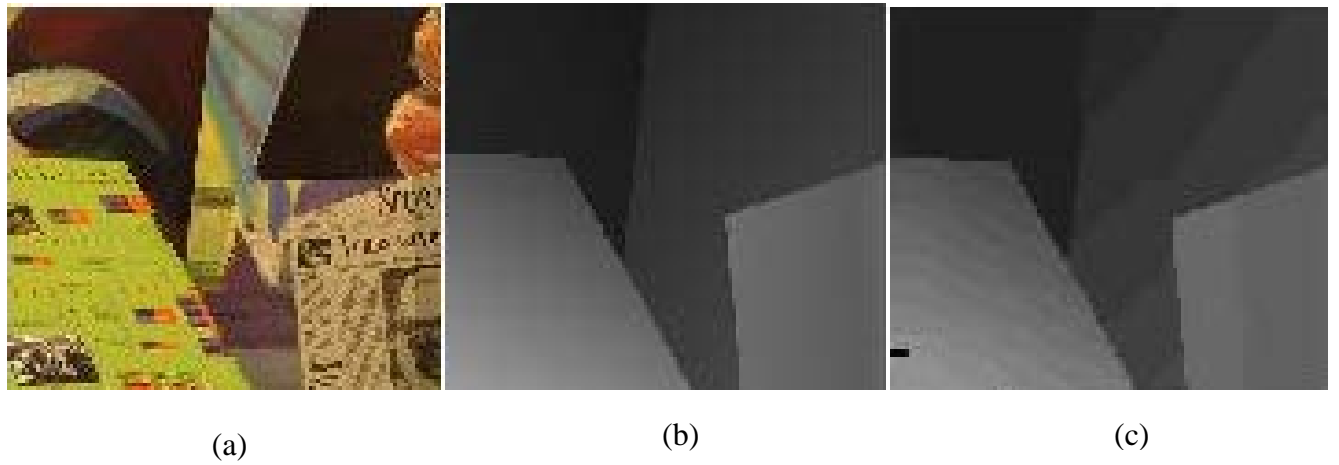


Figure 5.3 Experimental results of Venus. (a) Reference image; (b) Disparity map provided in Ground-Truth Database [70]; (c) Disparity map obtained by our proposed algorithm.

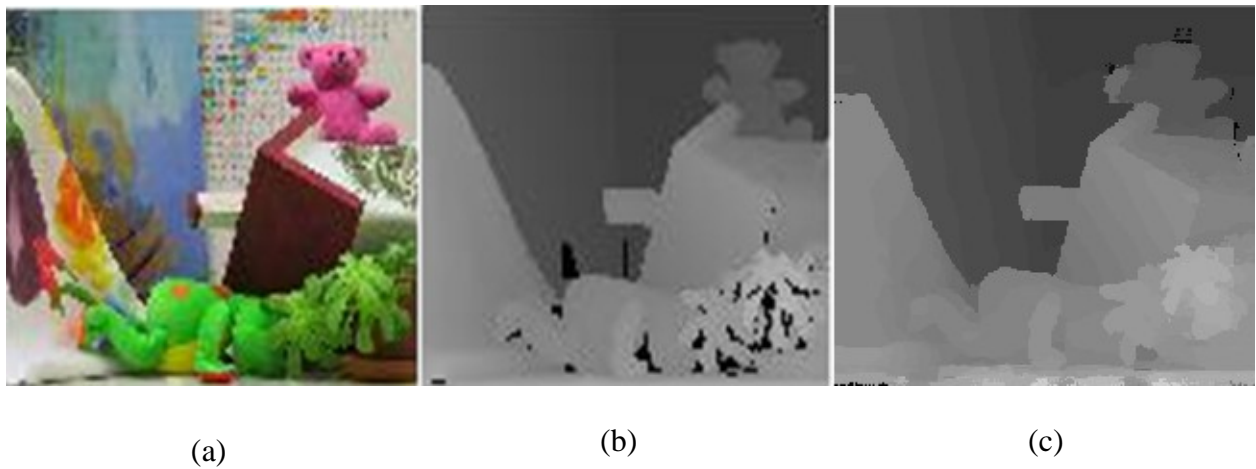


Figure 5.4 Experimental results of Teddy. (a) Reference image; (b) Disparity map provided in Ground-Truth Database [70]; (c) Disparity map obtained by our proposed algorithm.

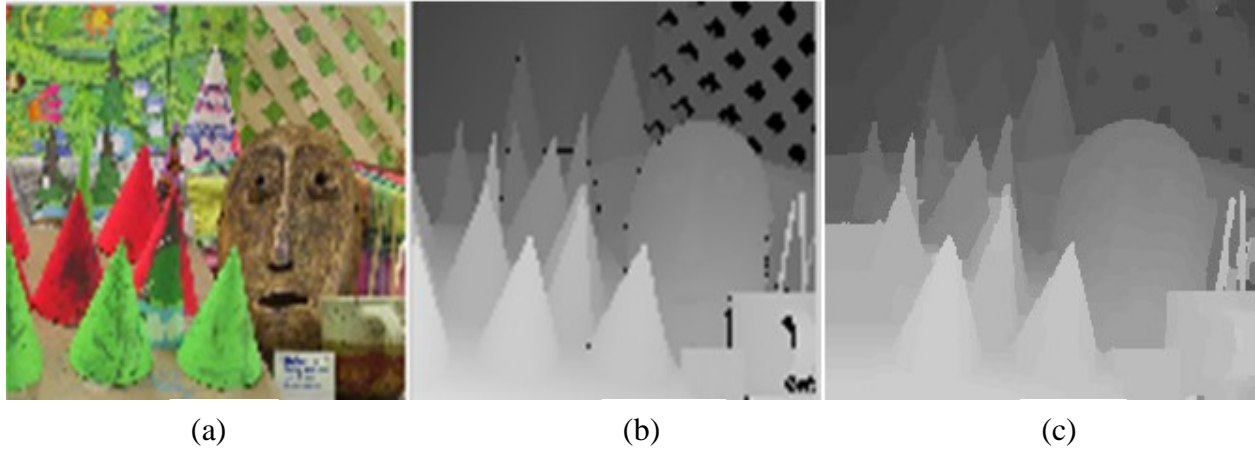


Figure 5.5 Experimental results of Cones. (a) Reference image; (b) disparity map provided in Ground-Truth Database [70]; (c) disparity map obtained by our proposed algorithm.

Figures 5.2-5.5 are the experimental results to show the performance of the proposed algorithm by comparing the disparity map of the image pairs “Tsukuba”, “Venus”, ”Teddy” and ”Cones” obtained by our algorithm with the Groud Truth data provided by [70]

Normally, in stereo matching research, the way to test the accuracy of the stereo matching algorithm is to compare the matching error defined by Equation (5.1) below. We compare the number of incorrectly matched pixels in the non-occluded region, whole image and regions near depth discontinuity. Note that an incorrect match is defined as, at a given pixel, the value of the disparity differs from that in the Ground Truth Database which is provided by the Middlebury database.

$$B = (n_{ic} / N) \times 100\% \quad (5.1)$$

where n_{ic} is the number of incorrect matches, and N is the total number of pixel in the image.

The three errors below are evaluated and compared:

- B_{Non} : Percentage of the incorrect matching points evaluated in the non-occluded segments;
- B_{all} : Percentage of the incorrect matching points evaluated over the whole image;
- B_{disc} : Percentage of the incorrect matching points evaluated in regions near depth discontinuities.

Table 5.1 Performance of different algorithms based on the Middlebury Stereo Evaluations

Algorithms	Avg. Rank	Tsukuba (Groud Truth)			Venus (Groud Truth)			Teddy (Groud Truth)			Cones (Groud Truth)		
		<i>nonocc</i>	<i>all</i>	<i>disc</i>	<i>nonocc</i>	<i>all</i>	<i>disc</i>	<i>nonocc</i>	<i>all</i>	<i>disc</i>	<i>nonocc</i>	<i>all</i>	<i>disc</i>
ADCensus[87]	13.0	1.07	1.48	5.73	0.09	0.25	1.15	4.10	6.22	10.9	2.42	7.25	6.95
RDF[88]	22.2	0.97	1.39	5.00	0.21	0.38	1.89	4.84	9.94	12.6	2.53	7.69	7.38
DoubleBP[89]	23.3	0.88	1.29	4.76	0.13	0.45	1.87	3.53	8.30	9.63	2.90	8.78	7.79
Our method	25.4	1.95	2.24	8.59	0.13	0.25	1.62	2.17	3.68	7.02	2.23	6.55	6.17
SurfaceST[90]	30.1	1.28	1.65	6.78	0.19	0.28	2.61	3.12	5.10	8.65	2.89	7.95	8.26
Wapmat [91]	34.0	1.16	1.35	6.04	0.18	0.24	2.44	5.02	9.30	13.0	3.49	8.47	9.01
LM3C[92]	46.8	2.10	2.44	8.01	0.12	0.39	1.23	5.46	10.9	14.9	2.12	7.59	6.14
HistoAgg[93]	50.5	1.93	2.30	6.39	0.16	0.46	2.22	5.88	11.3	14.7	2.41	7.78	6.89
PMhuber[94]	51.5	3.49	4.09	9.12	0.22	0.43	2.50	3.38	5.56	10.7	2.15	6.69	6.40
SSMP[95]	63.9	1.60	1.97	6.44	0.20	0.38	2.51	6.15	11.5	15.8	2.60	7.92	7.48

Note: *nonocc* represents Non-Occluded Regions; *all* represents over the whole image and *disc* represents regions near discontinuities.

Table 5.1 shows the experimental results based on the standard image pairs chosen from the Middlebury database [70]. The performance based on the four standard image pairs is listed. In each image, the first column is the percentage of incorrect matching points evaluated in non-occluded region, the 2nd and 3rd columns are the percentage of incorrect matching points evaluated over the whole image and regions near depth discontinuity, respectively. The performance of our propose algorithm is good; and it is ranked 25.4 amongst the top 100 Stereo matching algorithms in the Middlebury Database [70]. It is also very efficient in the occluded regions and regions near depth discontinuities as shown in Table 5.1. The incorrect matching pixels are less than other algorithms in our experiments. The performance in the region with discontinuities is better than some algorithm with the top rank (B_{disc} in the “cone” image and “Teddy” image).

The performance of our algorithms is good because, (1) the disparity refining process (Section 4.5) helps to filter out the outliers that contain unreliable disparity results. (2) The belief propagation optimization (Section 4.7) propagates the disparity information into the regions which are occluded and texture-less, which would have been labeled unstable in Section 4.5). (3) The embodying of smoothness constraint encourages coherence in regions of similar colors

We have also conducted experiments to show the performance of our method in determining the final disparity map. In one such experiment, we use the stereo image pair, “Books” available in the Middlebury standard database [70], and the result is shown in Figure 5.6 below. Figure 5.6 (c) shows the initial disparity map obtained using our method described in Section 4.3. Figure 5.6 (d) is the final disparity map, which has undergone the refinement process proposed in this thesis.

The refinement process includes the following steps: (1) our proposed algorithm presented in Section 4.5.1 has filtered out the outliers; (2) and that presented in Section 4.5.2 has merged those small segments with the same disparity plane; (3) next, the resulting image from the two above mentioned step is refined with the help of the optimization of the energy function created for the resulting image (presented in Sections 4.6 and 4.7). Finally, we obtain a more accurate disparity map which is comparable with the ground true data shown in Figure 5.6 (b). Other experiments are executed to compare the initial disparity map obtained in Section 4.5.3 with the final disparity map obtained using our proposed stereo matching algorithm. The result is shown in Figure 5.6.

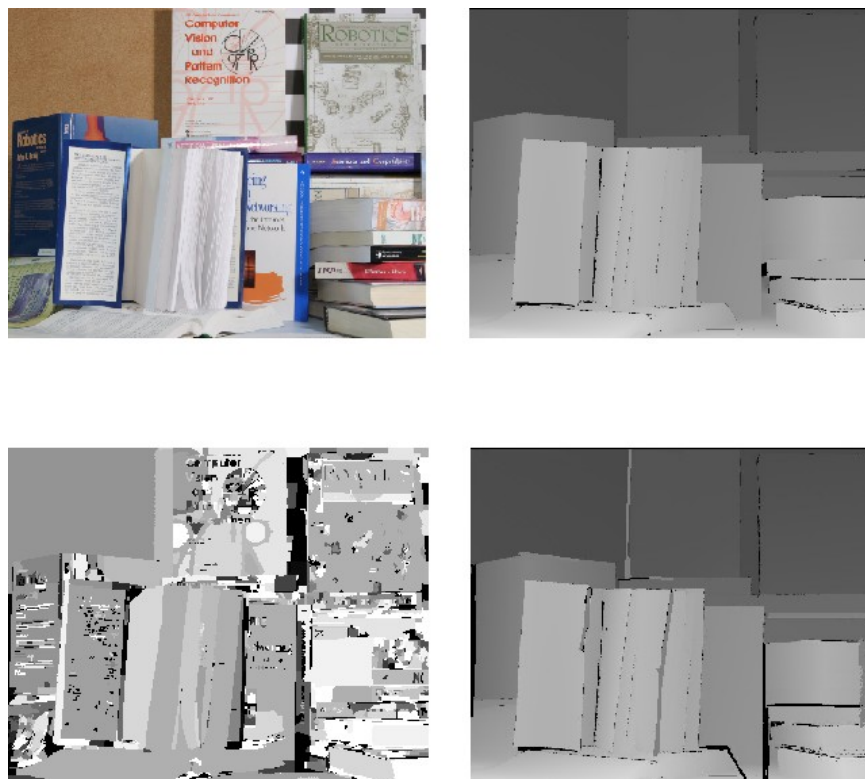


Figure 5.6 Experimental results of image “Books”. (a) Reference image; (b) disparity map provided in Ground-Truth Database; (c) initial disparity map and (d) disparity map obtained by our algorithm.

5.2.2 Experimental results using image pairs captured by single-lens bi-prism system

In this section, the experiments are conducted using the stereo image pairs captured by our single-lens bi-prism stereovision system. Each of the image pairs is first rectified using our algorithm in Chapter 3. Thence, the segment-based stereo matching algorithm proposed in Chapter 4 is then utilized to obtain the final disparity map. Figures 5.7 to 5.12 show the results of the experiments.

The single-lens prism based stereovision used in capturing the image pairs has the following characteristics:

- the focal length of the CCD camera is 25mm;
- the pixel size is 0.00465mm;
- the resolution is 1024 x 768 pixels;
- the prism angle $\mu = 10^\circ$;
- the refractive index $n = 1.48$.

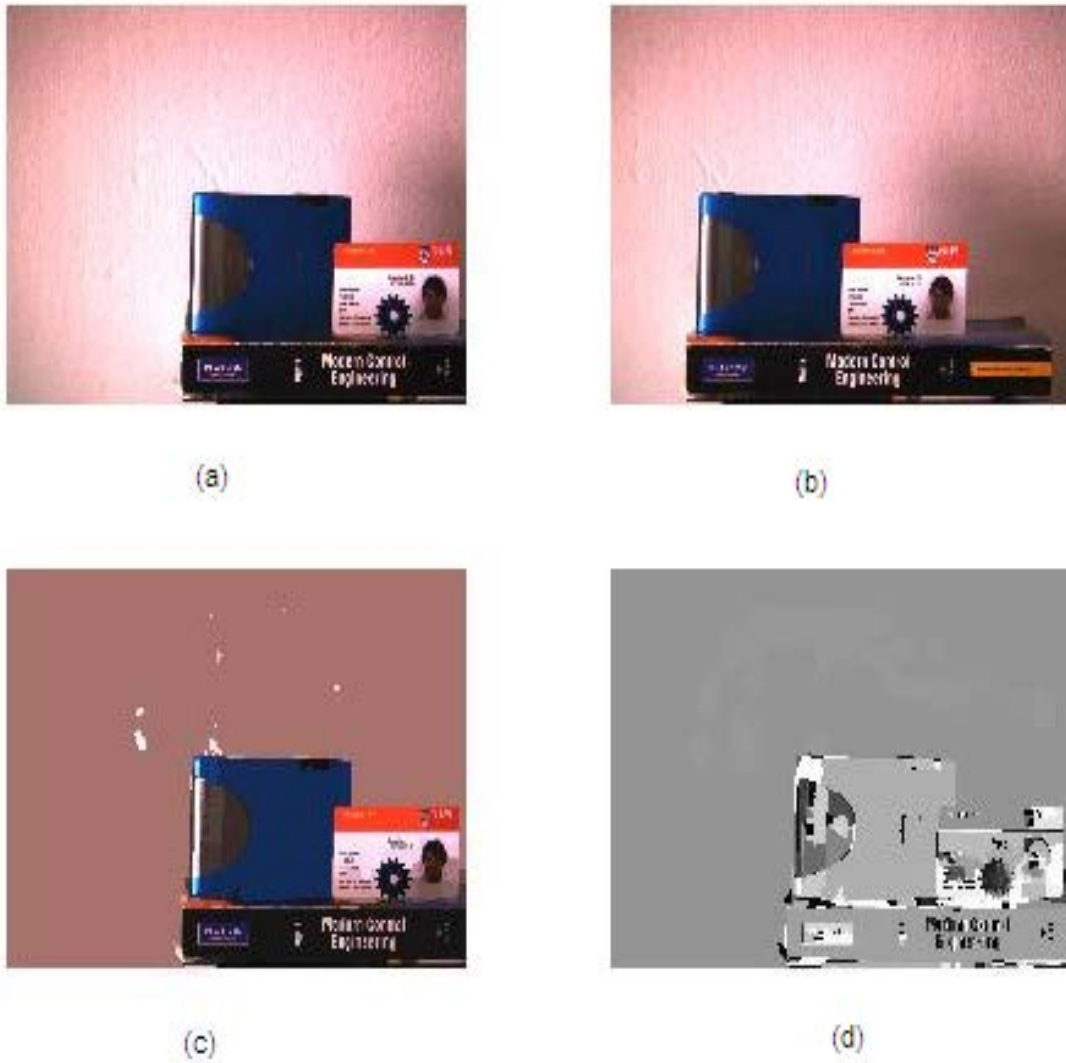


Figure 5.7 Result of Image pair 1 captured by single-lens bi-prism system: (a) Rectified left image, (b) rectified right image, (c) segmentation result of rectified left image and (d) final disparity map.

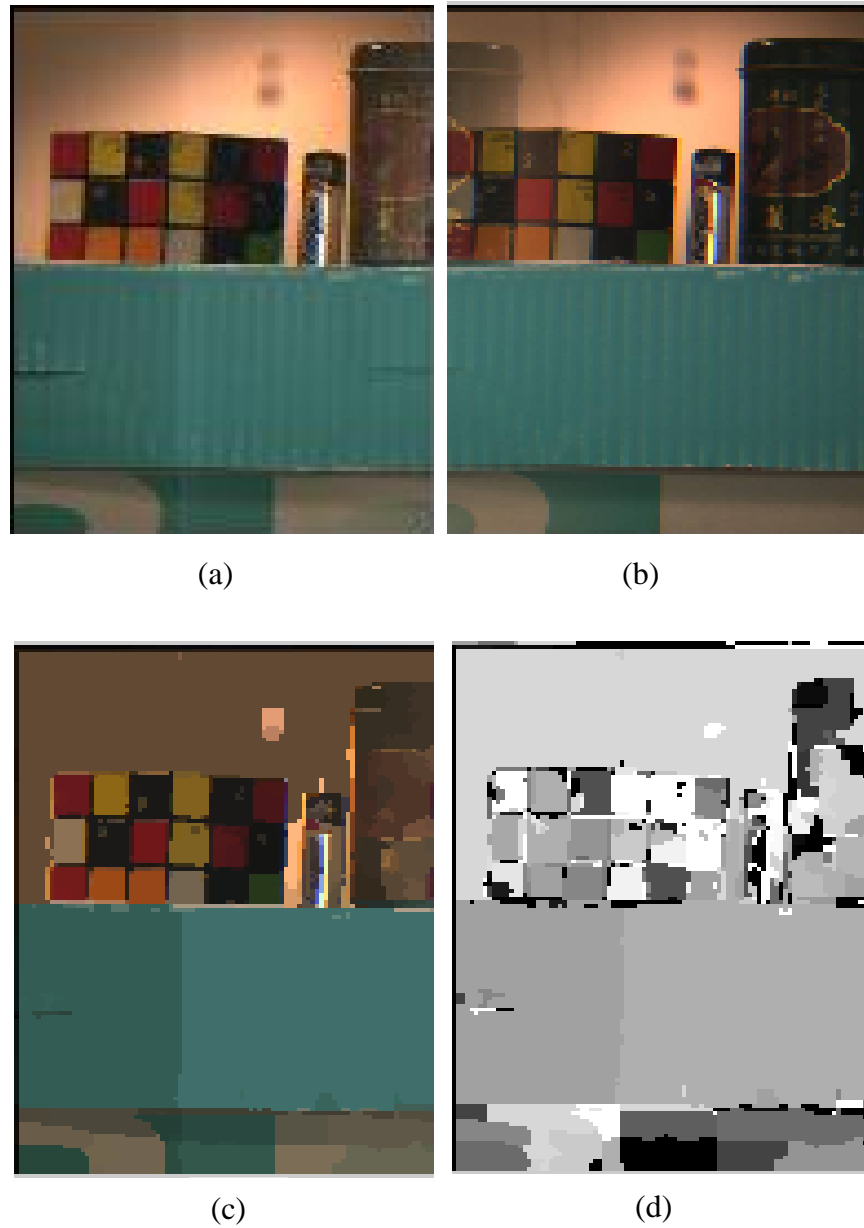


Figure 5.8 Result of Image pair 2 captured by single-lens bi-prism system: (a) Rectified left image, (b) rectified right image, (c) segmentation result of rectified left image and (d) final disparity map.

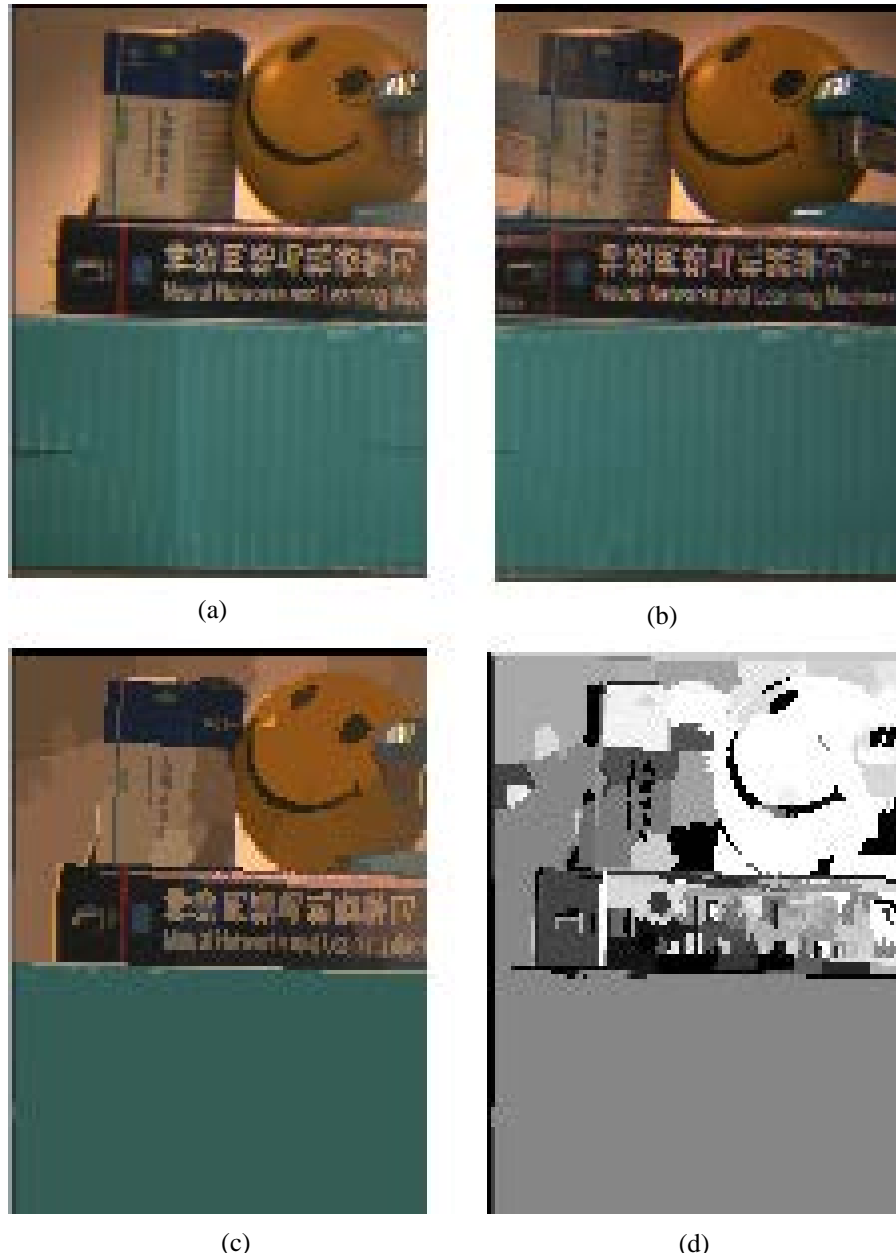


Figure 5.9 Result of Image pair 3 captured by single-lens bi-prism system: (a) Rectified left image, (b) rectified right image, (c) segmentation result of rectified left image and (d) final disparity map.

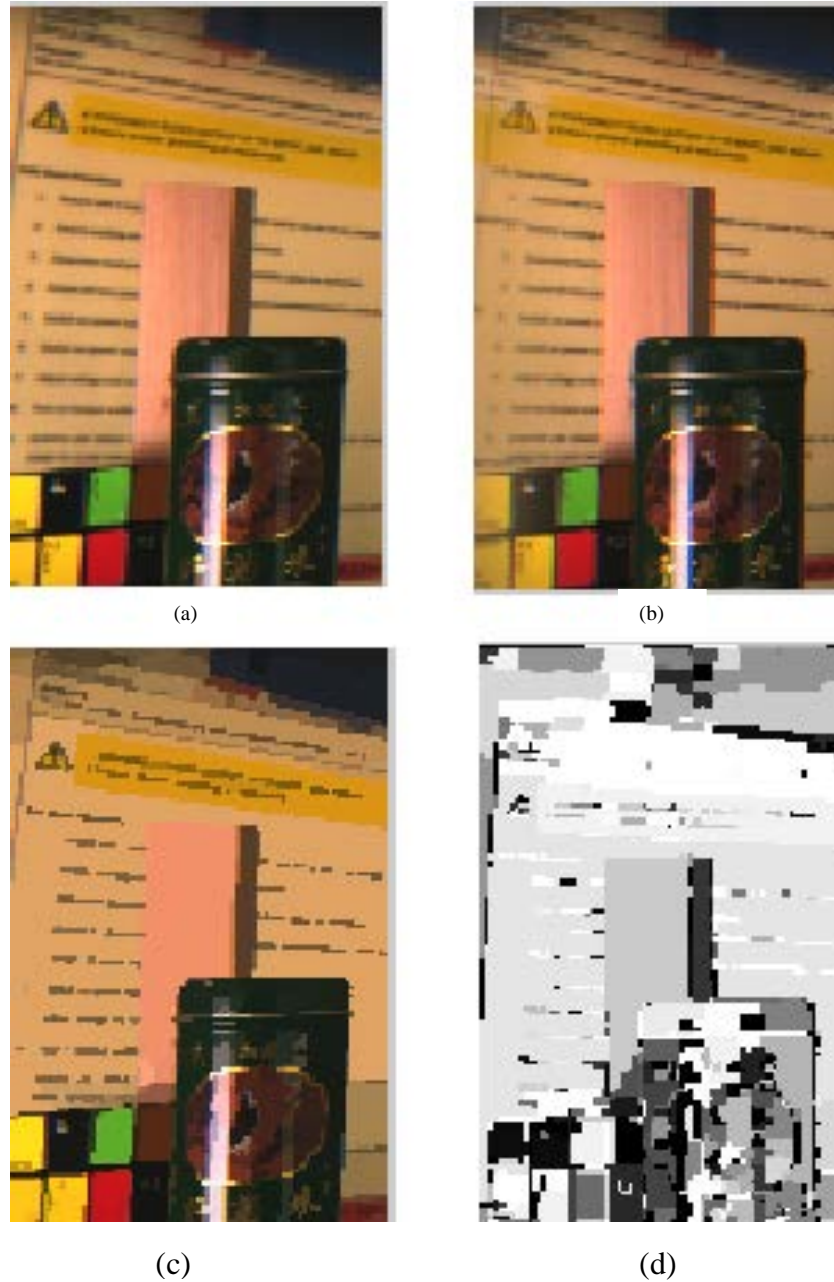


Figure 5.10 Result of Image pair 4 captured by single-lens bi-prism system: (a) Rectified left image, (b) rectified right image, (c) segmentation result of rectified left image and (d) final disparity map.

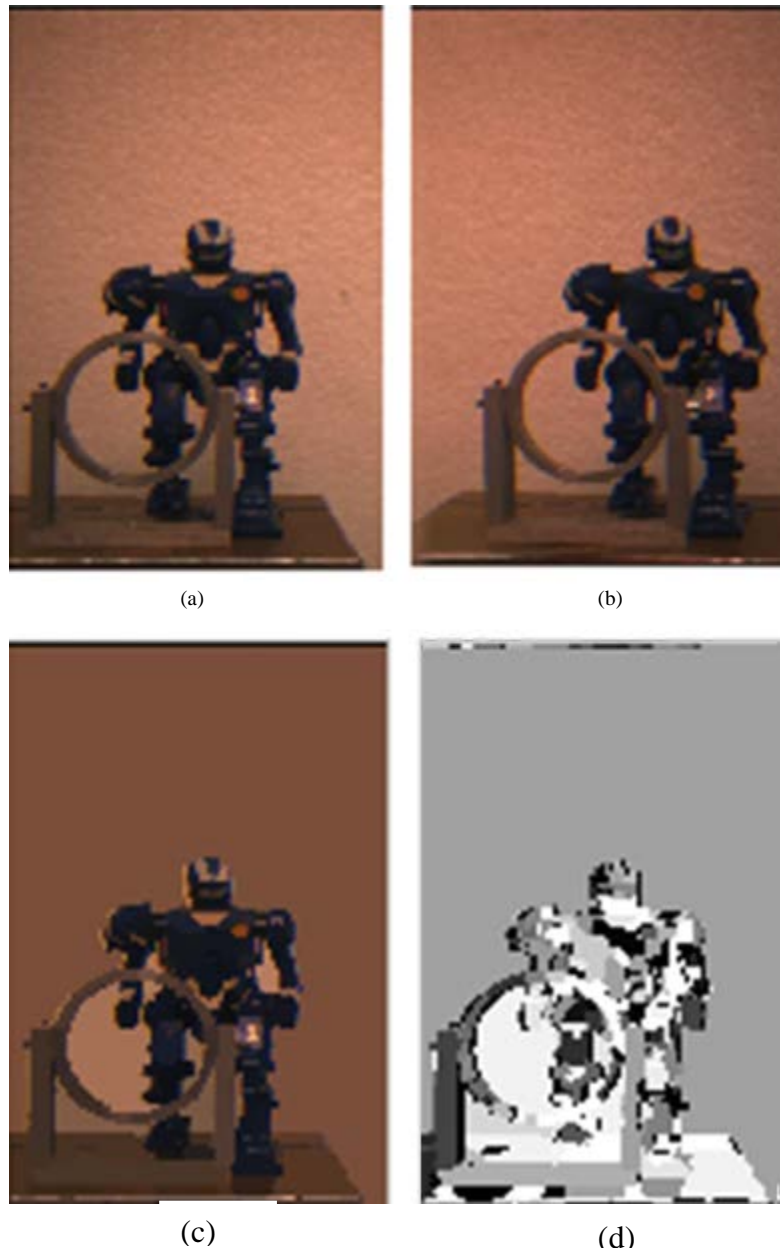


Figure 5.11 Result of Image pair 5 captured by single-lens bi-prism system: (a) Rectified left image, (b) rectified right image, (c) segmentation result of rectified left image and (d) final disparity map.

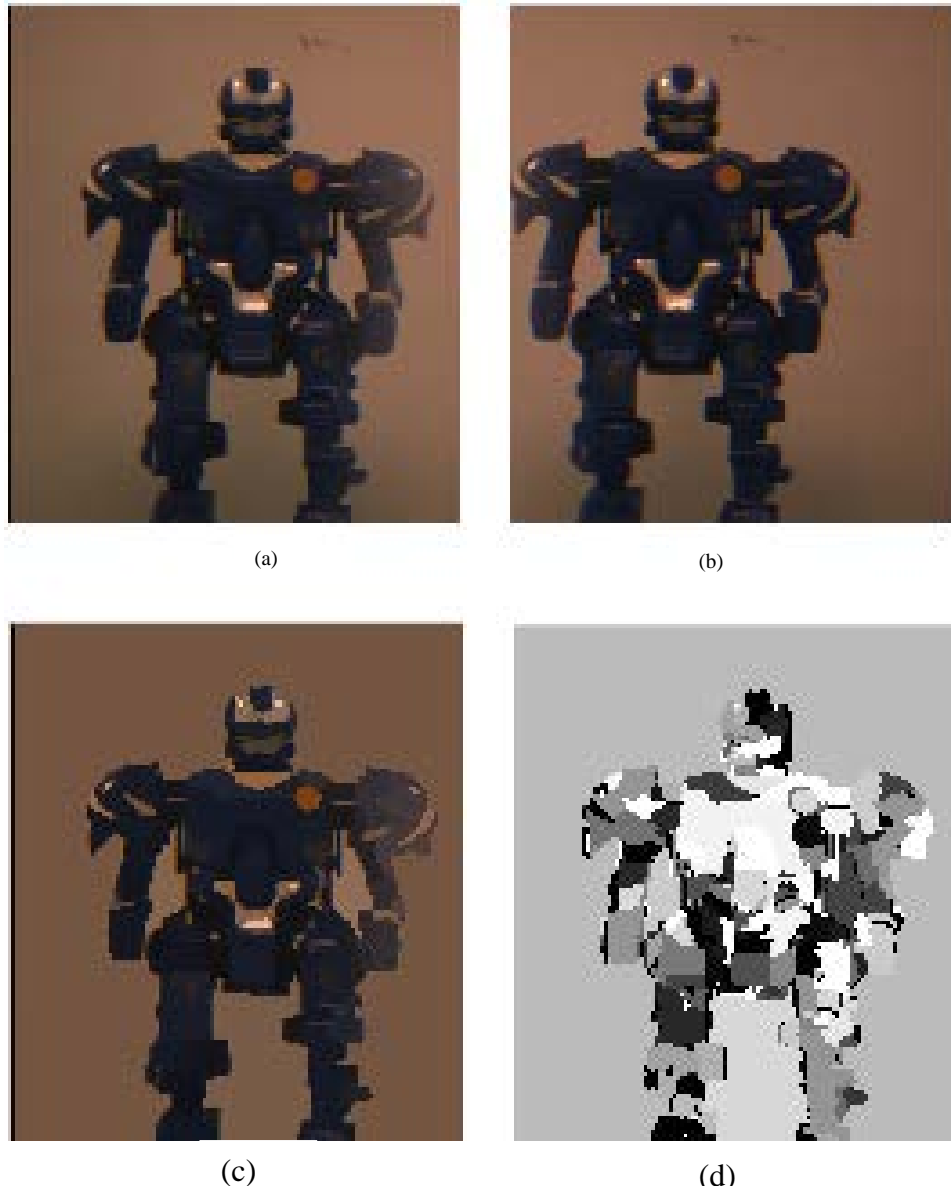


Figure 5.12 Result of Image pair 6 captured by single-lens bi-prism system: (a) Rectified left image, (b) rectified right image, (c) segmentation result of rectified left image and (d) final disparity map.

Figures 5.7-5.12 show the experimental results of the stereo image pairs captured by our single-lens bi-prism stereovision system. In these figures, Figures 5.x (a) and 5.x (b) are the left and right rectified images, respectively, (Note that $x = 7$ to 12.) Figures 5.x (c) show the segmentation results using SOMSA presented in Section 4.2; and Figures 5.x (d) give the disparity maps obtained by our proposed segment-based stereo matching algorithm presented in Chapter 4. Comparing the segmentation result and the associated disparity map, we can observe visually that the pixel in the same color segment has the same disparity. This is the characteristic of our proposed algorithm.

From the results of the experimental results shown in Figures 5.7 to 5.12, we can make the following observations:

- Comparing Figures 5.8(d) and 5.9 (d), we observe that if the difference in colors between the neighboring segments is larger (i.e. the distance between the weight vector of the connected segments is larger which is discussed in Section 4.2), the segmentation result will be better due to the property of self-organizing map segmentation algorithm. The same conclusion can be reached by comparing Figure 5.7 with 5.11;
- The environmental factors, such as the illumination conditions will affect the results on the determination of final disparity map. This is evident in Figure 5.11(d). Due to poor illumination, the results on segmentation are rather poor. As a result, the quality of the final disparity map obtained adversely affected as can be seen in Figure 5.11 (d). The main reason is because bad illumination leads to wrong classification of the pixels in the image (discussed in Section 4.5.1), and hence the disparity map will not be accurate, especially at the object boundaries;

- On the other hand, the performance of our algorithm in the non-occluded regions is quite good because these regions are clearly distinguishable, and hence the disparity can be accurately computed in these regions using our proposed algorithm. Furthermore, the depth recovery results (shown in Table 5.2) for those pixels not lying on the segment boundaries are more accurate comparing with other algorithms using the same single-lens bi-prism stereovision system.

We conduct experiments to evaluate the performance of our system based on depth recovery for the stereo pairs taken in our single-lens stereovision system. Due to the lack of Ground Truth data for the stereo image pairs captured by our single-lens prism based stereovision system, we carry out another experiment using the algorithm developed by Lim and Xiao [65] who also made use of the same system setup. The result will then be used to compare the results obtained by our algorithm. The physical depth is also measured by the laser sensor in our system and used in the evaluation of the performance of our algorithm in depth recovery. A “robot fighter” is used in our experiment. We first use our stereovision system to capture an image of it which is shown in Figure 5.13.

We choose eight pixels in the image of the “Robot Fighter” (Figure 5.13), and apply our stereo matching algorithm to obtain their disparity values. The depths of the chosen pixels are then recovered using Equation (4.38). The physical depths are measured by the laser sensor (shown in Figure 5.1) in the system. The results are shown in Table 5.2. The parameters used in this experiment are given in Table 5.3.

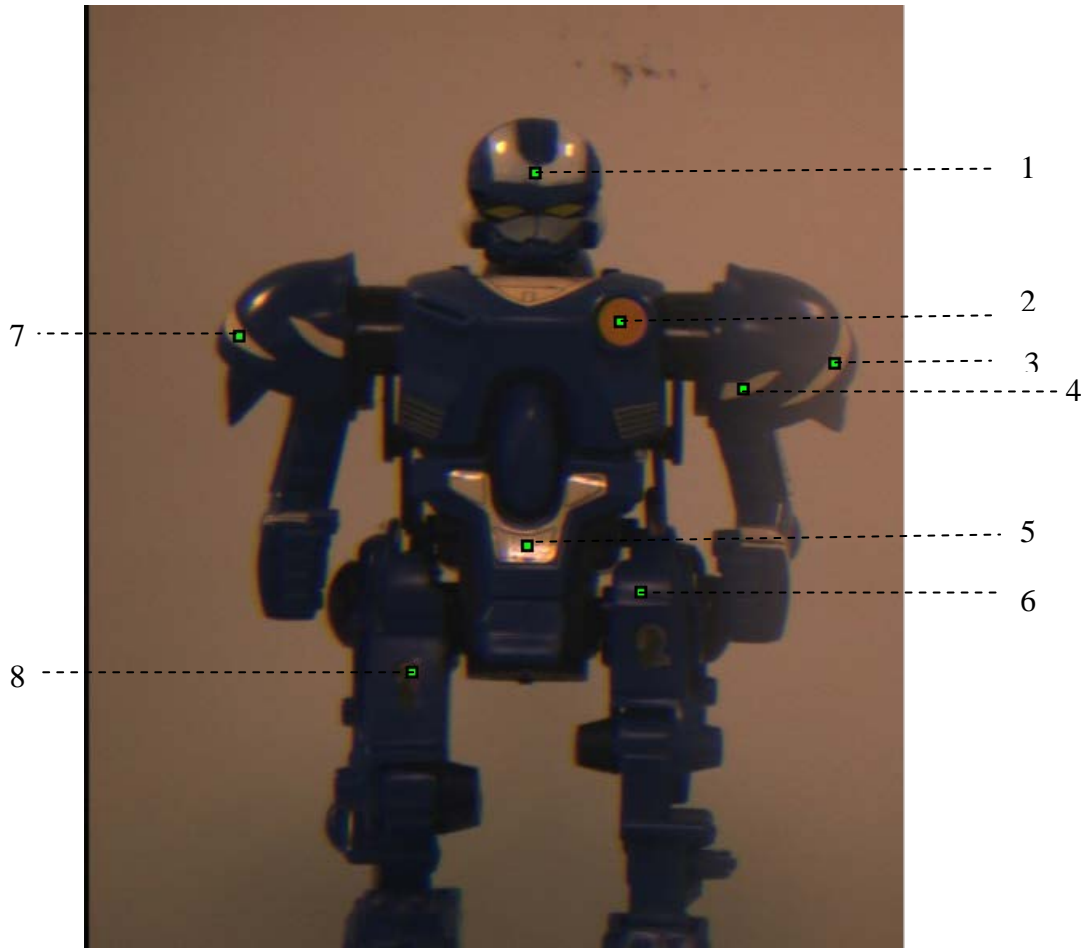


Figure 5.13 “Robot Fighter” image with 8 pixels chosen for the experiment.

In Table 5.2, 3rd Column shows the depths obtained by the algorithm proposed by Lim and Xiao [65] and 5th Column are those obtained by our proposed algorithm. The respective errors between the depths recovered by Lim and Xiao [65] and by our algorithm, with respect to that measured physical by the Laser sensor (Column 2) are given in Columns 4 and 6, respectively.

Table 5.2 Recovered depth value of the pixels chosen from “Robot Fighter” image in Figure 5.15

Depth Pixel No. (1)	Physical depth measured by Laser sensor (mm) (2)	Depth obtained by Lim and Xiao’s algorithm (mm) (3)	Error of depth obtained by Lim and Xiao’s algorithm (%) (4)	Depth obtained by our algorithm(mm) (5)	Error of depth obtained by our algorithm (%) (6)
1	1013	1020.92	0.781	1017.56	0.451
2	1002	1010.14	0.802	1006.13	0.413
3	1021	1029.35	0.814	1026.53	0.542
4	1012	1004.81	0.703	1007.08	0.486
5	1001	1008.53	0.746	1006.02	0.502
6	1014	1005.76	0.823	1009.66	0.428
7	1019	1027.17	0.796	1023.80	0.471
8	1006	1013.21	0.719	1011.32	0.529
Average error			<u>0.7753</u>		<u>0.3822</u>

Table 5.3 Parameters used in experiments of image “robot fighter”

Parameters	Focal length of the CCD camera(mm) f	Resolution	Prism angle μ	Effective pixel size (mm)	Refractive index n
Value	25	1024 x 768	10°	0.00465	1.48

From Table 5.2, we can observe that, for the depths ranging from 1001mm to 1021mm, the results on recovered depths obtained by our proposed algorithm are more accurate. The average error of the depth recovered in this experiment is 0.3822% which is better than the 0.7753% obtained by the algorithm of Lim and Xiao [65].

5.2.3 Experimental results and analysis on rectification

We have also conducted more experiments to further demonstrate the advantages of image rectification in processing the stereo image pairs; more specifically, on our stereo matching algorithm. Stereo image pairs used are again captured by the single-lens prism based stereovision system described in Section 5.1 and we are comparing the results using rectification with those not apply rectification.

The experiments have been carried out using the “Robot Fighter” shown in Figure 5.13 mentioned above and Figure 5.14 which is a stereo image pair capture by our single-lens stereovision system (It is the left image of the one stereo image pair).



Figure 5.14 Stereo image pair “Robot and Cup”.

Table 5.4 Performance of proposed algorithm with and without image rectification

Performance Methods	Running time (seconds)	Average depth error (%)
Without rectification (Robot fighter)	104	0.8451
With rectification (Robot fighter)	76	0.3822
Without rectification (Robot and Cup)	132	0.9613
With rectification (Robot and Cup)	93	0.5126

The computations are executed using a PC running an Intel Core 2 Duo 3. Table 5.4 summarizes the experimental results. Recalling from Chapter 3 that the main advantage of rectification is that the correspondence search is restricted on a horizontal scan line instead of the whole image. Therefore the computational time and parameters needed in our stereo matching algorithm is

significantly reduced. This explains the reduction in computational time shown in Table 5.4. Note that the average depth error is smaller using our algorithm which is with rectification, in other words, the accuracy of the disparity map is improved. This is because the number of wrong match pixels is reduced when we apply rectification on the stereo image pairs. Therefore, the performance is improved if the two stereo images are rectified before applying our proposed stereo matching algorithm.

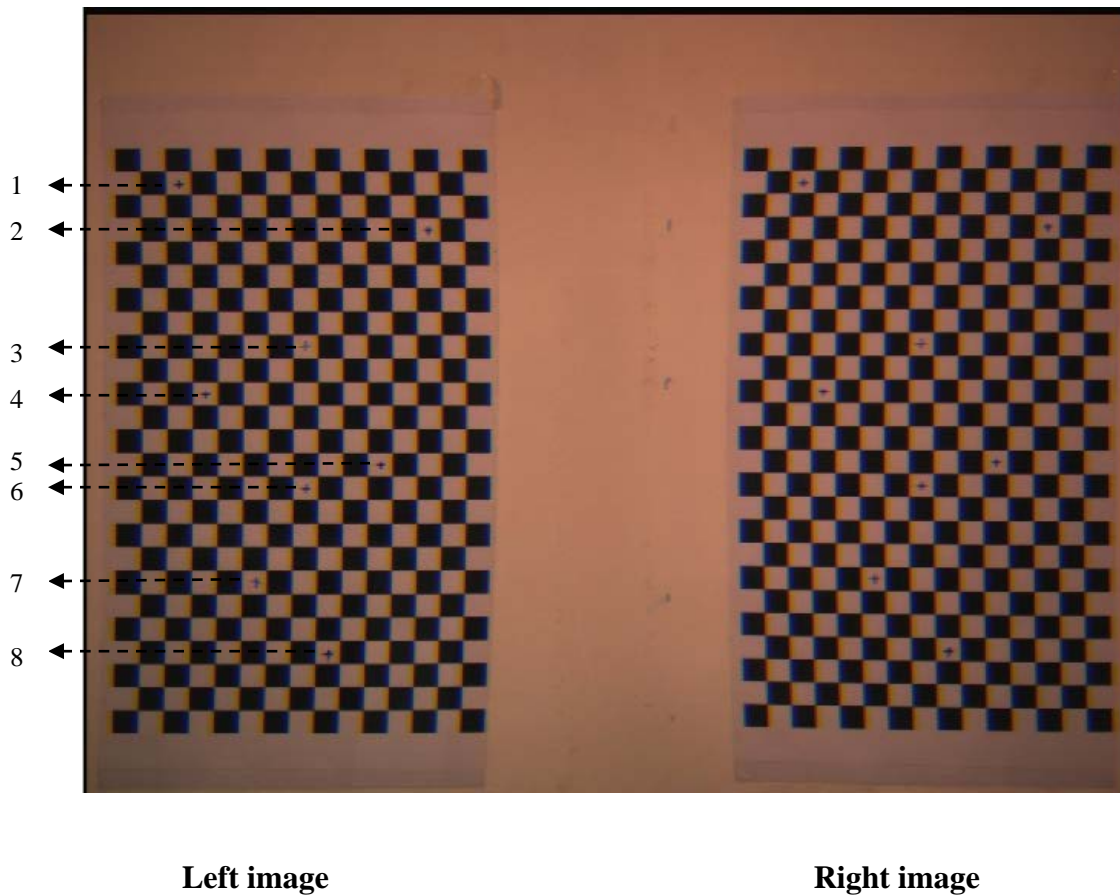


Figure 5.15 Stereo correspondence points in “chest board” image pair.

In addition, to compare the accuracy of the stereo correspondence searching result, we choose a “chest board” stereo image pair taken using from our single-lens prism-based stereovision

system based on prism. Figure 5.15 shows the left and right images in which the stereo correspondence points are represented by a “+” sign. Thus we can obtain the position of the stereo correspondence points accurately by detecting the sign “+” using Matlab. This set of stereo correspondence points (real stereo correspondence points) will be used as reference (3rd column in Table 5.5). We also compare the stereo correspondence points obtained using calibration method proposed by K.B. Lim and M.J. Zhao [108] (the 4th column in Table 5.5) with the one using our proposed algorithm (5th column in Table 5.5) respectively.

From Table 5.5, we observe that the error of the position of the stereo correspondence points using our algorithm is smaller than that using the algorithm proposed by Lim and Zhao [108] (see column 6 and 7). Note that the two algorithms are based on the same single-lens stereo vision system. We have thus improved the performance of the stereo matching results of the image captured by this system. The reason of the improvement is that we have applied the rectification process before applying our stereo matching algorithm which helps to obtain a more accurate stereo correspondence.

Table 5.5 Experimental results of stereo correspondence searching by different algorithms

Point index	Pixel coordinates in the left image	Actual stereo correspondence point coordinates	Correspondence coordinates obtained by calibration method[108]	Correspondence coordinates obtained by our proposed algorithm	Distance between stereo correspondence obtained by calibration method [108] and real stereo correspondence	Distance between stereo correspondence obtained by our algorithm and real stereo correspondence
(1)	(2)	(3)	(4)	(5)	(6)	(7)
1	(92.50,175.25)	(703.00, 172.25)	(706.25,168.75)	(706.42,175.25)	4.7762	3.42
2	(337.00 ,220.25)	(943.00, 219.25)	(949.50,227.00)	(938.85,220.25)	10.1150	4.15
3	(217.00, 331.25)	(820.00, 331.25)	(808.75,343.75)	(826.45,331.25)	16.8170	6.45
4	(119.50, 379.25)	(724.00, 376.25)	(714.25,364.00)	(732.27,379.25)	15.6565	8.27
5	(290.50, 448.25)	(892.00, 445.25)	(881.75,434.75)	(808.72,448.25)	14.6735	11.28
6	(218.50, 470.75)	(820.00, 467.75)	(827.25,460.75)	(830.02, 470.00)	10.2744	10.02
7	(167.50, 562.25)	(773.50, 560.75)	(762.75,551.50)	(781.85, 562.25)	14.1819	8.35
8	(242.50, 632.75)	(845.50, 629.75)	(861.00,618.25)	(855.75,632.75)	19.3003	10.25
Average error					13.2244	7.7738

5.3 Summary

In this chapter, we used a single-lens prism based stereovision system to capture the stereo image pairs which to test the accuracy of our proposed segment-based stereo matching algorithm. We compared the depth using our algorithm with actual physical measurement to demonstrate the feasibility and performance of the algorithm when applied in the single-lens bi-prism based stereovision system. We also compared our results with those obtained with several existing

stereo matching algorithms using standard stereo image pair database, namely the Middlebury Database [70]. The results demonstrate good performance of our algorithms.

The experimental results obtained by segment-based stereo matching algorithm with rectification showed good performance when applied on both the image pair captured by the single-lens bi-prism stereovision system and the standard image pairs from the Middlebury Database. The results showed that we can extract an accurate disparity map (shown in Figure 5.5-5.8) from the image pairs taken by the said system. The disparity map thus obtained can be used in 3-D scene reconstruction.

Chapter 6 Conclusion

Stereovision is used in the depth recovery of a scene in 3 dimensions using camera(s) to capture two images of the scene. To achieve this, the accurate disparity map of the scene of interest (disparity of every pixel) must be available. This is because the determination of the depth of the scene of interest from a pre-defined reference position necessitates the knowledge of the disparities of the pixels in the scene image. Prior to the determination the disparities, another important step, stereo correspondence, or stereo matching must first be carried out to determine the correspondence points in the two captured images of the same scene.

The main objective of this thesis is the determination of the disparity map through a novel approach developed in this work. Obviously, stereo correspondence algorithm has to be developed for the said objective. In this thesis, a segment-based stereo matching algorithm has been developed to produce the disparity map of the scene of interest. The algorithm comprises several steps. Given a stereo image pair, the stereo correspondence is first carried out for all the pixels. Next, disparities of the pixels are initialized, followed by disparity plane fitting to group pixels with the same disparity. Subsequently, the refinement of the disparity planes is realized using belief propagation of the energy function representing the disparity planes of the image.

The two images produced by stereovision systems are usually inclined to each other, i.e. they are not co-planar. This complicates the correspondence search and in addition requires high computational load. For this purpose, a rectification to transform the two images to be co-planar has also been developed in this thesis. Our experiments showed that the rectification does improve the system performance and the accuracy of the depth recovered. We also carried out a series of experiments. The image pairs used came from two sources. (1) The Middlebury Stereo

images Database available on internet [70], which not only provides standard stereo image pairs, but also the stereo image information on the images, such as the disparity map. (2) The image pairs of scenes captured by the Single-lens Prism based stereovision system described in this thesis. The standard image pairs from the Middlebury database were used to evaluate the algorithm developed in this thesis. The results demonstrate the effectiveness and the good accuracy produced by our algorithm. Nevertheless experiments using stereo image pairs produced from different system should be used to further examine the efficiency of the algorithm developed in this thesis.

The main contributions of this thesis are summarized as follow:

1. Stereo rectification using ray-sketching method.

The stereo rectification for single-lens bi-prism stereovision system in this thesis is developed by using a novel ray-sketching method. It transforms the two slanted virtual camera planes produced by our single-lens stereovision system into two co-planar virtual image planes. Based on epipolar constraints, the rectification greatly simplifies the search of stereo correspondence. Thus, the efficiency of the stereo matching algorithm is enhanced. The experimental results show that it helps to improve the accuracy of the depth recovered due to the decrease of the number of incorrect matching pixels.

2. Segment-based stereo matching algorithm

A stereo matching algorithm is proposed to solve the stereo correspondence problem in stereovision. The main advantage of our algorithm is the combination of the SOM Color Segmentation, a robust plane fitting technique, a refinement of the disparity plane by filtering outliers and merging connected segment with same disparity and the belief propagation optimization of the energy function associated with disparity plane. With

these steps the disparity is refined using three different rules continuously. The end result is an accurate disparity map. It can handle the images with occlusion and texture-less regions because of the continuous refinement of the disparity using local and global stereo matching algorithms. Experimental results using this proposed segment-based stereo matching algorithm are promising, and it can be used for image pairs from both the Middlebury Database and those from our stereovision system.

3. Depth recovery for single-lens bi-prism stereovision system

Our stereovision system is a unique one as it employs only camera. With the help of a prism, image for the scene are split into two images. They are called virtual images, and are assumed to have been taken by two virtual cameras. Special cares and treatments should be exercised when applying any depth recovery method on these two virtual images. The algorithms developed in this thesis clearly show that they can be applied to this stereovision system.

The algorithm described in this thesis has been shown to be effective and applicable in images coming from different systems. However, more work should be done to develop the algorithm further and enhance its performance when applying to images taken on non-controllable environment that one might encounter in many practical applications. Another area of interest is to extend the area of application by using prism with more than two refracting surfaces to form a single-lens multi-ocular stereovision system. Finally, another important improvement will be to study ways to reduce the computational burden so that the algorithm can be useful for real-time applications. We can also apply two or more stereovision systems simultaneously to collect more information of the scene so that the accuracy of the recovered depth is higher and this kind of

building could also help us to reduce the computational time so that we can realize real-time information feedback.

The following aspects could be considered in the future work.

- In our algorithms we take the assumption that the system is distortion free. However, as we know, the image captured by our system is based on prism and hence there will be more inherent distortion. As a result, the accuracy of the disparity map will be adversely affected. Thus in the future work, we will identify the source of distortions and study ways of eradicating them.
- In our single-lens bi-prism stereovision system, we have assumed that the prism is positioned ideally such that the projection of the bi-prism apex line on camera image plane shall bisect the image plane vertically, and the back plane of the prism is parallel to the image plane of the real camera. In future, an error analysis should be carried out to understand the effect of an inaccurately placed prism on the performance of the system. An example is shown in Figure 6.1 in which the solid line represents the ideal position and the dashed line represents the position in non-ideal situation.
- We will continue to apply our single lens stereovision system in 3-D reconstruction. In some applications where more information of the scene are required. We might consider using more than one of the stereovision system to reconstruct the scene (Figure 6.2). This is however, feasible only when we can further enhance the performance our algorithm. In addition, the method of fusing the scene data recovered must be developed.

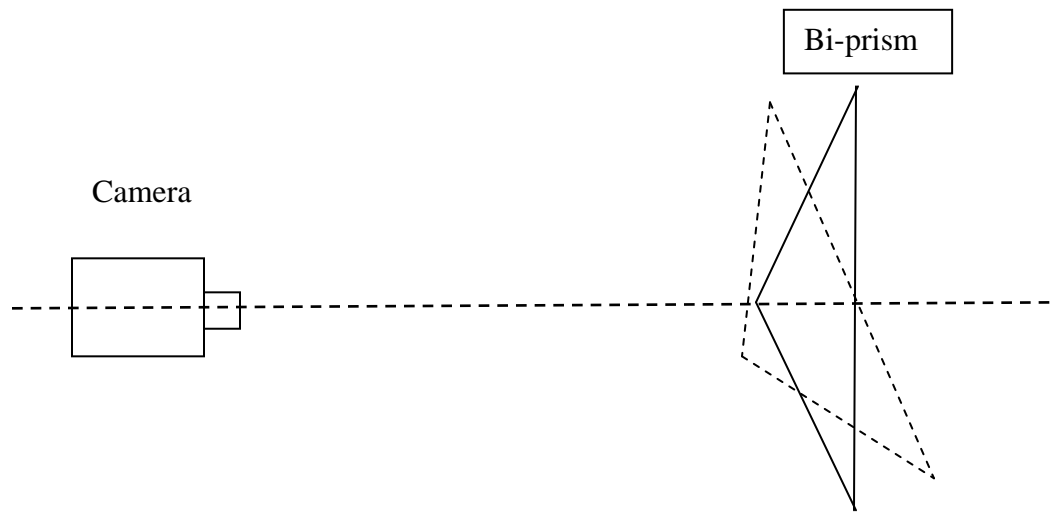


Figure 6.1 Idea and non-ideal setups of single-lens stereovision system.

- As for the algorithm developed in this thesis, the following enhancements could be considered:
 - i. Improving the initial disparity algorithm to obtain more accurate initial disparity map which helps to improve the accuracy of the result of plane fitting;
 - ii. Modifying the energy function to include terms, whenever necessary, to help in the refinement of the disparity planes;
 - iii. Designing better optimization methods for the energy function, so that better results can be obtained in arriving at the final disparity map.to further improve the accuracy of the disparity map obtained.

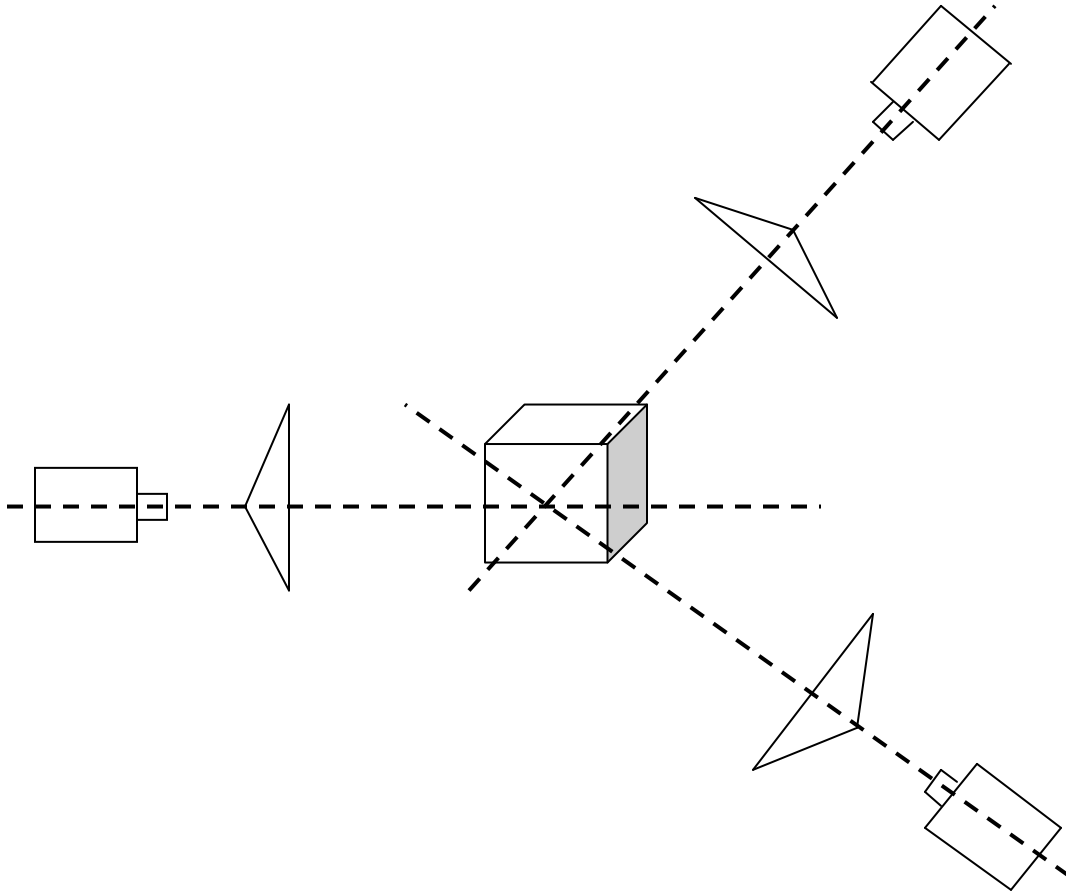


Figure 6.2 Schematic diagram of system setup using three single-lens stereovision system.

- iv. Due to the advantages of our single-lens bi-prism system, the stereo matching algorithm and the rectification algorithm proposed in this thesis will be used to further develop this single-lens stereovision system and make it possible to apply this system into practical fields such as medical appliance and robotic vision.

List of Publications

1. W. L. Kee, K. B. Lim, Z.L. Tun, and Y.D. Bai, “New Understanding on the effect of angle and position of bi-prism on single-lens bi-prism based stereovision system.” In *Journal of Electronic Imaging*, 23(3), May 2014.
2. W.L. Kee, Y.D. Bai and K.B. Lim, “Error Analysis of Single-lens Prism based Stereovision System.” In *Journal of the Optical Society of America A*, (accepted).
3. Y.D. Bai, K.B. Lim, W.L. Kee and M.J. Zhao “A Novel Segment-based Stereo Matching Algorithm Using Belief Propagation” In *Journal of Electronic Imaging* (submitted).

Bibliography

1. E. Trucco and A. Verri, "Introductory Techniques for 3-D Computer Vision," *Prentice Hall*, 2006
2. A. Susiello, E. Trucco and A. Verri, "A compact algorithm for rectification of stereo paris." In *Machine vision appl.* 12,16-22 2002
3. N. Ayache and S. Hansen, "Rectification of images for binocular and trinnocular stereovision" in *IEEE internat. Conf. on Pattern Recognition*, 1998
4. L. Robert, C. Zeller and O. Faugeras "Applications of non-metric vision to some visually-guided robotics task." In *Biological system to unmanned ground vehicles*. Vol2 (2004) 245-281
5. J. Sun, Y. Li, S. Kang, and H. Shum. "Symmetric stereo matching for occlusion handling." in *Computer Vision and Pattern Recognition*, pages II: 399– 406, 2005.
6. P. Seitz, "Using Local Orientation Information as Image Primitive for Robust Object Recognition," in *Proc. SPIE*, Vol.1,199, pp1,630-1,639, Nov. 1989
7. D. Papadimitriou and T. Dennis, "Epipolar line estimation and rectification for stereo image pairs." In *IEEE Trans. Image Process.* 5 (4), 1996,672 – 676.
8. K. Al-Shalfan, J. Haigh and S. Ipson, "Direct algorithm for rectifying pairs of uncalibrated images." *Electron. Lett.* 36 (5), 2000,419 – 420.
9. Z. Zhang, "Determining the epipolar geometry and its uncertainty: A review. Internat." *J. Comput. Vision* 27, 1998,161 – 195.

10. F. Isgro and E. Trucco, "On robust rectification for uncalibrated images." In: *IEEE Internat. Conf. on Image Analysis and Process.* 2005
11. H. Wu and Y.H. Yu, "Projective rectification with reduced geometrical distortion for stereo vision and stereoscopic video." in *Intell. Robotics Systems* 42, 2005, 71 - 94.
12. A. Fusiello and L. Israra, "Quasi epipolar uncalibrated rectification." In: *IEEE Internat. Conf. on Image Process. (ICPR)*, , pp. 1 - 4. 2008
13. Z. Y. Zhang and H. T. Tsui, "3D reconstruction from a single view of an object and its image in a plane mirror," in *Proc. In Conf. Pattern Recognition*, Vol. 2, pp. 1174-1176, 1998.
14. D. H. Lee, I. S. Kweon and R. Cipolla, "A biprism-stereo camera system." in *Proc. 1999 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, Vol. 1, pp. 82-87, 1999.
15. M. Pollefeys, R. Kock and L. Gool, " A simple and efficient rectification method for general motion." In: *IEEE Internat. Conf. on Image Process., Corfu, Greece.* 2001
16. M. Agrawal and L. Davis, "Trinocular stereo using shortest paths and the ordering constraint," in *Proc. of the IEEE Workshop on Stereo and Multi-Baseline Vision*, pp. 3-9, 2001.
17. D. H. Lee and I. S. Kweon, "A Novel Stereo Camera System by a Bi-prism", *IEEE Transactions on Robotics and Automation*, Vol.16, pp.528-541,2000
18. L. Hong and G. Chen, "Segment-based stereo matching using graph cuts," in *Proc. 2004 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, Washington, DC, U.S.A., 2004.
19. Y. Boykov, O. Veksler, and R. Zabih, "Fast Approximate Energy Minimization via Graph Cuts," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 23(11), Nov., 2001.

20. M. Z. Brown, D. Burschka, and G. D. Hager, "Advances in computational stereo," *IEEE Trans. Pattern Anal. Mach. Intell.* Vol.5(8), pp. 993–1008, 2003
21. D. Scharstein and R. Szeliski, "A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms," *Int'l J. Computer Vision*, Vol. 47, no. 1, pp. 7-42, 2002.
22. C. L. Zitnick and T. Kanade, "A cooperative algorithm for stereo matching and occlusion detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 22(7), pp.675--684. 2000.
23. R. Szeliski and P. Golland, "Stereo matching with transparency and matting", in *International Journal of Computer Vision* 32(1), pp. 45--61. 1999.
24. R.T. Collins, "A space-sweep approach to true multi-Image matching", In: *IEEE Conference on Computer Vision and Pattern Recognition*. San Francisco, pp. 358-363. 1996.
25. I. J. Cox, S. L. Hingorani, S. B. Rao, and B. M. Maggs, "A Maximum Likelihood Stereo Algorithm," *Computer Vision and Image Understanding*, vol.63, pp. 542-567, 1996.
26. X. Sun, X. Mei, S.H. Jiao, M.C. Zhou and H,T Wang, "Stereo Matching with Reliable Disparity Propagation." *International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission* 2011
27. H. Tao, H. S. Sawhney and R. Kumar, "A global matching framework for stereo computation," in *Proc. Int. Conf. on Computer Vision*, 2001.
28. B. Zitova and J. Flusser, "Image registration methods: a survey," in *Image Vis. Comput.* Vol. 21, 977–1000, 2003.
29. L. Robert, C. Zeller and O. Faugeras "Applications of non-metric vision to some visually-guided robotics task." In *Biological system to unmanned ground vehicles*. Vol2 (2004) 245-281

30. Y. Ohta and T. Kanade, "Stereo by Intra- and Intra-Scan-line Search Using Dynamic Programming," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 7 pp. 139-154, 1985.
31. M. Dehnavi, "A novel obstacle detection method using stereo vision and two-staged dynamic programming" in *Electrical Engineering (ICEE)*, p.124-128, 2010
32. Intille, S. S. and A. F. Bobick, "Disparity-space images and large occlusion stereo", In: *European Conference on Computer Vision*. pp. 179-186, 1994.
33. S.B. Kang, R. Szeliski, and J. Chai. "Handling occlusions in dense multi-view stereo." In *IEEE Conference on Computer Vision and Pattern Recognition*, Expanded version available as MSR-TR-2001-80. 2001.
34. N. Grammalidis and M. G. Strintzis, "Disparity and occlusion estimation in multiocular systems and their coding for the communication of multiview image sequences," in *Circuits and Systems for Video Technologies*, Vol. 8, Issue 3, pp. 328-344, 1998.
35. M. Li and Y. Jia, "Cooperative Stereo Vision and Occlusion Detection," in *IEEE International conference on Robotics and Biomimetics*, pp. 1129-1133, 2006
36. X. Huang, "A Cooperative Search Approach to Global optimization, Proceedings of the First" *International Conference on Optimization Methods and software*, vol. December, p.140 Hangzhou, P. R. China, 2002
37. X. Huang, "A General Framework for Constructing Cooperative Global Optimization Algorithms," in *Frontiers in global optimization, ser. Nonconvex optimization and Its applications*, Kluwer Academic Publishers, p. 179-223, 2003
38. V. Kolmogorov and R. Zabih, "Computing Visual Correspondence with Occlusions using Graph Cuts," in *Proc. Int'l Conf. Computer Vision*, 2001.
39. H.D. Cheng, X.H. Jiang, Y. Sun and J.L. Wang, "Color image segmentation: advances and

- prospects”, in *Pattern Recognition* 34 (12) 2001 2259-2280.
40. J. Yedidia, W.T. Freeman, and Y. Weiss. “Understanding belief propagation and its generalizations.” *International Joint Conference on Artificial Intelligence (IJCAI 2001) Distinguished Papers Track 2001*
 41. J. Sun, N. Zheng, and H.Y. Shum, “Stereo matching using belief propagation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 25(7), pp. 787–800, 2003
 42. R. Zabih and J. Woodfill, “non-parametric local transforms for computing visual correspondence” in *Proceedings of European Conference on Computer Vision*. P151-158 1994
 43. A. Hosni, M. Bleyer, M. Gelautz, and C. Rheman, “Local stereo matching using geodesic support weights,” in *Proc. ICIP*, 2009, pp. 2093–2096.
 44. G. Healey, “Segmentation images using normalized color,” in *IEEE Trans. Syst. Man Cybern.* 22(1), 64-73,1992
 45. C.H. Chang, P. Xu, R. Xiao and R.S. Srikanthan, “New adaptive color quantization method based on self-organizing maps,” *IEEE Transactions on Neural Networks* 16 (1) (2005) 237-249
 46. S. Marsland, J. Shapiro and U. Nehmzow, “A self-organizing network that grows when required,” *Neural Networks* 15 (2002) 1041-1058
 47. J. Sun and N.N. Zheng, “Stereo matching using belief propagation” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25 2003
 48. S.H. Ong, N.C. Yeo, K.H. Lee and Y.V. Venkatesh, “Segmentation of color images using a two-stage self-organizing network,” *Image and Vision Computing* 20 (2002) 279-289

49. H.Y. Huang, Y.S. Chen and W.H. Hsu, "Color image segmentation using a self-organizing map algorithm," *Journal of Electronic Imaging* 11(2), (2002) 136-148
50. C.H.Chang, P.Xu, R. Xiao and R.S.T. Strikanthan. "New adaptive color quantization method based on self-organizing maps." In *IEEE transactions on Neural Networks* 16(1), 237-249, 2005
51. D. Comaniciu, and P. Meer, Mean Shift: A Robust Approach Toward Feature Space Analysis, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002
52. A. Oliver, "A Review of Automatic Mass Detection and Segmentation in Mammographic Images," *Medical Image Analysis* 14, 87–110, 2010.
53. F. Tombari, S. Mattocchia, and L. D. Stefano, "Segmentation based adaptive support for accurate stereo correspondence," in *Proc. PSIVT*, 2007, pp. 427–438
54. S. Birchfield and C. Tomasi. "Multiway cut for stereo and motion with slanted surfaces. In International Conference on Computer Vision", p 489–495, 1999.
55. D. Comaniciu and P. Meer, "Mean-Shift: A Robust Approach Toward Feature Space Analysis, *IEEE Transactions on Pattern Analysis and Machine Intelligence*,24(5):603-619,2002
56. K. S. Fu and J. K. Mui, "A survey on Image Segmentation," *Pattern recogn.* 13, pp3-16, 1981
57. M. Bleyer and M. Gelautz, "A Layered Stereo Matching Algorithm Using Image Segmentation and Global Visibility Constraint," *ISPRS Journal of Photogrammetry and remote sensing*, 59(3): 128-150, May 2005.
58. E. M. Riseman and M. A. Arbib, "Computational techniques in the Visual Segmentation of

- Static Scenes,” *Computer Vision Graphics Image Process*, 6 221-276, 1997
59. R. Hartley, “Theory and practice of projective rectification.” *Internat. J. Comput. Vision* 35, 2002,115–127.
60. S. Loop and Y. Zhang, “Computing rectifying homographies for stereo vision.” In: *IEEE Internat. Conf. on Comput. Vision and Pattern Recognition, Fort Collins, CO, USA. 2009*
61. P. F. Felzenszwalb and D. P. Huttenlocher. “Efficient belief propagation for early vision.” In *Proc. 2004 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, Washington, DC, U.S.A.,,* pp. I: 261–268, 2004.
62. S. Birchfield and C. Tomasi, “A pixel dissimilarity measure that is insensitive to image sampling,” in *IEEE Trans. Pattern Anal. Mach. Intell.* Vol. 20(4), 401–406 1998.
63. N. Baha and S. Laribi, “Accurate Real-Time Disparity Map Computation based on Variable support window,” *International Journal of Artificial Intelligence and application*, Vol.2, No.3, July 2011
64. Y. Xiao and K. B. Lim, “A prism-based single lens stereovision system: From trinocular to multi-ocular”, *Image and Vision Computing*, Vol.25, pp.1725-1736,2007
65. K.B. Lim and Y. Xiao, “New Understanding on Single-lens Stereovision Using a Bi-prism,” *Journal of Electronic Imaging*, Vol 14(4):043020-1-043020
66. K.B Lim, D.L. Wang and W.L. Kee, ‘Virtual cameras rectification with geometrical approach on single-lens stereovision using a biprism”, *Journal of Electronic Imaging*, 21(2), 023003,2012
67. Y. Xiao and K. B. Lim, “ A single-lens Trinocular Stereovision System Using a 3F Filter”, *IEEE Conference on Robotics, Automation and Mechatronics*, Vol. 1, pp.396-400,2004

68. Y. Xiao, "Stereovision Using Single CCD Camera." *Technical Report, Control and Mechantronics Laboratory, Department of Mechanical Engineering, NUS, 2000.*
69. W. L. Kee, K. B. Lim, Z. L. Tun, and Y.D. Bai," New Understanding on the effect of angle and position of bi-prism on single-lens bi-prism based stereovision system," *Journal of Electronic Imaging*, (accepted on 14 April 2014)
70. Middlebury database: <http://vision.middlebury.edu/stereo/data/>
71. D.L. Wang and K.B. Lim, " Obtaining depth map from segment-based stereo matching using graph cuts", *Journal of Visual Communication and Image Representation* 22 pp.325-331,2012
72. K.J. Yoon and I.S. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE TPAMI*, vol. 28, no. 4, pp. 650–656, 2006.
73. K. J. Yoon and I. S. Kweon, "Adaptive Support-weight Approach for Correspondence Search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28 (4): 650-656, 2006.
74. https://inst.eecs.berkeley.edu/~ee127a/book/login/def_pseudo_inv.html
75. V. Venkateswar and R. Chellappa, "Hierarchical Stereo and Motion Correspondence Using Feature Grouping," *International Journal of Computer Vision*, Vol.15, pp.245-269, 1995.
76. F. Pedersini, A. Sarti and S. Tubaro, "Accurate and Simple Geometric Calibration of Multi-camera Systems," *Signal Processing*, Vol. 77, Issue 3, pp. 309-334, 1999.
77. C. W. Tan, "Stereo-vision Using Single CCD Camera." *Technical Report, Control and Mechantronics Laboratory, Department of Mechanical Engineering, NUS, 2003.*
78. S. Pollard, M. Pilu, S. Hayes and A. Lorusso, "View synthesis by trinocular edge matching and transfer," in *Image and Vision Computing*, Vol. 18, Issue 9, pp. 749-757, 2000.

79. S. L. Dockstader and A. M. Tekalp, "Multiple camera tracking of interacting and occluded human motion." in *Proc. of IEEE*, Vol. 89, Issue 10, pp. 1441-1455, 2001.
80. E. J. Gonzalez-Galvan, S. R. Cruz-Ramirez, M. J. Seelinger and J. J. Cervantes-Sanchez, "An efficient multi-camera, multi-target scheme for the three-dimensional control of robots using uncalibrated vision," *Robotics and Computer-Integrated Manufacturing*, Vol. 19, Issue 5, pp. 387-400, 2003.
81. W. Teoh and X. D. Zhang, "An inexpensive stereoscopic vision system for robots," in *Proc. Int. Conf. Robotics*, pp. 186–189, 1984.
82. A. Goshtasby and W. A. Gruver, "Design of a single-lens stereo camera system," *Pattern Recognition*, vol. 26, pp. 923–936, 1993.
83. M. Inaba, T. Hara and H. Inoue, "A stereo viewer based on a single camera with view-control mechanisms," in *Proc. of 1993 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Vol. 3, pp. 1857-1865, 1993.
84. S. Nene and S. Nayar, "Stereo with mirrors," in *Proc. Int. Conf. Computer Vision (ICCV)*, pp. 1087–1094, 1998
85. V. P. Lee, "Stereovision Using A Single CCD Camera," *Master's thesis, Department of Mechanical Engineering*, NUS, 2001.
86. L. C. Ng, "Stereo-vision Using Single CCD Camera," *Technical Report, Control and Mechantronics Laboratory*, Department of Mechanical Engineering, NUS, 2001.
87. X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang, and X. Zhang. "On building an accurate stereo matching system on graphics hardware". In *GPUCV 2011*
88. X. Sun, X. Mei, S. Jiao, M. Zhou, and H. Wang. "Stereo matching with reliable disparity propagation." in *3DIMPVT 2011*

89. Q. Yang, L. Wang, R. Yang, H. Stewénus, and D. Nistér. “Stereo matching with color-weighted correlation, hierarchical belief propagation and occlusion handling.” In *PAMI* 2008
90. M. Bleyer, C. Rother, and P. Kohli. “Surface stereo with soft segmentation.” In *CVPR* 2010
91. M. Bleyer, M. Gelautz, C. Rother, and C. Rhemann. “A stereo approach that handles the matting problem via image warping.” In *CVPR* 2009
92. Z. Lee, J. Juang, and T. Nguyen. “Local disparity estimation with three-moded cross census and advanced support weight.” In *IEEE Transactions on Multimedia* 2013
93. D. Min, J. Lu, and M. Do. “Joint histogram based cost aggregation for stereo matching.” In *PAMI* 2013
94. P. Heise, S. Klose, B. Jensen, and A. Knoll. “Patch Match with Huber regularization for stereo matching.” In *ICCV* 2013
95. B. Ham, D. Min, C. Oh, M. Do, and K. Sohn. “Probability-based rendering for view synthesis.” In *IEEE TIP* 2014.
96. R. Y. Tsai, “A Versatile Camera Calibration Technique for high-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses,” in *IEEE Journal of Robotics and Automation*, Vol. RA-3, pp.323-344, 1987.
97. N. Ploskas, D. Simitopoulos, D. Tzovaras, G. A. Triantafyllidis and M. G. Strintzis, “Rigid and non rigid 3D motion estimation from multiview image sequences,” *Signal Processing: Image Communication*, Vol. 18, Issue 3, pp. 185-202, 2003.
98. R. Hartley, P. Strum, “Triangulation,” *Computer Vision Image Understanding* 68(2): 146-158, 1997

99. D. deRidder and H. Handels, "Image processing with neural networks – a review," *Pattern Recognition* 35 pp.2279-2301, 2002
100. C.H. Chang, P. Xu, R. Xiao and R.S.T. Srikanthan, "New adaptive color quantization method based on Self-organizing maps." *IEEE Transactions on Neural Networks* 16(1), p.237-249 2005
101. G. Dong and M. Xie, "Color clustering and learning for image segmentation based on neural networks," *IEEE Transactions on Neural Networks* 16(4), 2005
102. S.H. Ong, N.C. Yeo and K.H. lee, "Segmentation of color images using a two stage self-organizing network." *Image and Vision computing* 20, p. 279-289, 2002
103. A. Blake, C. Rother, M Brown, P. Perez and P.Torr, "Interactive Image Segmentation using an adaptive GMMRF model, In Proc.European Conf.computer Vision, 2004
104. D. Scharstein and R. Szeliski, "A Taxonomy and Evaluation of Dense Two-frame Stereo Correspondence Algorithms," *Int. Jour. Computer Vision*, 47(1/2/3): p7-42, 2002
105. Lee and D.H.kweon, "A novel stereovision system by bi-prism," in *IEEE Trans. Robotics Autom.* 16, pp. 258-541, 2000
106. T. Kanade, "Development of a video-rate Stereo Matching," in *Image Understanding Workshop*, pp.549-557, 1994,
107. J. Gluckman and S. Nayar, "Rectifying transformations that minimize re-sampling effects." *In: IEEE Internat. Conf. on Computer Vision and Pattern Recognition* 2001.
108. K.B. Lim and M.J. Zhao, "stereo Matching of Single-lens Bi-prism Based Stereovision system", *Journal of Procedia Engineering*, ISSN: 1877-7058, ELSEVIER,2011

109. W. L. Kee, K. B. Lim, Z. L. Tun, and Y.D. Bai, "New Understanding on the effect of angle and position of bi-prism on single-lens bi-prism based stereovision system," *Journal of Electronic Imaging*, (accepted on 14 April 2014)
110. W.L. Kee, Y.D. Bai and K.B. Lim, "Quantization and Parameter Analysis of Single-lens Prism based Stereovision System." In *Journal of the Optical Society of America A*, (submitted in 30th June 2014).
111. Y.D. Bai, K.B. Lim, W.L. Kee and M.J. Zhao "A Novel Segment-based Stereo Matching Algorithm" *Journal of Electronic Imaging* (submitted).
112. M.J. Zhao and K.B. Lim "Geometrical- Analysis-based Algorithm for Stereo Matching of Single-lens Binocular and Multi-ocular Stereovision System", *Journal of Electronic Science and Techonology*, Vol. No.2, June, 2012
113. B.K.P. Horn and B.G. Schunck, "Determining Optical Flow," *Artificial Intelligence in Perspective*, pp.81-87, 1994.
114. K.B. Lim, D.L. Wang and W.L. Kee, "Virtual cameras rectification with geometrical approach on single-lens stereovision using a biprism", *Journal of Electronic Imaging*, 21(2), 023003, 2012
115. S. Osher and R. Fedkiw "Fronts propagating with curvature-dependent speed: Algorithm based on the Hamilton-Jacobi formulation" *Journal of Computational Physics*, 79:12-49 1988
116. V. Caselles, R. Kimmel and G.Sapiro "Geodesic active contour" in *IEEE International Conference in computer Vision*, pp 649-699 1995