

~~2397~~

2397

# Stellingen (Propositions)

behorende bij het proefschrift

**Integration of Segmentation and Stereo Matching**

door **Yaonan Zhang**

16 juni 1994

1. Because recovery of information about the world from a single cue such as motion parallax, binocular stereo disparity, or shading in images tends not to be very robust, there is now a great deal of interest in integrating information from multiple cues. The integration of early vision modules will be required for most practical applications of vision systems.

Chapter 1, this thesis

2. Curve matching methods should be capable of dealing with perspective transformation and of handling possible errors due to bad segmentation and occlusion without any assumptions on the shape of the curves.

Chapter 4, this thesis

3. It is very difficult and it may be impossible to improve any single segmentation method for consistent segmentation results in a stereo pair.

Chapter 8, this thesis

4. It is well known that a human being can use prior knowledge to enhance the visual interpretation, but "even organisms with relatively simple nervous systems can use prior experience to help the processing of visual images".

S.W. Zhang & M.V. Srinivasan, "Prior experience enhances pattern discrimination in insect vision", *Nature*, Vol. 368, pp. 330-332, 1994

5. "The simulation of evolution for the purpose of parameter optimization has generally demonstrated itself to be a robust and rapid optimization technique. ... Indeed, the simulation of evolution may eventually prove to be the only practical path to the development of ontogenetic machine intelligence".

W. Atmar, "Notes on the simulation of evolution",  
*IEEE Transactions on Neural Networks*, Vol. 5, pp. 130-147, 1994

6. Scale is poorly understood, and mathematics has nothing to say about it, so we should be careful with the use of scale.
7. Image interpretation (understanding) will be generally unrealistic and impractical within the next few decades, and the increasing computer power will not help very much in solving the problem.
8. Many problems and the ways to solve them in science and technology show similarities and hierarchies; maybe they can be abstracted as an object-oriented tree.
9. Many scientific publications are publication oriented, not problem-solving oriented.
10. The Internet is a network of neural networks.
11. Knowledge is power, but power is not necessarily based on knowledge.

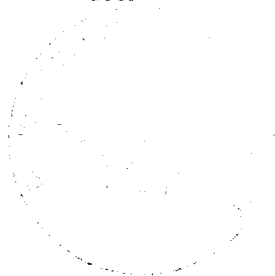
# Integration of Segmentation and Stereo Matching

Yaonan Zhang

# Integration of Segmentation and Stereo Matching

## Proefschrift

ter verkrijging van de graad van doctor  
aan de Technische Universiteit Delft,  
op gezag van de Rector Magnificus Prof.ir. K.F. Wakker,  
in het openbaar te verdedigen ten overstaan van een commissie,  
door het College van Dekanen aangewezen,  
op donderdag 16 juni 1994 te 16.00 uur  
door



Yaonan ZHANG,

master of science in  
photogrammetry and remote sensing,  
geboren te Rugao City, Jiangsu Province, China.

Dit proefschrift is goedgekeurd  
door de promotoren  
prof.dr.ir. E. Backer en prof.dr.ir. G.H. Ligterink  
en de toegevoegd promotor dr.ir. J.J. Gerbrands

CIP-DATA KONINKLIJKE BIBLIOTHEEK, DEN HAAG

Zhang, Yaonan

Integration of segmentation and stereo matching / Yaonan

Zhang. - [S.l. : s.n.]

Thesis Technische Universiteit Delft. - With ref.

ISBN 90-9007256-X

Subject headings: computer vision / stereo matching.

Copyright © 1994 by Yaonan ZHANG

All rights reserved. No part of this thesis may be reproduced or transmitted in any form or by any means, electronic, mechanical, photocopying, any information storage and retrieval system, or otherwise, without written permission from the copyright owner.

To my parents

To Xie Chen

# Contents

<b>Summary</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
1.1 General problems of computer vision . . . . .	1
1.2 The stereo vision problem . . . . .	3
1.2.1 General description of the problem . . . . .	3
1.2.2 Signal-based matching . . . . .	6
1.2.3 Point-based and edge-based matching . . . . .	7
1.2.4 Region-based matching . . . . .	7
1.2.5 Structure-based matching . . . . .	8
1.2.6 Other matching methods . . . . .	9
1.3 Fusion of vision modules . . . . .	9
1.3.1 Shape from shading and motion . . . . .	9
1.3.2 Shape from texture . . . . .	10
1.3.3 Shape and 3-D motion from contour and stereo . . . . .	10
1.4 Scope of this thesis . . . . .	11
<b>2 MDL-Based Region Growing</b>	<b>15</b>
2.1 Introduction . . . . .	15
2.2 Minimum description length . . . . .	17
2.3 Using MDL to encode the intensities . . . . .	21
2.4 Surface models and their robust estimation . . . . .	22
2.4.1 Surface models . . . . .	22
2.4.2 M-estimators . . . . .	23
2.5 Region Growing Algorithm . . . . .	26
2.6 Experiments . . . . .	29
2.7 Conclusions . . . . .	30

<b>3</b>	<b>Shape-Constrained Region Growing</b>	<b>37</b>
3.1	Introduction . . . . .	37
3.2	Encoding the shape of regions . . . . .	38
3.3	Boundary Fitting Algorithm . . . . .	39
3.3.1	Detection of extreme points . . . . .	40
3.3.2	Splitting process . . . . .	42
3.4	Examples . . . . .	44
3.5	Concluding remarks . . . . .	44
<b>4</b>	<b>Curve Matching</b>	<b>51</b>
4.1	Introduction . . . . .	51
4.2	Decomposition of slant, tilt and scale factor . . . . .	55
4.2.1	Projective geometry . . . . .	55
4.2.2	Determination of slant $\Omega$ and tilt $\Gamma$ of an object plane . . . . .	57
4.2.3	Determination of the scale factor . . . . .	59
4.3	Coordinate transformation . . . . .	60
4.4	Curve matching algorithm . . . . .	61
4.4.1	Point corresponding for stereo curves . . . . .	62
4.4.2	Weighting the curve direction . . . . .	65
4.5	Parameter estimation and matching evaluation . . . . .	65
4.5.1	Three-point configuration . . . . .	65
4.5.2	Histogram analysis . . . . .	66
4.5.3	Curve normalization . . . . .	67
4.5.4	Distance transformation . . . . .	68
4.5.5	Matching measure . . . . .	68
4.6	Experimental results . . . . .	69
4.7	Conclusions . . . . .	70
<b>5</b>	<b>Fusion of Stereo Clues</b>	<b>77</b>
5.1	Introduction . . . . .	77
5.2	Fuzzy measure and fuzzy integral . . . . .	79
5.2.1	Fuzzy measures . . . . .	80
5.2.2	$g_\lambda$ - Fuzzy measures . . . . .	82
5.2.3	Fuzzy integrals . . . . .	83
5.3	Extension and generalization of the fuzzy integral . . . . .	86
5.4	Identification of fuzzy measure . . . . .	87
5.5	Learning of fuzzy measure . . . . .	91
5.6	Fusion of stereo clues . . . . .	92

5.7	Experimental results . . . . .	97
5.8	Conclusions . . . . .	98
<b>6</b>	<b>Integration of Segmentation and Stereo Matching</b>	<b>99</b>
6.1	Introduction . . . . .	99
6.2	A feedback system for integrating segmentation and matching . . . . .	101
6.2.1	Representation of each layer . . . . .	102
6.2.2	Operations on one layer . . . . .	105
6.2.3	Operations between the layers . . . . .	105
6.2.4	Data fusion . . . . .	106
6.2.5	Domain models . . . . .	106
6.2.6	Computation mechanism . . . . .	107
6.3	Algorithms . . . . .	107
6.4	Experimental results . . . . .	111
6.5	Conclusions . . . . .	112
<b>7</b>	<b>Global Matching</b>	<b>117</b>
7.1	Introduction . . . . .	117
7.2	Relaxation labeling . . . . .	118
7.2.1	Definition of the labeling problem . . . . .	119
7.2.2	Exactly consistent labeling . . . . .	120
7.2.3	Minimally inconsistent labeling . . . . .	121
7.2.4	Extension of conventional relaxation labeling . . . . .	122
7.3	Formulation of global stereo matching as relaxation labeling . . . . .	123
7.4	Objective function of global matching . . . . .	126
7.5	Gradient method for minimization . . . . .	129
7.6	Select unique matches . . . . .	131
7.7	Results . . . . .	133
7.8	Conclusions . . . . .	135
<b>8</b>	<b>General Conclusions and Discussions</b>	<b>143</b>
8.1	The problem . . . . .	143
8.2	Our solutions . . . . .	143
8.2.1	Segmentation . . . . .	143
8.2.2	Local matching . . . . .	144
8.2.3	Global matching . . . . .	146
8.3	Future work . . . . .	146
	<b>Bibliography</b>	<b>148</b>

---

<b>Samenvatting</b>	<b>163</b>
<b>Acknowledgements</b>	<b>167</b>
<b>Curriculum Vitae</b>	<b>169</b>
<b>List of Publications</b>	<b>171</b>

# Summary

Two eyes or cameras looking at the same objects from different perspective provide the means to determine three-dimensional shape and position. Scientific investigation of this effect is called variously *stereo vision*, *stereopsis*. Stereo is an important method for machine perception because it leads to relatively direct measurements and, unlike monocular techniques, does not infer depth under weak and un-verifiable photometric and statistical assumptions, nor does it require specific detailed models of objects.

Segmentation and stereo matching are the two essential steps in stereo vision and are both difficult problems to solve. In the traditional paradigm, segmentation and stereo matching are usually treated separately. The segmentation (we only deal with region-based segmentation in this thesis) aims to partition an image into regions which should be uniform and homogeneous with respect to some characteristics such as grey tone or texture. Segmentation techniques are basically ad hoc and differ in the way they emphasize one or more of the desired properties and in the way they balance and compromise one desired property against another. Image regions are not only caused by the geometric properties of surfaces of the objects, but also by the optical properties of surfaces and the direction of the sun, as well as shadows, etc. Different considerations of image models and different implementations lead to many segmentation techniques. It is observed by many researchers that no single method can provide a complete segmentation of an image. However, each method may provide a subset of the information necessary to produce a more meaningful interpretation of the scene. It is reasonable to expect that there will be complications in fusing the results from different sources, such as from stereo images as we propose in this thesis. Combining results from stereo images needs stereo matching, which, on the other hand, requires good segmentation. Imperfect segmentation and possible occlusion make it difficult to compare regions of two images projected from the same object. In order to get a good matching result, it is necessary to re-segment these regions after matching. Based on the above observations, we propose a new scheme to integrate segmentation and stereo matching. After an initial region-based segmentation, a candidate stereo matching is carried out, which assigns the corresponding regions from one

image to the other image by shape-based matching. During the next step of segmentation, stereo information is included, that is, in considering the merging of one region with its neighboring region, the corresponding regions in the candidate pools are extracted and the matching is carried out using intensity and shape information from both images. Finally, global matching is performed which introduces the other matching constraints such as uniqueness, ordering and topological relations.

This thesis can be roughly divided into three parts: segmentation, local matching and global matching.

## **Part I: Segmentation**

Chapter 2 and 3 concern with the segmentation problem. We use the Minimal Description Length (MDL) principle to encode image intensities and introduce a robust method to estimate the image intensity model from raw image data. We also develop an approach for shape constrained image segmentation which integrates region growing and region boundary fitting. The shape model used is a closed polygon depicted by a number of smooth segments. In order to get shape constraints, we introduce a method for optimal curve fitting which combines the technique of detecting extreme points and a splitting process.

## **Part II: Local matching**

Chapter 4 and 5 deal with local matching in which the local attributes of image regions are considered only. In order to get precise matching of the boundaries of stereo regions, a method is described to solve the stereo matching problem of general closed planar curves under possibly imperfect segmentation and occlusions, provided that the camera parameters are known. The method decomposes the parameters related to an object plane, i.e. slant, tilt and scale factor, and uses a histogram technique to estimate these parameters. The parameter estimation is based on the disparity information of the stereo curves. Point correspondence plays an important role in the method. We solve this problem in a dynamic programming style. The final matching is assessed by applying a distance transformation.

Because each region of images is associated with a number of attributes, the problem of aggregating stereo clues needs to be addressed in the local matching. The theory of fuzzy measure and fuzzy integral is used in this thesis. The fuzzy measure and fuzzy integral provide a general framework of representing and aggregating multiple criteria information. We generalize the fuzzy integral following the ideas of Yager to get a variety of aggregation functions. A major problem associated with the usage of the fuzzy

measure and fuzzy integral is the proper identification and learning of the fuzzy measure. We develop a learning procedure to get precise formula for the fusion purpose.

Chapter 6 contains a detailed discussion on the integration of segmentation and stereo matching. The introduction of feedback enables the proposed system to send back the matching result to a resegmentation stage to improve the segmentation result and consequently the matching result.

## **Part III: Global matching**

The local matching only compares the local attributes of stereo regions. It is possible that there are multiple matches from the local matching, i.e., one region in the left image can have more than one region assigned to it in the right image. The goal of the global matching discussed in chapter 7 is to get unique matches, taking care of topological and other matching constraints. We formulate the global matching as a relaxation problem. We modify the traditional relaxation method to meet our requirements: 1) the support function is soft-thresholded in order to suppress any bad matches and promote possible good matches; 2) a balance factor is introduced to control between the topological constraints and the initial probability from local matching. In this chapter, the final results of this study are presented.



# Chapter 1

## Introduction

### 1.1 General problems of computer vision

Computer vision is concerned with the construction of explicit, meaningful descriptions of physical objects from images. Computer vision systems are artificial vision systems that are capable of understanding indoor or outdoor scenes at a human level. The final aim of a computer system should be to be able to look at an image and describe the scene depicted in words such as, "There is a tall oak tree to the left of a colonial-style house, and a blue Volvo is visible in the garage on the right side of the house." The central problem of computer vision is [1]:

*From one or a sequence of images of a moving or stationary object or a scene, taken by a monocular (one eye) or polynocular (many eyes) moving or stationary observer, to understand the object or the scene and its three-dimensional properties.*

The reader will immediately observe that all the terms in the above definition are well defined, with the exception of the term "understand". What is really the meaning of "understand" with respect to this problem? The problem of finding the meaning of meaning is the the central one in artificial intelligence and it is by no means answered. For this reason, because various researchers understand meaning in different ways, there have basically been two schools of thought in computer vision: *Reconstruction* and *Recognition*.

The reconstruction school worries about the reconstruction of the physical parameters of the visual world, such as the depth or orientation of surfaces, the boundaries of objects, the direction of light sources and the like. The recognition school worries about the recognition or description of objects that we see and involves a process whose end product is some piece of behavior like a decision or a motion. Both schools have strong ties with psychology and neuroscience and it is believed by many at this point that both schools will merge into a new one that will, it is hoped, find an answer to the difficult questions of the

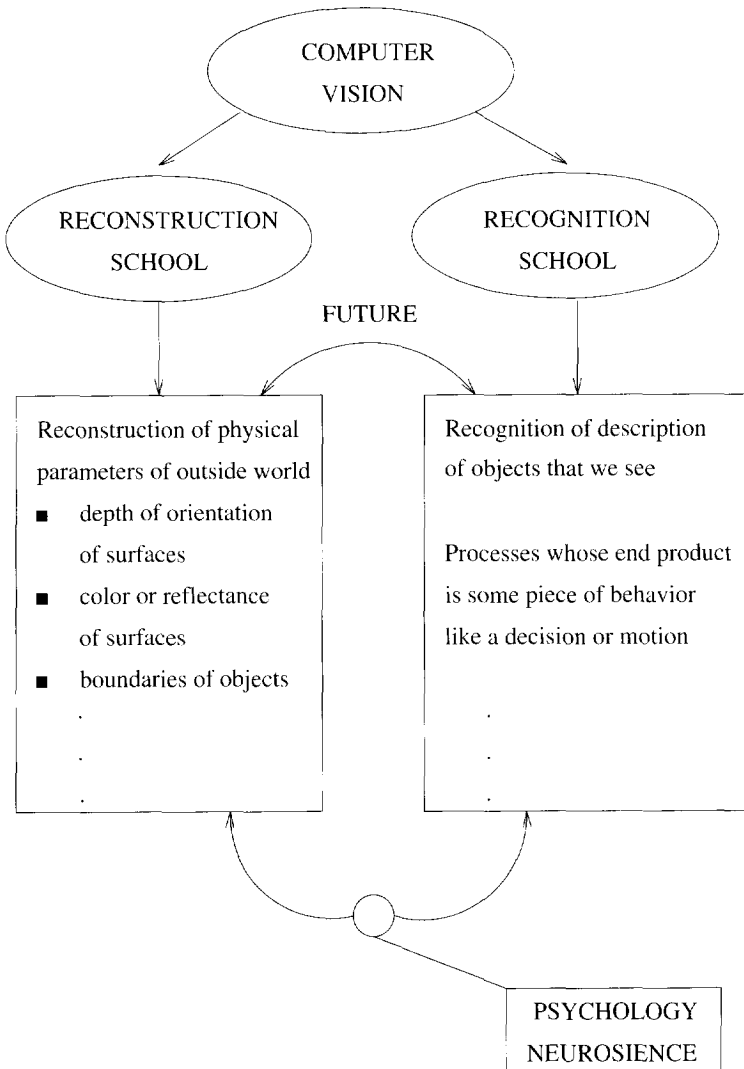


Figure 1.1: The two schools in computer vision [1].

vision problem. Fig.1.1 shows a schematic description of the above-mentioned schools of thought. The recognition school is goal directed, and from this point of view it might be classified, in a sense, as a bottom-up Marr paradigm school, while the reconstruction school can be considered as top-down in the Marr paradigm [2].

As used here, a top-down methodology (top-down in the Marr paradigm) refers to an approach which starts with the development of a general theory for some visual process. The hope is that the theory will provide useful insights into a number of specific applications. This approach has the advantage that it is not tied to any particular application. The corresponding disadvantage is that, because the framework with which the theory is developed is always a simplification of the real world, theoretical development may not lead to practically usable techniques. An example of such a situation is the field "shape from motion", which is well developed theoretically but has found little practical application because the algorithms tend to be extremely sensitive to inaccuracy in the input, and it has proven difficult to obtain motion estimates from real images sequences which are sufficiently accurate to yield usable results.

A bottom-up approach, on the other hand, starts by developing systems which actually perform certain practical tasks. Here the hope is that commonalities observed among several such systems will allow the eventual formulation of more principles. This approach has been criticized on the grounds that it will produce results of too narrow a scope, and without adequate theoretical foundation. On the other hand, if it produces anything at all, it is guaranteed to produce results which can actually be applied. Moreover, a solution to a specific problem can certainly have a solid theoretical foundation within the domain of its applicability. The success of such an approach depends on the appropriate selection of the specific problems. In particular, the problems must have a kind of environmental invariance which makes their solution applicable in a wide range of situations.

In this thesis, we concentrate on the stereo vision problem of computer vision. In the next section, we briefly discuss the various aspects of the stereo vision.

## 1.2 The stereo vision problem

### 1.2.1 General description of the problem

Two eyes or cameras looking at the same objects from different perspective provide the means to determine three-dimensional shape and position. Scientific investigation of this effect (called variously *stereo vision*, *stereopsis*) has a rich history in psychology, biology, and, more recently, in the computational modeling of perception. The human visual system is able to perceive three-dimensional relationships effortlessly, but the means by which one does so are largely unknown and hidden from introspection. Stereo

vision, however, is one way to perceive depth that is relatively well understood from a computational standpoint. Stereo is an important method for machine perception because it leads to relatively direct measurements and, unlike monocular techniques, does not infer depth from weak and un-verifiable photometric and statistical assumptions, nor does it require specific detailed models of objects [3]. Once two stereo images are brought into point-to-point correspondence, recovering depth by triangulation is straightforward.

Many computer models of stereo vision have been proposed. In some cases these models have been offered with a theory of human stereo vision; in other cases the models have been pragmatically task-oriented and unconstrained by knowledge of biological vision. Generally, any stereo vision should confront to a greater or lesser degree the following issues:

- **image acquisition.** Stereoscopic images can be recorded in a large variety of ways. For example, they may be recorded either simultaneously or in a time sequence of any duration; from very similar or from very different perspectives; with accurate, well calibrated instruments or with a crude hobbyist camera.
- **camera modeling.** In order to calculate depth from the disparity, one must have full knowledge of the geometric arrangement of the camera, which means that one must know the *interior orientation* of each camera (the relationship between points in the image and the coordinate system of the camera) and the *exterior orientation* of the two-camera system (the relationship between the local camera coordinate systems and an invariant world coordinate system). A stereo camera model is normally separated into an *absolute* component that specifies the transformations between a camera coordinate system and a world coordinate system, and a *relative* component that specifies the transformations between the coordinate systems of two cameras without reference to a world coordinate system. Knowledge of the relative camera model alone is sufficient to exploit the epipolar constraints to simplify matching. The absolute model is required to translate disparity measurement into absolute measurements of the positions of points in the world.
- **preprocessing.** Preprocessing is typically a linear filtering operation in which the raw image intensities are convolved with one or more digital kernels. In some cases, one can construct a series of images restricted to different spatial frequencies. The lower-frequency images can be subsampled into smaller images to create a pyramid-like resolution hierarchy. Such hierarchies are extremely useful because the range of possible disparity values across the field of view decreases linearly with the sample spacing, which implies that the number of possible matches is relatively smaller at lower resolutions. Coarse matching can be done quickly at low resolution

and the result used to initialize the search at the next finer scale.

- **feature extraction.** Features can be used either as elementary tokens for matching or as “interest operators”, indicating the locations of promising correlations windows. Many kinds of features have been used:
  - Isolated point-like features can be found by a variety of methods, often modeled along the lines of Moravec’s interest operators [4], which select points where there is high variance in four directions.
  - Edge features can be detected by a variety of methods. One commonly used method is to select *zero crossings* on the second derivative of the images;
  - High-level features, such as line junctions, specifically shaped “primitives,” and regions.
- **image matching.** The matching step – solving the correspondence problem – is clearly a critical component of any stereo approach. This issue is further discussed in the following subsections.
- **depth determination.** Once accurate matches have been found, the determination of distance is a relatively straightforward matter of triangulation using the absolute camera model.
- **interpolation.** Stereo applications usually demand a dense array of depth estimates. A lot of matching methods can often only produce sparse correspondences. Consequently, an interpolation step is required.

There are many existing and potential applications of stereo vision, but perhaps the two mostly important are photogrammetry and robot vision. The current state of the art of computational stereo is most relevant to photogrammetry applications, where real-time performance is not required.

In photogrammetric practice, a human analyst uses a stereo analytic plotter to develop a digital elevation model (DEM) from stereo images. This work is tedious, slow and costly. Computational methods that automate stereopsis, sometimes implemented in special-purpose hardware or on massively parallel computers, are now practical to use in this important application.

Stereo vision would obviously be very useful for robots, either as a straightforward mensuration device in a simple industrial robot or, at the other extreme, as part of an integrated perceptual system in an autonomous vehicle. The passive nature of stereo vision is highly desirable for some applications. Because these applications demand

real-time performance, practical robot stereo vision awaits the development of faster and cheaper hardware.

A general formalization of stereo vision is presented by Wolff [5]. There is a good survey paper done by Lemmens [6] (some text and references are quoted in the following discussion of matching methods). In the next section, we review the methods used in solving the stereo matching problem.

### 1.2.2 Signal-based matching

In this category of matching methods, image pixels (signals) are directly used. The simplest method is to take some statistical measure (e.g. cross-correlation) to shift the target area (a group of pixels within a window) defined in the one image, over a search space in the other image, and to compute the correlation function. The size of the search space depends on knowledge of the camera geometry and a priori knowledge on disparity. The central pixel of the target area and that of the most similar search area are taken as corresponding points. The similarity measure should exceed a threshold.

Another approach is to formulate the problem as a least-square problem, i.e. to treat the geometrical transformation and the possible radiometric transformation as the unknown variables. With this method, a more accurate geometric transformation can be produced.

The currently popular method in photogrammetry is known as object-space-based matching. This method emerged originally from the task of reconstructing a digital terrain model from a pair of digital images, independently developed by Wrobel [7,8] and Helava [9]. Helava used the concept of "groundel" as a unit in object space similar to the "pixel" in the image space. The image intensities corresponding to each groundel can be analytically computed, if all pertinent geometric and radiometric parameters (including groundel reflectance, etc.) are known. A least-squares method is adopted to determine a set of unknown quantities or improvements to their approximate values used in the analytical prediction process.

Some references on signal-based matching are [10–15]. Gruen and Baltsavias [16] use multiple views to incorporate geometrical constraints. Rosenholm [17] introduces multiple point matching.

All methods based on signal matching suffer from the following limitations [6, 18]:

- the presence of detectable texture or edges is required; in areas with a smooth gray value function no optimal match will appear;
- repetitive micro structure will cause there to be several equally likely matches;
- linear edges will cause there to be many pronounced matches along the edges;

- surface discontinuities can't be handled;
- the target area may have no counterpart in the search space because of e.g. occlusion;
- sensitive to absolute intensity, contrast, and illumination.
- computationally expensive;
- not rotational and scale invariant.

### 1.2.3 Point-based and edge-based matching

The environment of an interesting point is characterized by a steep autocorrelation function in all directions, high variances in all directions and steep gradients in all directions. These properties lead to many approaches. Autocorrelation considerations are used by Hannah [10] for signal matching purposes. The statistical variance view leads to the approaches of Moravec [4] and Förstner [19]. The gradient view has led to the surface curvature examination of Deschler [20] and Dreschler and Nagel [21].

There are quite a lot of articles dealing with curve-based matching. A review of the methods is given in Chapter 4.

### 1.2.4 Region-based matching

The current author [22, 23] has developed a method using straight lines as primitives and carrying out matching in image/object dual spaces. After carefully examining the method, the writer found that it has several disadvantages:

- difficulties with line detection, another old problem in computer vision. Although tremendous effort has been made, no single method can be found that it is perfect for all kinds of images;
- when using line features, the image intensity is lost. Although some properties of image intensity near the lines (such as contrast) can be used, the internal intensity between the lines can't be employed;
- due to imperfect line detection and following stereo matching, the reconstructed 3-D line segments are not complete. Therefore it is very difficult to reconstruct the whole 3-D scene which includes the surface boundaries and the surface regions.

However, there are a number of advantages to use regions in stereo matching:

- the number of primitives to be considered are fewer than that of line features;

- they are less sensitive to noise;
- much geometric structural information about the scenes is contained in the regions.
- image intensity is included.

In this thesis, we will focus on the usage of region features in stereo matching. Region-based matching requires good segmentation. Because of imperfect segmentation and occlusion, the shapes of regions in two images projected from the same object may be different. In order to get a good matching result, it is necessary to resegment these regions after matching. We analyze this problem in Chapter 6.

There are relatively few references with the idea of using segmented regions in the matching [24–27].

### 1.2.5 Structure-based matching

In addition to descriptive attributes, structure-based matching or relational matching takes also relationships between the phenomena into account. Therefore, a set of elementary tokens, like points, blobs, line fragments and regions, are detected and characterized by attributes, like length, area, shape and average gray value. To these relationships, which describe not only the spatial connections, also attributes which describe their properties [28], the primitives form an entity. It is the task of matching to find a mapping of the primitives of one entity to those of the second entity that best perceives the characteristics and relationships. The basis of entity description is found in graph theory. The description yields long lists. With the aid of searching algorithms, like backtracking tree search or one of its variants, the best mapping of the different viewpoints of stereo images can be obtained, the same object structure has different graphs in both images. Thus, the matching procedures must tolerate these differences which demands *inexact matching techniques*. Shapiro and Haralick [28] formulate the concepts of exact and inexact matching of graph representation of structures.

In [29] Boyer developed an entropy-based approach to the selection of the most effective attributes for primitive descriptions from a larger starting set. The information-theoretic approach to inter-primitive matching should be superior to some obvious techniques, such as representing each primitive as a vector in attribute space and computing Euclidean distances, because the probabilistic nature of the situation is considered. Further, for attributes which take on symbolic values, it may be difficult or impossible to order the values in such a manner as to make the Euclidean distance meaningful.

One of the most difficult but interesting problems is optimum matching between structural descriptions, i.e., the problem of finding the correspondence between their parts

in order to make the corresponding property and relations as consistent as possible. A great deal of material dealing with this problem has been published. You [30] employed a combinatorial (tree search) approach and applied the branch-and-bound algorithm to this problem. The algorithm always gives the true optimum match; however, it may be impractical in analyzing highly complex structures because of its inherent combinatorial property. Tsai and Fu [31] also described a tree search technique for finding isomorphisms between graphs which bear both symbolic and numerical labels. Kichen [32] solved this problem in both qualitative and quantitative cases by using a relaxation method. The method is tolerant to noise, and it can take advantage of hardware parallelism. However, the relaxation method is inherently a local optimization method, and it is sometimes difficult to determine a suitable updating rule (the local compatibility function). More articles tackling this problem can be found in [33–36].

### 1.2.6 Other matching methods

More references on matching can be found in [16, 37] (dealing with geometric constraints), [38] (Hough transform-based matching), [37, 39] (perceptual grouping), [40] (error detection), [41] (model-based matching).

## 1.3 Fusion of vision modules

Because recovery of information about the world from a single cue such as motion parallax, binocular stereo disparity or shading in images, tends not to be very robust, there is a great deal of interest in integrating information from multiple cues. The integration of early vision modules will be required for most practical applications of vision systems. In the simplest case, this means interlacing iterations of different schemes for recovering shape, or more formally, constructing a compound functional that contains penalty terms for mismatching information available from all cues being considered. The new approach is particularly attractive because it suggests a systematic methodology for this integration, enabling new cues to be included easily.

### 1.3.1 Shape from shading and motion

Most of the work in shape from shading [42–44] assumes that the albedo of the surface in view and the illuminant direction are known *a priori*; in other words, this work assumes that the reflectance map specifies how the brightness of a surface patch depends on its orientation, under given circumstances. It therefore encodes information about the reflecting properties of the surface and information about the distribution and intensity

of the light sources. In fact, the reflectance map can be computed from the bidirectional reflectance-distribution function and the light source arrangements, as shown by [45].

When encountering a new scene, we usually do not have the information required to determine the reflectance map. Yet, without this information, we are unable to formulate the shape from shading problem.

The dilemma may be resolved if a calibration object of known shape appears in the scene, since the reflectance map can be computed from its image, but what happens when we are not that fortunate? It is evident from the above discussion that at least knowledge of the illuminant direction is required. Work has been done on illuminant direction determination [46–48]. Pentland's method is based on the assumption that surface orientation, when considered as a random variable over all possible scenes, is isotropically distributed. A consequence of this assumption is that the change of surface normals is also isotropically distributed. Under his assumptions, Pentland solves the problem uniquely, but his assumptions are restrictive.

Brooks and Horn [47] presented a method in the general framework of ill-posed problems and regularization in early vision. Their theory proposes to solve the shape from shading problem and at the same time to compute the illuminant direction, by minimization of an appropriate functional. Uniqueness or convergence proofs of their iterative methods were not presented, but their experimental results for synthetic images were promising.

Finally, the method described in [48] is based on Lambertian reflectance and a Hough transform technique, attempted the recovery of the direction of the light source.

### 1.3.2 Shape from texture

The problem of shape from texture has received a lot of attention in the past few years and some good research on the topic has been published [49–53]. The problem is defined as “finding the orientation of a textured surface from a static monocular view of it.” This problem is ill-posed in the sense that there are infinitely many solutions. To restrict the space of solutions, assumptions such as directional isotropy and uniform density have been employed in previous research.

### 1.3.3 Shape and 3-D motion from contour and stereo

The recovery of three-dimensional shape and surface orientation from a two-dimensional contour is a fundamental process in any visual system. Recently, a number of methods have been proposed for computing shape from contour. For the most part, previous techniques have concentrated, according to the Marr paradigm, on trying to identify a few simple, general constraints and assumptions that are consistent with the nature of all

possible objects and imaging geometries in order to recover a single “best” interpretation, from among the many possible for a given image. For example, Kanade [54] defines shape constraints in terms of image space regularities such as parallel lines and skew symmetries under orthographic projection. Witkin [55] looks for the most uniform distribution of tangents to a contour over a set of possible inverse projection in object space under orthography. Similarly, Brady and Yuille [56] search for the most compact shape (using the measure of area over perimeter squared) in the object space of inverse projected planar contours.

## 1.4 Scope of this thesis

Segmentation and stereo matching are both difficult problems in computer vision. In the traditional paradigm, segmentation and stereo matching are usually treated separately. The segmentation (we only deal with region-based segmentation in this thesis) aims to partition an image into regions which should be uniform and homogeneous with respect to some characteristics such as gray tone or texture. Segmentation techniques are basically ad hoc and differ precisely in the way they emphasize one or more of the desired properties and in the way they balance and compromise one desired property against another. Image regions are not only caused by the geometric properties of surfaces of the objects, but also by the optical property of surfaces and the direction of the sun, as well as shadows. Different considerations of image models and different implementations lead to a variety of segmentation techniques. It is observed by many researchers that no single method can provide a complete segmentation of an image. However, each method may provide a subset of the information necessary to produce a more meaningful interpretation of the scene. It is reasonable to expect that there will be complications in fusing the results from different sources, such as from stereo images as we propose in this thesis. Combining results from stereo images needs stereo matching, which, however, requires good segmentation. Imperfect segmentation and occlusion make it difficult to compare regions of two images projected from the same object. In order to get a good matching result, it is necessary to resegment these regions after matching. Based on the above observations, we propose a new scheme to integrate segmentation and stereo matching. After an initial region-based segmentation, a candidate stereo matching is carried out, which assigns the corresponding regions from one image to the other image by shape-based matching. During the next step of segmentation, stereo information is included, that is, in considering the merging of one region with its neighboring region, the corresponding regions in the candidate pools are extracted and the matching is carried out using intensity and shape information from both images. Finally, global matching is performed which introduces the other matching

constraints such as uniqueness, ordering and topological relations.

In Chapter 2, we first describe the Minimal Description Length (MDL) principle and how to use MDL to encode image intensities. We also introduce the method of modeling region intensities and estimating the model from raw image data. Finally the region growing algorithm and experimental results are given.

Chapter 3 presents an approach to shape-constrained image segmentation which integrates region growing and region boundary fitting. The shape model used is a closed polygon depicted by a number of smooth segments. The MDL principle is also used to encode the shape of the regions. In this chapter, we also introduce a method for optimal curve fitting which combines the techniques of detecting extreme points and the splitting process.

In Chapter 4, a method is described to solve the stereo matching problem of general closed planar curves under possibly imperfect segmentation and occlusions, provided that the camera parameters are known. Our method decomposes the parameters related to an object plane, i.e. slant, tilt and scale factor, and uses a histogram technique to estimate these parameters. The parameter estimation is based on the disparity information of the stereo curves. Point correspondence plays an important role in the method. We solve this problem in a dynamic programming style. The final matching is assessed by applying a distance transformation. The method has been applied successfully to several practical examples.

In Chapter 5, the problem of aggregating stereo clues is addressed, using the theory of fuzzy measure and fuzzy integral introduced by Sugeno [57] [58]. The fuzzy measure and fuzzy integral provide a general framework of representing and aggregating multiple criteria information. We generalize the fuzzy integral following Yager's ideas [59] [60], to get a variety of aggregation functions. A major problem associated with the usage of a measure and fuzzy integral is the proper identification and learning of the fuzzy measure, which is also treated in this chapter. Finally, we formulate the stereo clues and fusion process under the developed scheme, and some practical examples are demonstrated.

In Chapter 6, the method for the integration of segmentation and stereo matching is proposed, which uses the feedback control principle from system theory. The introduction of feedback enables the proposed system to send back the matching result to a resegmentation stage to improve the segmentation result and consequently the matching result. A general discussion on existing control methods in computer vision is given first in this chapter. The system layout, detailed description, associated algorithms, and related experimental results on integrating segmentation and matching follow.

Chapter 7 deals with the global matching problem. The local matching only compares the local attributes of stereo regions. It is possible that there are multiple matches from

the local matching, i.e., one region in the left image can have more than one region assigned to the right image. The purpose of the global matching is to get unique matches, taking care of topological and other matching constraints. We formulate the global matching as a relaxation problem. We modify the traditional relaxation method to meet our requirements: 1) the support function is soft-thresholded in order to suppress any bad matches and promote possible good matches; 2) a balance factor is introduced to control between the topological constraints and the initial probability from local matching. In this chapter, the final results of this thesis are presented.

In Chapter 8, we conclude our thesis by summarizing our work, the contribution and some future perspectives.



## Chapter 2

# MDL-Based Region Growing

### 2.1 Introduction

Image segmentation is one of the fundamental problems in computer vision. The techniques for image segmentation roughly fall into two general categories:

- Edge-based methods. In this category, any one of a group of operators is applied to the raw images, yielding primitive edge elements, followed by a concatenating procedure to make a coherent one-dimensional feature from many local edge elements.
- Region-based methods. These methods usually depend on pixel statistics over localized areas of the image. Regions of an image segmentation should be uniform and homogeneous with respect to some characteristics such as gray tone or texture. Adjacent regions of a segmentation should have significantly different values with respect to the characteristics [61].

In this thesis, we concentrate on the region-based methods. The existing region-based image segmentation techniques can be classified as: measurement space guided spatial clustering, single linkage region schemes, hybrid linkage region growing schemes, centroid linkage region growing schemes, spatial clustering schemes [62], and split and merging schemes [63–74].

Recently, there has been increasing interest in applying information theory to automatically interpret and analysis image data [75–82]. The fundamental concept in information theory is the idea that the amount of information derived from some event, or experiment, is related to the number of degrees of freedom available beforehand, or the reduction in uncertainty about some other event gleaned from an observation of the outcome. Among

other tools provided by information theory, the Minimum Description Length (MDL) principle has been quite successfully applied to the field of computer vision.

The MDL principle studies estimation based upon the principle of minimizing the total number of binary digits required to rewrite the observed data, when each observation is given with a certain precision. Instead of attempting at an absolutely shortest description, which would be futile, the MDL principle looks for the optimum relative to a class of parametrically given distributions. The MDL principle turns out to degenerate to the more familiar Maximum Likelihood (ML) principle when the number of parameters in the models is fixed, so that the description length of the parameters themselves can be ignored. In another extreme case, where the parameters determine the data, it similarly degenerates to Jaynes' principle of maximum entropy. The main power of the MDL principle is that it permits estimates of the entire model, its parameters, their number, and even the way the parameters appear in the model; i.e., the model structure [83–85].

There are already a number of articles using the MDL principle for image analysis. Darell [86] formulated the segmentation task as a search for a set of descriptions which minimally encodes a scene. A new framework for cooperative robust estimation is used to estimate descriptions that locally provide the most saving in encoding an image. A modified Hopfield-Tank network finds the subset of these descriptions which best describes an entire scene, accounting for occlusion and transparent overlap among individual descriptions.

Fua and Hanson [76–78] applied the MDL principle to extract buildings from images. Their interpretation consists of two steps: 1) derive a set of likely hypotheses of image descriptions using search or estimation techniques. Here all available knowledge of the type of objects and of efficient strategies may be explored; 2) choose the best competing hypotheses based on the simplicity of the description. The simplicity or likelihood is measured by the number of bits necessary to describe a specific realization of the model and the deviation of the actual image from the ideal model.

Leclerc has done some elegant work on image segmentation using the MDL principle. In 1989, he [79] presented a hierarchical optimization approach to the image partitioning problem: that of finding a complete and stable description of an image, in terms of a specified descriptive language, which is simplest in the sense of shortest description length. The first stage in the hierarchy uses a low-order polynomial description of the intensity variation within each region and a chain-code-like description of the region boundaries. By using a regular-grid finite element representation for the image, the optimization technique, called a continuation method, reduces to a simple, local, parallel, and iterative algorithm that is ideally suited to the Connection Machine. In his later work [80], he added another layer (region grouping) which groups together regions belonging to a single surface.

One possible basis is the "good continuation" of segments of region boundaries. A second possible basis is the "good continuation" of the intensity variation within the regions. The interpretation of good continuation means that the intensity variation within a group of regions is simpler to describe using a single polynomial model than with independent polynomials (one per region) originally recovered by the segmentation algorithm.

Another relevant contribution was done by Keeler [87] who regarded a "segmentation" as a collection of parameters defining an image-valued stochastic process by separating topological (adjacency) and metric (shape) properties of the subdivision and intensity properties of each region. The priori selection is structured accordingly. The novel part of the representation, the subdivision topology, is assigned a priori by universal coding arguments, using the minimum description-length philosophy that the best segmentation allows the most efficient representation of visual data.

In the next sections, we will first describe the MDL principle (Section 2.2 and how to use MDL to encode image intensities (Section 2.3). We will also introduce the method of modeling region intensities and estimating the model from raw image data (Section 2.4). Finally the region growing algorithm and experiment results will be given (Section 2.5 and 2.6).

## 2.2 Minimum description length

The theory of information was developed by C. Shannon for analyzing communication systems. Specifically it deals with measuring the information content of a message and the efficiency of sending the message over a channel which possibly is noisy. According to Shannon a discrete information source can be modeled as a Markov-Process, which randomly selects letters out of a prespecified alphabet. The information, which is transmitted per letter, is the larger the less likely the letter is selected and can be interpreted as the degree of surprise when the letter reaches the receiver or as the uncertainty when no knowledge about the letter is available.

In the most simple case the transmitted letters are independent. Let  $P(a = w_i)$  be the probability that the letter  $\bar{a}$  (a random variable) is equal to the value  $w_i$ . Then the gain of information when being told  $w_i$ , i.e. the *information* of  $w_i$  is

$$I(\bar{a} = w_i) = I(w_i) = \log \frac{1}{P(w_i)} = -\log P(w_i) \quad (2.1)$$

If the logarithm is taken to basis 2, the unit of information is the "bit".

We want to introduce the principle of minimum description length encoding using a simple example similar to the one given by Förstner [88]. Let  $n_0$  points  $(x_i, y_i)$  in a plane be given as in Fig.2.1a. The scope is to explain the data in the most intuitive manner. The

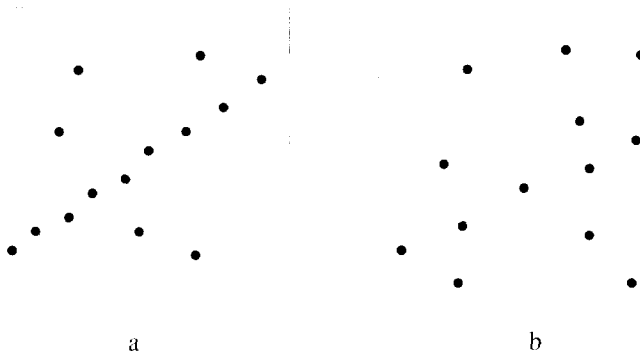


Figure 2.1: 14 points within a square, most likely interpretations: a) 9 points on a straight line and 5 outliers; b) random set or 5 points on a straight line and 9 outliers?

figure suggests the larger number  $n = 9$  of the  $n_0 = 14$  points to approximately sit on a straight line, while the other  $n = n_0 - n = 5$  points do not belong to this line. Fig.2.1b shows a different pattern, where we are not sure where we should assume the 5 points in the middle of the figure to belong to a straight line or whether we then should treat the figure as consisting of 14 randomly distributed points or even 3 vertical nearly straight lines.

The situation is representative of a large class of interpretation tasks:

- We have to deal with several competing hypotheses which have a different structure.
- We have to deal with a significant amount of spurious data.
- There may be no explanation of the data within the assumed set of hypothesis.

The problem of explaining the data sets in Fig.2.1 lies in the fact that the pure fit between a selected number of data points and a set of hypothesized straight lines, say, is not sufficient as a quality measure, as this fit can be made perfect by restricting it to just 2 data points or by increasing the number of postulated lines. Therefore the evaluation of an explanation has to balance the fit between data and model and the complexity of the model. The principle of description length encoding fulfills these requirement.

We want to derive the description lengths in bits for the case when no model is assumed with the case when the data essentially are assumed to consist of points sitting approximately on a straight line admitting some outliers.

Let the coordinates be given up to a resolution of  $\epsilon$  (e.g. 1 pixel) and be within a range  $R$  (e.g. 256 pixel). Then  $\log_2(R/\epsilon)$  bits are necessary to describe one coordinate. The

<sup>1</sup> $\log$  = logarithm with basis 2.

description length for the  $n_0$  points, when assuming no model, therefore is

$$\Phi_0 = \text{\#bits (points | no model)} = n_0 2lb(R/\epsilon) \quad (2.2)$$

thus  $2 \cdot n_0 \cdot 8 = 16n_0$  in the case of  $n_0$  points in a  $256 \cdot 256$  pixel image or 224 bits on the plot of Fig.2.1.

The deviations of the points from the straight line are considered from a normal distribution, the probability of an element with deviation  $r$  is

$$\begin{aligned} P(r) &= \int_{r_0}^{r_0+1} \frac{1}{\sqrt{2\pi}\sigma^2} \exp\left[-\frac{x^2}{2\sigma^2}\right] dx \\ &\simeq \frac{1}{\sqrt{2\pi}\sigma^2} \exp\left[-\frac{r^2}{2\sigma^2}\right] \end{aligned} \quad (2.3)$$

where  $r_0$  is the integer such that  $r_0 \leq r < r_0 + 1$ , and the approximation is valid provided  $\sigma \gg 1$ . The total cost of encoding the  $n$  pixels within the Gaussian peaks is then

$$\begin{aligned} C &= \sum -lb \frac{1}{\sqrt{2\pi}\sigma^2} \exp\left[-\frac{r^2}{2\sigma^2}\right] \\ &= n \frac{lb2\pi}{2} + nlbe(lb_e\sigma + \frac{1}{2\sigma^2} \sum \frac{r^2}{n}) \end{aligned} \quad (2.4)$$

If we assume  $n$  points to sit on a straight line and the other  $n = n_0 - n$  points to be outliers we need

$$\begin{aligned} \Phi_m &= \text{bits (points | 1 straight line)} \\ &= n_0 + n \cdot 2lb(R/\epsilon) + \\ &\quad + \left[ nlb(R/\epsilon) + \sum_{i=1}^n \left\{ \frac{1}{2ln2} \cdot \left(\frac{v_i}{\sigma}\right)^2 + lb(\sigma/\epsilon) + \frac{1}{2} lb2\pi e \right\} \right] + 2lb(R/\epsilon) \end{aligned} \quad (2.5)$$

where the first term represents the  $n_0$  bits for specifying whether a point is good or bad, the second term is the number of bits to describe the bad points, the third term is the number of bits to describe the good points and the last term is needed to describe the two parameters of the straight line. We assume the good points to randomly sit on the straight line, which leads to the first term in the brackets, and to have Gaussian distributed deviations  $v_i$  from the line with standard deviation  $\sigma$ .

In the example of Fig.2.1a, with  $n = 9$  and  $n = 5$  we need *on average* need:

$$\begin{aligned} \Phi_m &= n_0 + n \cdot 2lb(R/\epsilon) + n(lb(R/\epsilon) + lb(\sigma/\epsilon) + \frac{1}{2} lb2\pi e) + 2lb(R/\epsilon) \\ &= 14 + 2 \cdot 5 \cdot 8 + 9 \cdot (8 + 1 + 2.04) + 2.8 \simeq 209 \text{bits} \end{aligned} \quad (2.6)$$

to code the point set, when assuming a straight line with outliers. This is less than the 224 bits, thus supporting this explanation. For Fig.2.1b we, however, need 229 bits, assuming 5 points sitting on a straight line, which obviously is no explanation for the data.

In this application there is a close relation to techniques of robust estimation (for more discussion, see Section 2.4): minimizing  $\Phi_m$  from Eq.(2.5) with respect to the parameters of the straight line is identical to minimizing

$$\Phi_m = d + a \sum_{i=1}^{n_0} \rho(v_i). \quad (2.7)$$

with  $a = 1/(2\ln 2)$ ,  $d = n_0(1 + lb(R/\epsilon) + lb(\sigma/\epsilon) + \frac{1}{2}lb(2\pi) + 2lb(R/\epsilon))$  and the optimization function

$$\rho(x) = \begin{cases} k^2 & \text{if } (x/\sigma)^2 \geq k^2 \\ (x/\sigma)^2 & \text{if } (x/\sigma)^2 < k^2 \end{cases} \quad (2.8)$$

and  $k^2 = 2\ln(R/(\sqrt{2\pi} \cdot \sigma))$ , thus equivalent to minimizing

$$\Phi'_m = \sum_{i=1}^n \rho(v_i) \quad (2.9)$$

When replacing  $\rho(x)$  by  $1 - \exp(-(x/\sigma)^2)$ , thus when blending the shoulders, minimizing  $\Phi'$  in Eq.(2.9) is equivalent to re-weighting the residuals with an exponential weight function. The optimization problem formulated there, however, had no link between the number of outliers and the degree of fit, as in Eqs.(2.7) and (2.8).

The balance between model complexity and data fit can be used to derive the *minimum number of good data points* which are *necessary to explain a model*. In our case of one straight line we obtain from  $\Phi_m(n) < \Phi_0$

$$n \geq \frac{n_0 + 2lb\frac{R}{\epsilon}}{lb\frac{R}{\sigma\sqrt{2\pi e}}} \quad (2.10)$$

In our example we obtain  $n > 6$ , again proving that the 5 points in Fig.2.1b, which may seem to lie on a straight line, are not sufficient to support this explanation. For increasing precision, i.e. for decreasing  $\sigma$  (leaving  $R, \epsilon$  and  $n_0$  fixed) we obviously may accept an explanation with fewer data points supporting it.

The example revealed several important properties of the description length encoding principle:

- It is able to compare explanations of different structure, here random data with one line plus outliers.
- It is able to cope with spurious data. Any additional explanation of these spurious data using a simple model would further decrease the description length.

- The decision whether data are spurious or not depends on the model not on some significance level.
- A decision on the *admissibility* of a model or of an explanation is available, rejecting explanations which are too complicated- an extremely useful and necessary property of the theory.
- The principle of minimum description length encoding for fixed model structure reduces to the principle of maximum likelihood and under the Gaussian assumption to the least-squares principle.

## 2.3 Using MDL to encode the intensities

We model the interior intensities of an image region by a smooth intensity surface with a Gaussian distribution of deviations from the surface and a number of possible outliers. Combining the result of the previous section and developments by Hua and Hanson [78], the number of bits to describe image intensities in a region is calculated by

$$L_I = n_1 \left( \sum_{i=1}^{n_1} \frac{1}{2 \ln 2} \left( \frac{v_i}{\sigma} \right)^2 + lb \sigma + \frac{1}{2} lb 2\pi \right) + 8n_2 + \left( n_1 lb \left( \frac{n_0}{n_1} \right) + n_2 lb \left( \frac{n_0}{n_1} \right) \right) + N_p \quad (2.11)$$

Where

$L_I$  is the number of bits to describe the intensity information in a region

$n_0$  is the total number of pixels in one region

$n_1$  is the number of pixels in the Gaussian peak

$n_2$  is the number of outliers

$N_p$  is the number of bits to describe the surface model

$\sigma$  is the standard deviation of intensity noise

In (2.11), the first term  $n_1 \left( \sum_{i=1}^{n_1} \frac{1}{2 \ln 2} \left( \frac{v_i}{\sigma} \right)^2 + lb \sigma + \frac{1}{2} lb 2\pi \right)$  is the cost of Huffman-encoding the pixels in a Gaussian distribution, the second item  $8n_2$  is the cost of encoding the pixel outliers in full 8 bits, the third one  $\left( n_1 lb \left( \frac{n_0}{n_1} \right) + n_2 lb \left( \frac{n_0}{n_1} \right) \right)$  is the entropy for encoding whether or not the pixel is anomalous, and the fourth term  $N_p$  specifies the coding of the model parameters. The surface models and their estimation is addressed in

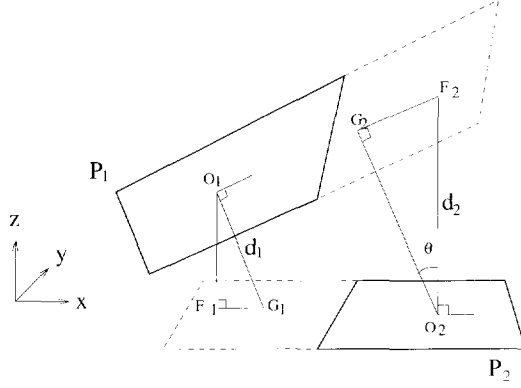


Figure 2.2: The distance between two planes.

the next section; and the usage of the MDL principle in the segmentation is discussed in Section 2.5. In the segmentation, a normalized version of Eq.(2.11) is used, i.e.;

$$L_l = L_l/n_0. \quad (2.12)$$

## 2.4 Surface models and their robust estimation

In our current algorithm, we use a polynomial plane for the underlying model of image intensities within one region. In this section, we first describe the expression of the plane and the method of measuring the distance between two underlying planes of neighboring regions (the distance is used in the segmentation algorithm). We will also present the robust method to estimate the plane from raw image data.

### 2.4.1 Surface models

In Fig. 2.2,  $P_1$  and  $P_2$  are the two underlying planes estimated from two neighboring regions, with  $O_1(x_{o1}, y_{o1})$  and  $O_2(x_{o2}, y_{o2})$  the corresponding region geometric centers.  $F_1$  is the intersection point of plane  $P_2$  with the normal vector of  $P_2$  passing through  $O_1$ ,  $G_1$  the intersection point of plane  $P_2$  with the normal vector of plane  $P_2$  passing through  $O_1$ , similarly with  $F_2$  and  $G_2$ .  $\theta$  is the angle between the two planes.

Denoting the equation of plane  $P_1$  as

$$A_1x + B_1y + C_1z + D_1 = 0, \quad (2.13)$$

and the equation of plane  $P_2$  as

$$A_2x + B_2y + C_2z + D_2 = 0, \quad (2.14)$$

it is easily seen that

$$\overline{O_1 A_1} = \frac{A_2 x_{o1} + B_2 y_{o1} + C_2 z_{o1} + D_2}{\sqrt{A_2^2 + B_2^2 + C_2^2}}, \quad (2.15)$$

$$\overline{O_2 B_2} = \frac{A_1 x_{o2} + B_1 y_{o2} + C_1 z_{o2} + D_1}{\sqrt{A_1^2 + B_1^2 + C_1^2}}, \quad (2.16)$$

$$\cos(\theta) = \frac{A_1 A_2 + B_1 B_2 + C_1 C_2}{\sqrt{A_1^2 + B_1^2 + C_1^2} \sqrt{A_2^2 + B_2^2 + C_2^2}} \quad (2.17)$$

and

$$d_1 = \overline{O_1 A_1} / \cos(\theta) \quad (2.18)$$

$$d_2 = \overline{O_2 B_2} / \cos(\theta). \quad (2.19)$$

The distance between the two planes is defined as

$$d_p = \max(d_1, d_2). \quad (2.20)$$

### 2.4.2 M-estimators

The least-squares and the standard normal theory are very attractive because of their flexibility and wide applicability to complex linear models. They are good provided that the normal distribution is reasonably close to the real problem at hand and when outliers are of little concern. The classical procedures are highly sensitive to the gross errors (i.e. to the outliers and long-tailed distributions): 10% of the outliers with standard deviation  $3\sigma$  contribute a variance equal to that of the remaining 90% of the cases with standard deviation  $\sigma$  [89]. The outliers can double or triple the variance, so that cutting out their effect could really increase the precision. In the light of these facts, we must seek statistical procedures that are good not only for one model but also for a broad class of possible underlying models; they need not necessarily be best for any one of them. Box and Anderson [90] introduced the notion *robustness* as follows: procedures are required which are 'robust' (insensitive to changes in extraneous factors not under test) as well as powerful (sensitive to specific factors under test).

Statisticians have developed various sorts of robust statistical estimators, such as M-estimators, L-estimators, and R-estimators. M-estimators are usually the most relevant class for model-fitting, that is, estimation of parameters. We therefore consider these estimate in a little more detail below.

For data  $(x_i, y_i), i = 1, \dots, N$ , we obtain M-estimators of regression parameters  $\mathbf{a}$  in a model  $y(x; \mathbf{a})$  ( $a = \{a_1, \dots, a_M\}$ ), if we minimize

$$\mathcal{M} = \sum_{i=1}^N \rho(y_i - y(x_i; \mathbf{a})) \quad (2.21)$$

where  $\rho$  is some (usually convex) function.

If we now define the derivative of  $\rho(z)$  to be a function  $\psi(z)$ ,

$$\psi(z) \equiv \frac{d\rho(z)}{dz} \quad (2.22)$$

then the estimators should meet the following conditions

$$0 = \sum_{i=1}^N \psi(y_i - y(x_i)) \left( \frac{\partial y(x_i; \mathbf{a})}{\partial a_k} \right) \quad k = 1, \dots, M. \quad (2.23)$$

We can readily see that the specialization for normal distributed errors is

$$\rho(z) = \frac{1}{2} z^2 \quad \psi(z) = z. \quad (2.24)$$

A distribution with an even more extensive – therefore sometimes even more realistic – tail is the *Cauchy* or *Lorentzian* distribution,

$$\text{Prob}\{y_i - y(x_i)\} \sim \frac{1}{1 + \frac{1}{2}(y_i - y(x_i))^2} \quad (2.25)$$

This implies

$$\rho(z) = \log(1 + \frac{1}{2} z^2) \quad \psi(z) = \frac{z}{1 + \frac{1}{2} z^2} \quad (2.26)$$

In our algorithm, we use the following  $\rho(z)$  function:

$$\rho(z) = \begin{cases} \frac{1}{2} z^2 & \text{if } |z| \leq a \\ a|z| - \frac{1}{2} a^2 & \text{if } a < |z| \leq b \\ \frac{a}{c-b}(c|z| - \frac{1}{2} z^2 + \rho_0) & \text{if } b < |z| \leq c \\ \frac{1}{2}(c + b - a) & \text{if } c < |z| \end{cases} \quad (2.27)$$

where  $\rho_0 = \frac{1}{2}(ab - b^2 - ac)$ , the corresponding  $\psi(z)$  function is

$$\psi(z) = \begin{cases} z & \text{if } |z| \leq a \\ a \cdot \text{sign}(z) & \text{if } a < |z| \leq b \\ \frac{c-|z|}{c-b} a & \text{if } b < |z| \leq c \\ 0 & \text{if } c < |z| \end{cases} \quad (2.28)$$

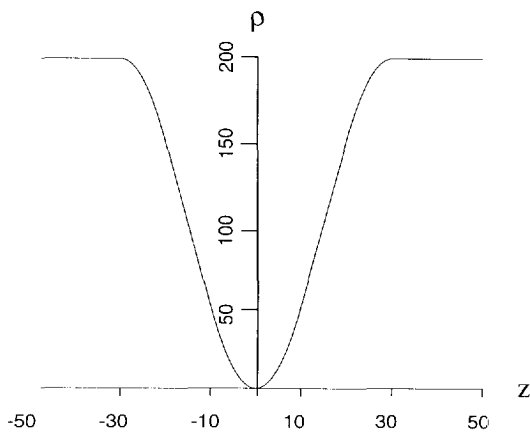


Figure 2.3:  $\rho(z)$  function for Eq.(2.27).

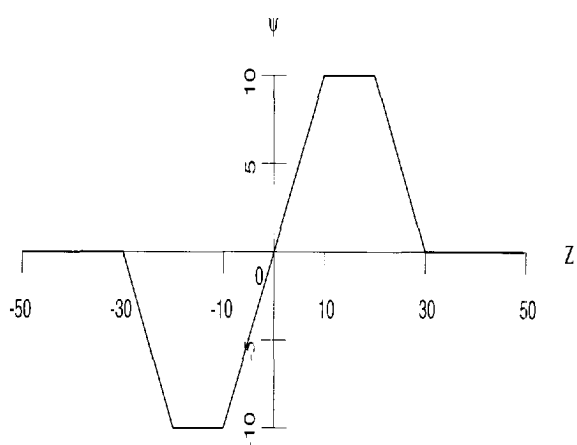


Figure 2.4:  $\psi(z)$  function for Eq.(2.28).

In Fig. (2.3) and (2.4), the  $\rho(z)$  and  $\psi(z)$  with  $a = 10, b = 20, c = 30$  are illustrated.

Note that the  $\psi(z)$  function occurs as a weighting function in the generalized normal equations (2.23). For normally distributed errors, Eq.(2.24) says that the more deviant the points, the greater the weight. By contrast, when tails are somewhat more prominent, as in Eqs.(2.27) and (2.28), then all deviant points get the same relative weight, with only the sign information used. Finally, when the tails are even longer, (2.28) says the  $\psi$  increases with deviation, then starts *decreasing* (see Fig.2.4), so that very deviant points – the true outliers – are not counted at all in the estimation of the parameters.

In our algorithm, we treat the model-fitting problem as an optimization and use the downhill simplex minimization algorithm (for details, see [91], pp.408) to solve the problem as follows.

---

**Algorithm 2.1 :**

1. find the median of the gray values within a region.
  2. if the region area is large enough, do this step:
    - set the vertices of the starting simplex.
    - start the simplex method to minimize the function  $\mathcal{M}$  in Eq.(2.21).
    - after the simplex algorithm is successfully finished, check the distances between the vertices of the final simplex to make sure the reliability of the estimated parameters. If the simplex algorithm is not successfully finished or the parameters are not reliable, do not use a multi-parametered plane model for region intensities, only median is used.
  3. estimate standard deviation and the number of outliers.
- 

## 2.5 Region Growing Algorithm

The region segmentation algorithms in the literature can be divided into three categories: pure merging, pure splitting, and split-and-merge schemes. In the first scheme, the picture is first divided into many small primitive regions (even pixels) which are then merged to form larger regions on the basis of certain homogeneity criteria. In contrast, a pure split method views the entire image as the initial segmentation. Then it successively

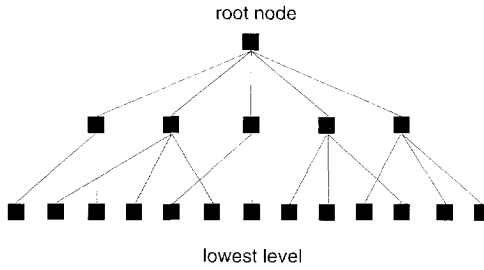


Figure 2.5: N-node Tree data structure.

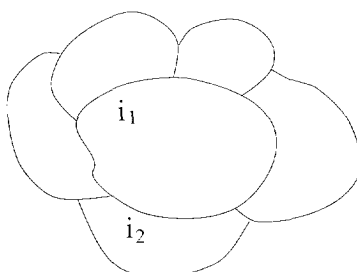


Figure 2.6: Region merging.

splits each current segment if the segment is not homogeneous enough. In the split-and-merge scheme, the efficiency of processing is improved by first partitioning the image into square subregions and then splitting them further if they are not homogeneous. Afterward, a merging process is applied to those adjacent segments that satisfy some uniformity criteria. The pyramid quad tree structure is often employed as the basic data structure. A detailed implementation on the split-and-merge scheme can be found in [92].

For the implementation of our algorithm, the quad tree structure is not attractive because the splitting and merging in this data structure is usually not carried out where the real object region boundary occurs. If we want to include the shape as one important component of region uniformity (we treat shape constraints in the next chapter), we have to find a data structure which can describe each region more directly. Therefore, the "N-node tree" has been developed. An example of an "N-node tree" is given in Fig. 2.5.

One can view an "N-node tree" as a generalization of the quad tree data structure in the sense that while the number of children under a node in the quad tree is fixed to four, the "N-node tree" decides the number of children according to the number of small regions from which the bigger region emerges. In Fig. 2.5, the lowest level of the tree actually corresponds to the original image pixels which are indexed from left to right, top

to bottom in the image, and the top level or root node represents the whole image. The intermediate levels of the tree describe the segmentation results at different stages. One node on each level represents a complete region which has a unique and sequential label on this level. The original image and the segmented regions can be found via tracing through the "N-node tree". In practice, a label image, which has the same size as the original image, is created and is valued by its corresponding label in order to facilitate the indexing.

The whole segmentation procedure is carried out in four phases:

1. Forming the lowest level of the tree. It is a simple indexing which puts the original raster image pixels into the lowest level of the tree.
2. Initial merging based on a statistical test on the image intensity. It is a traditional method used in split-and-merge algorithms, which examines the merging result with the previous small regions using a statistical method to decide whether or not such merging is good. The decision making for merging is shown in Figure 2.6. For region  $i_1$ , in order to find the neighboring region to merge with, all the regions surrounding region  $i_1$  should be examined and the one with closest similarity with region  $i_1$  is selected as the candidate region to merge with (say, e.g. region  $i_2$ ). If the uniformity measurement of region  $i_1$  and  $i_2$  is within a certain threshold, these two regions are merged to form a new region. This step can be regarded as providing hypotheses for possible homogeneous regions.
3. Adaptive merging based on MDL. The decision making procedure for merging is similar to the last step. In this case, a candidate region is chosen if it has the minimal length measure by formula (2.12), if it merges with region  $i_1$  (we still use Fig. 2.6 as example), compared with other regions surrounding region  $i_1$ . The selected candidate region  $i_2$  is merged with region  $i_1$  if the MDL measurement after merging is smaller than the sum of individual MDL measurements from two regions before the merging.

We express such a procedure more rigorously in mathematic formula in the following:

Denoting  $R_i^k$  as the  $i^{th}$  region at  $k$ -stage merging (each region under each level is uniquely and sequentially labeled),

for each region  $i_1$  in  $k^{th}$  merging result,

$$R_j^{k+1} = R_{i_1}^k \cup R_{i_2}^k \quad \text{if } L_l(R_{i_1}^k \cup R_{i_2}^k) \leq L_l(R_{i_1}^k) + L_l(R_{i_2}^k) \quad (2.29)$$

where  $R_j^{k+1}$  is  $(k+1)^{th}$  merging result from current region  $i_1$  and candidate region  $i_2$ .

After each level of merging, labels are updated to produce a unique and sequential label for each region. Then  $j$  does not necessarily have the same label value as  $i_1$  in  $k^{th}$  level.

In equation (2.29), candidate region  $i_2$  is selected from all candidate neighbors  $l$  of  $i_1$  requiring that

$$L_I(R_{i_2}^k \cup R_{i_1}^k) \leq L_I(R_l^k \cup R_{i_1}^k) \quad (2.30)$$

4. Removing small abnormal regions. In this step, small abnormal regions are merged with an adjacent larger region. The existence of small abnormal regions may be due to: a) there are small objects on big object surfaces, and these small objects are not relevant; b) the presence of high-frequency noise in the image. For this purpose, formula (2.30) is still used to find the most suitable big region. Sometimes, several small areas can congregate together, which makes it impossible to find a direct-neighboring big region to merge with. Under the circumstances, the algorithm should allow current small regions to jump over neighboring small regions to the nearest big region.

## 2.6 Experiments

In this section, we apply the principle and algorithms we described in the previous sections to a number of images to illustrate their effectiveness. The first image in our experiment is a simulated ideal image Fig.(2.7a) with additive Gaussian noise (see Fig.(2.7b) <sup>2</sup>. The minimal gray value difference between the regions is 20, and the standard deviation of the Gaussian noise is 20. We perform a comparison between our method (independently implemented) and the split-and-merge algorithm implemented in SCILIMAGE software<sup>3</sup> by K.C. Strasters and J.J. Gerbrands [93]. Before applying region growing, we have to use some filters to suppress noise. The results of using Kuwahara filter (5x5 window) and uniform filter (3x3 window) are shown in Figs.(2.7c) and (2.7d). We found that in the most of the cases using a Kuwahara filter can help in achieving good segmentation results. In our algorithm, a thresholding value 2 is used for initial segmentation, which forms the basic small regions, followed by a merging algorithm using only the statistical test (in this case only the mean) with threshold 4. MDL-based operations further group the regions.

<sup>2</sup>ir. M.J.P.M. Lemmens provided some of the test images.

<sup>3</sup>SCILIMAGE software was developed by the Department of Computer Science, University of Amsterdam.

Fig.(2.7e) shows the segmentation output from the split-and-merge using variance criterion (for detail, see [93]). We found the variance criterion gets the best results compared with other criterion (e.g. max-min, sigma methods), for the split-and-merge method. Fig.(2.7f) is the result from our MDL-based segmentation.

We also show the results on other images. Figs.(2.8) are the results on brick wall image, with (a) the original image; (b) the filtered image; (c) the result from the split-and-merge algorithm; (d) the output from the MDL-based method. Similarly Figs.(2.9) (2.10) (2.11) and (2.12) show the results on several other images.

There are more results on segmentation from other images, such as the images we use for stereo matching. They are shown in the relevant chapters.

## 2.7 Conclusions

From the results shown in the previous section, also from our other experiments, we found that the MDL-based method can achieve the similar results as using the variance criterion when the noise on an image is relatively small, and it can get better results than the split-and-merge method using the variance criterion when a image contains more noise. Although there are a number of publications dealing with the MDL in feature extraction and segmentation, it is our contribution to use the MDL principle in a region growing algorithm. In a region growing algorithm, there are usually three main issues to be concerned, i.e. 1) the modeling; 2) the decision making mechanism for merging or splitting; 3) the implementation. When we consider whether two small regions should be merged or not, there are several things to be concerned: 1) what is the best model to be used for the studied data (assuming we have a number of preselected models); 2) the determination of the good points, noise, and outliers; 3) if we want to consider several different sources, how can we combine them to make a unique decision. The MDL principle has the power to permit estimates of the entire model, its parameters, their number, and even the way the parameters appear in the model, i.e., the model structure. In this chapter, the MDL principle has been used to describe the models, possible noise and outliers of image intensities. In the next chapter, we extend our algorithm to include shape constraints for image segmentation.

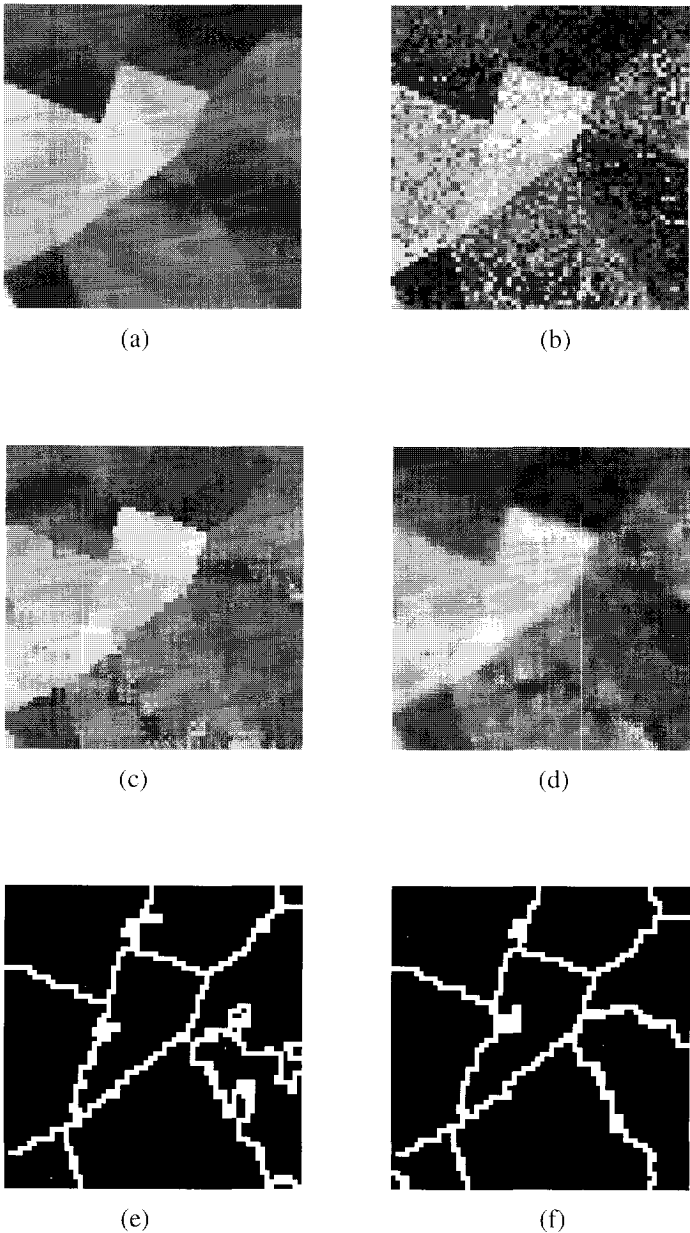


Figure 2.7: A simulated image (I): (a) the original image; (b) the noisy image; (c) the filtered image using Kuwahara filter; (d) the filtered image with uniform filter; (e) the segmentation result with variance criterion; (f) the segmentation result with MDL method.

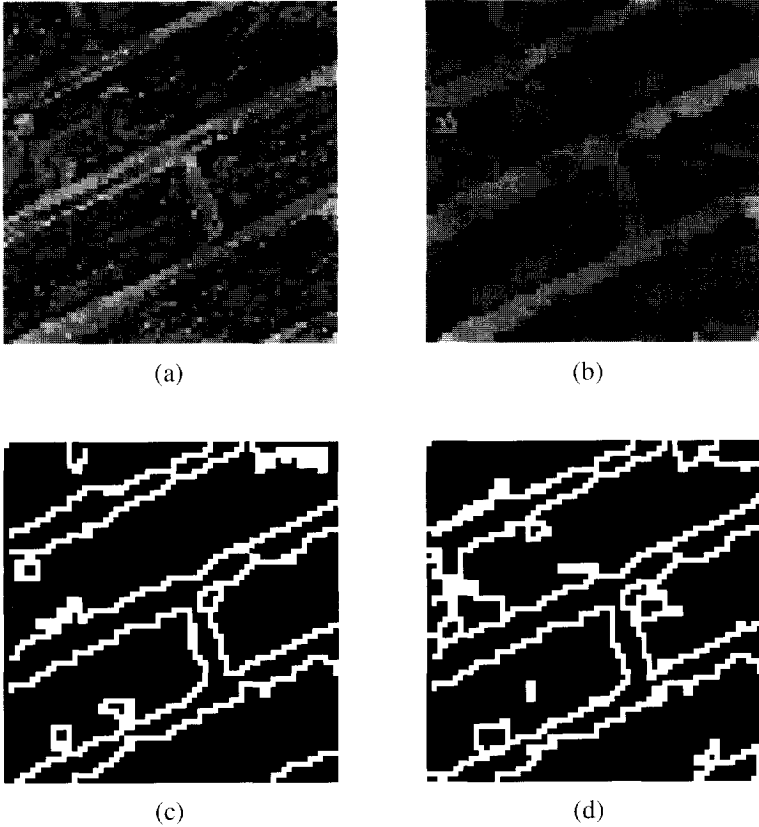


Figure 2.8: A wall image: (a) the original image; (b) the filtered image using Kuwahara filter; (c) the segmentation result with variance criterion; (d) the segmentation result with MDL method.

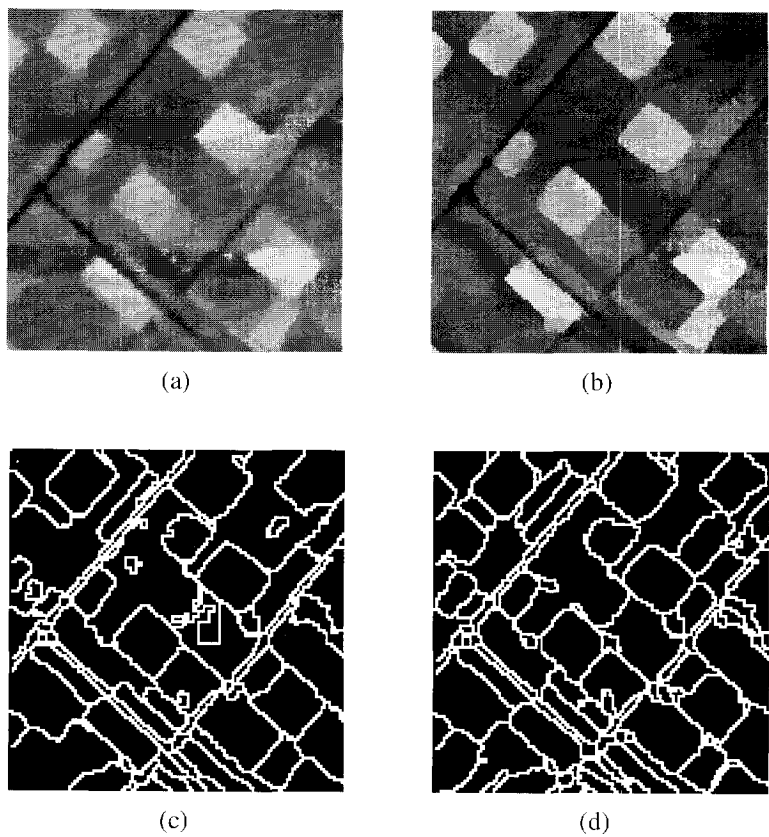


Figure 2.9: A remote sensing image: (a) the original image; (b) the filtered image using Kuwahara filter; (c) the segmentation result with variance criterion; (d) the segmentation result with MDL method.

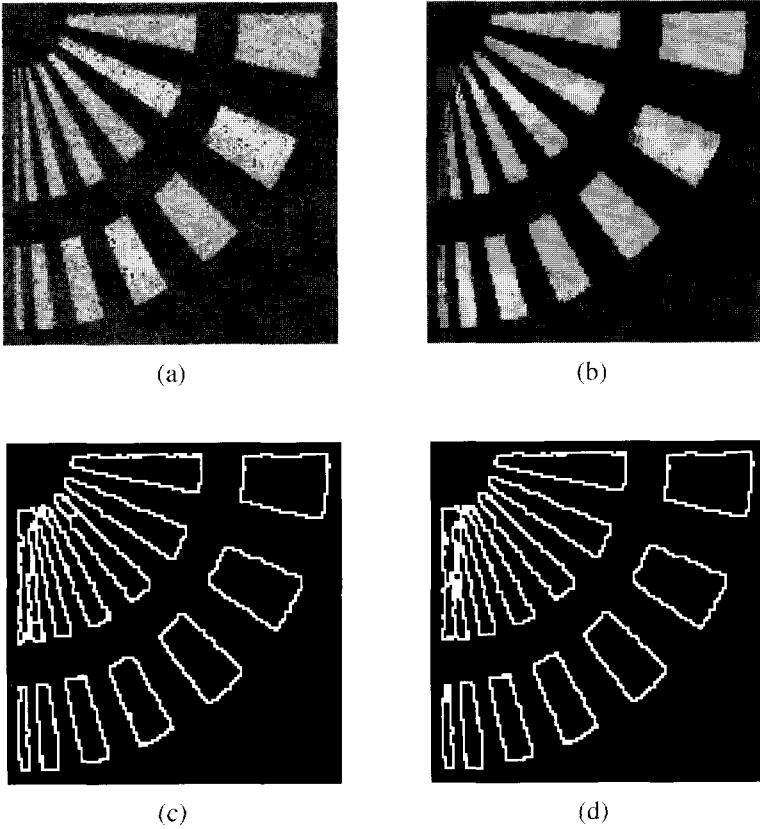


Figure 2.10: A simulated image (II): (a) the original image; (b) the filtered image using Kuwahara filter; (c) the segmentation result with variance criterion; (d) the segmentation result with MDL method.

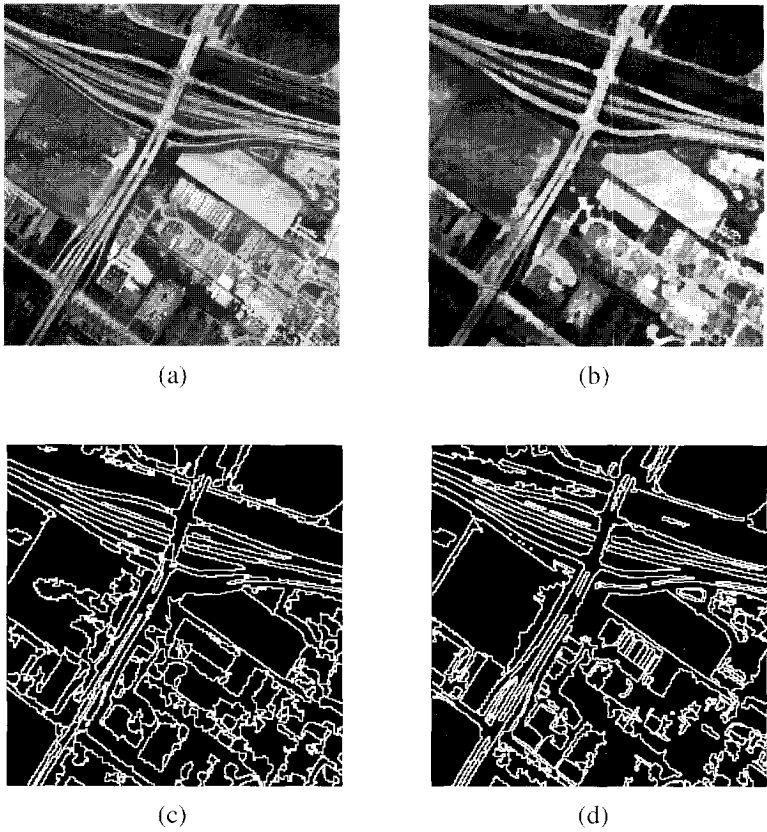
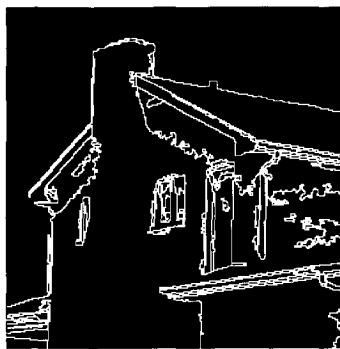


Figure 2.11: A remote sensing image: (a) the original image; (b) the filtered image using Kuwahara filter; (c) the segmentation result with variance criterion; (d) the segmentation result with MDL method.



(a)



(b)



(c)

Figure 2.12: A building image: (a) the original image; (b) the segmentation result with variance criterion; (c) the segmentation result with MDL method.

## Chapter 3

# Shape-Constrained Region Growing

### 3.1 Introduction

Almost all the traditional techniques of region-based segmentation have been based on region intensity. A lot of methods focused on choosing a proper discrimination strategy and implementation methods. In many applications, the scene consists of objects whose shapes are usually quite regular. It is reasonable to expect that use of a shape constraint can improve the segmentation result when applicable. In this chapter, we use the shape constraint in region growing. The shape model used here is depicted by a closed curve comprising several smooth segments. The Minimum Description Length (MDL) principle is used to encode the boundary shape of regions. The method of curve fitting is also described in this chapter.

For the purpose of using shape constraints in image segmentation, a multiple-level representation scheme is used here as illustrated in Fig. 3.1, which consists of the original image, the segmented image, vector data and shape models. After the segmentation using the method described in the previous chapter, a vectorization procedure transfers each region boundary into a vector description, followed by a curve fitting algorithm which derives compact vector data based on shape models. Based on the result of curve fitting, a measure is calculated to describe the uniformity of regions by shape constraints. Such a measure is used to further group the regions into more meaningful regions. In this chapter, we model the boundaries of image regions by closed polygons consisting of piecewise smooth functions. The discontinuities occur at the joining points of two smooth functions, and each smooth segment between two adjacent break points is described by

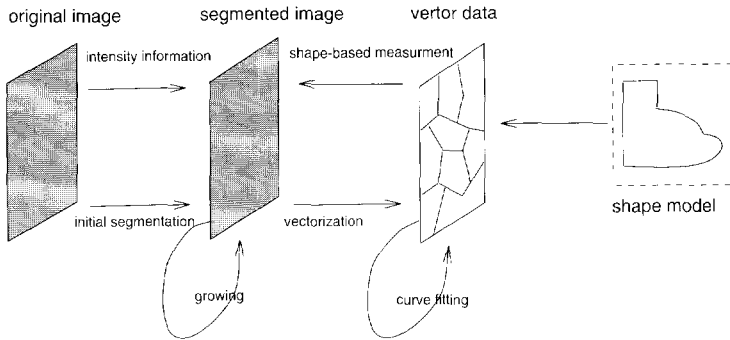


Figure 3.1: Multi-level representation of image segmentation.

one polynomial function.

This chapter is arranged as follows. The formulae for encoding the shape of regions are described in Section 3.2. Section 3.3 introduces the curve fitting algorithm. In Section 3.4, we present some examples based on the approach.

## 3.2 Encoding the shape of regions

Leclerc [87] developed some formulae to encode the boundaries that is based on the assumption that each segment of the boundary is described in terms of a straight line and the deviation from the line. The deviation is modeled as the perpendicular distance of a boundary point to the line, which obeys a Gaussian distribution. We argue that while it is reasonable to describe the image intensity deviation by a Gaussian distribution because the quantization of image intensity occupies a large range (usually from 0 to 255), it is not quite adequate to apply the same principle to encode the shape.

In order to analyze the coding formula for curves, we first take the model of a straight line. Its digitization fulfils the chord property which states [94] that "a digital arc A is said to have the chord property if for every two digital points, the chess-board distance of any point of arc A to the straight line connecting the two ending points nowhere exceeds 1". We consider the points on a curve which do not meet the chord property as outliers, and such outliers are constrained by their neighbors. For other kinds of segment models like polynomials, the principle is the same. That is, if the distance between a point to the fitted curve exceeds 1, it is considered to be an outlier.

The cost required to describe the image region shape is then formulated as:

$$\begin{aligned}
L_S = & 0 + 2m_2 \\
& + \left( m_1 \log\left(\frac{m_0}{m_1}\right) + m_2 \log\left(\frac{m_0}{m_2}\right) \right) \\
& + \log(D_x D_y) m_S
\end{aligned} \tag{3.1}$$

Where

$L_S$  is the number of bits describing the shape of a region

$m_0$  is the total number of points on the boundary

$m_S$  is the number of straight line segments on the boundary

$m_1$  is the number of points fulfilling the chord property

$m_2$  is the number of outliers

$D_x, D_y$  is the number of pixels along x and y direction of the image

In accordance with Eq.(2.11), we also use four terms to encode the shape of regions. For the points on the curve which meet the chord condition, no additional coding is needed if the nodes specifying the smooth segments are known, so the first item in this case is zero. The second term in Eq.(3.1) is the number of bits describing the outliers. If the boundary is encoded in Freeman chain code, 3 bits are required to store each pixel (for 8 directions). But if we store the edges between the pixels instead of the pixels themselves, only 2 bits are necessary (for 4 directions). Such way of representing the boundary allows us to depict closed regions, single lines as well as points in a unified way. The third term has similar meaning as in the one in Eq.(2.11). The final component is used for the coding of nodes connecting smooth segments.

### 3.3 Boundary Fitting Algorithm

Boundary fitting, or curve fitting is a part of the problem of shape analysis, which deals with using the minimum number of points to represent a curve under certain error criteria. Generally, there are two kinds of techniques for curve fitting:

- Local detection of distinguished points: A partition point is inserted at locations along the curve at which one or more of the descriptive attributes (e.g. curvature, distance from a coordinate axis or centroid) is determined to have a discontinuity, an extreme value (maxima or minima) or a zero value separating intervals of positive and negative values;

- Best global description: A set of partition points is inserted at those locations along a curve that allow the "best" description of the associated segments in terms of a priori models (e.g., the set of models might consist of all first and second degree polynomials, with only one model permitted to explain the data between two adjacent partition points; the quality of the description might be measured by the mean square deviation of the data points from the fitting polynomials).

Our method combines the detection of extreme points (we adopt the Eom and Park's method [95]) and a splitting mechanism. A digital contour is modeled by noisy observations which are represented by polynomial functions of coordinate variables. The maximum likelihood estimator is used to estimate the curvature function of the digital contour. The candidate extreme points are detected by finding the zero-crossings of the first derivative of the estimated curvature function. These candidate extreme points are treated as the seed points for the succeeding splitting mechanism, which selects the extreme points best representing the original curves using the model functions. Details of the method are presented below.

### 3.3.1 Detection of extreme points

#### Digital contour model

Let us assume that a sequence of contour points  $\{(x_i, y_i), i = 0, \dots, N - 1\}$  is a noisy observation of a continuous vector function  $z(t) = (x(t), y(t))$ , where  $z(t)$  is a non-observable contour function before digitization. The vector function  $z(t) = (x(t), y(t))$  is a non-observable, continuous, and smooth function, and its derivatives exist up to the second order. Since the observed sequence can change values in unit distance, the functions  $x(t)$  and  $y(t)$  can be modeled by low order polynomial functions of  $t$ , such as the second order polynomial functions given here:

$$\begin{aligned} x(t) &= a_0 + a_1 t + a_2 t^2 \\ y(t) &= b_0 + b_1 t + b_2 t^2 \end{aligned} \quad (3.2)$$

Then the digital contour may be modeled by

$$\begin{aligned} x_i &= A_0 + A_1 i + A_2 i^2 + w_i \\ y_i &= B_0 + B_1 i + B_2 i^2 + v_i \end{aligned} \quad (3.3)$$

where  $A_0 = a_0, A_1 = a_1 T, A_2 = a_2 T^2, B_0 = b_0, B_1 = b_1 T, B_2 = b_2 T^2$ , ( $T$  is related to digitization resolution) and where  $w_i$  and  $v_i$  are independent zero mean white Gaussian sequences with variances  $\rho_1$  and  $\rho_2$ , respectively.

Consider a contour point  $(x_i, y_i)$ , its  $K$  backward neighbors  $\{(x_{i-K}, y_{i-K}), \dots, (x_{i-1}, y_{i-1})\}$  and  $K$  forward neighbors  $\{(x_{i+1}, y_{i+1}), \dots, (x_{i+K}, y_{i+K})\}$ . Then maximum likelihood estimations  $A_0^*, A_1^*, A_2^*, B_0^*, B_1^*, B_2^*$  are obtained by maximizing the likelihood functions with respect to the model parameters [95]

$$\begin{aligned} [A_0^*, A_1^*, A_2^*]^T &= \left[ \sum U(j)U(j)^T \right]^T \left[ \sum x_j U(j) \right] \\ [B_0^*, B_1^*, B_2^*]^T &= \left[ \sum U(j)U(j)^T \right]^T \left[ \sum y_j U(j) \right] \end{aligned} \quad (3.4)$$

where  $U(j) = [1, j, j^2]^T$ , and the summation is over  $K$  backward and  $K$  forward neighbors.

### Optimal choice of neighborhood size

The neighborhood size  $K$  in the contour model is determined by the following maximum likelihood decision rule [95]

$$K_{opt} = \arg \{ \max J(K), K_{min} \leq K \leq K_{max} \} \quad (3.5)$$

where  $K_{min}$  and  $K_{max}$  define the range of  $K$ , and  $J(K)$  is the likelihood function given by

$$\begin{aligned} J(K) &= -(2K - 3) \log 2\pi - \log(2K + 1) \\ &\quad - K \log \rho_1^* - K \log \rho_2^* + \log |\Lambda_K| \end{aligned} \quad (3.6)$$

where  $\Lambda_K = \sum U(j)U(j)^T$ .

### Curvature estimation

Suppose  $z(t) = (x(t), y(t))$  is a vector function in 2-D space that traces a continuous contour. Then, the curvature function of the contour is defined by

$$\kappa(t) = \frac{\dot{x}(t)\ddot{y}(t) - \dot{y}(t)\ddot{x}(t)}{[\dot{x}^2(t) + \dot{y}^2(t)]^{\frac{3}{2}}} \quad (3.7)$$

Substituting the second order polynomial functions given in Eq.(3.2), we obtain the curvature function in terms of model parameters:

$$\kappa(t) = \frac{2(a_1 b_2 - a_2 b_1)}{C^{\frac{3}{2}}} \quad (3.8)$$

with

$$C = 4(a_2^2 + b_2^2)t^2 + 4(a_2a_1 + b_2b_1)t + (a_1^2 + b_1^2)$$

The curvature function for a digital contour is given by the following:

$$\kappa_i^* = \begin{cases} \frac{2(A_1^*B_2^* - A_2^*B_1^*)}{(A_1^{*2} + B_1^{*2})^{\frac{3}{2}}} & \text{if } A_1^* \neq 0 \text{ or } B_1^* \neq 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.9)$$

Extreme points are detected by finding zero-crossing of the first derivative of the estimated curvature function. The extreme detection algorithm is summarized in the following.

#### Extrema point detection algorithm

1. Estimate the curvature function  $\kappa(t)$  from the contour.
2. Compute its first derivative.
3. Detect an extreme point at  $t$  if the following two conditions are satisfied:
  - $|\kappa^*(t)| > K_1$ , where  $K_1$  is a constant.
  - The first derivative of the curvature function has a zero-crossing at  $t$ .
4. Repeat 3) for each point in the contour.

### 3.3.2 Splitting process

There are several versions of the split-and-merge algorithm. For example, Pavlidis and Horowitz [96] approximate a boundary by straight line segments. The method described below uses only a splitting mechanism to achieve the objective.

---

#### Algorithm 3.1 :

1. Detect the candidate extreme points as described in 3.3.1.
2. Choose the two best extreme points so that the sum of the errors over the curve is minimum.

- (a) choose the point with highest curvature as the first point.
  - (b) fix the first point, choose the second point among the candidate extreme points so that the error of fitting is minimum.
  - (c) fix the second point, update the first point using remaining candidate extreme point so that fitting error is minimum.
  - (d) repeat (b) and (c), until no changes in the positions of points.
3. Choose the interval between two neighboring extreme points which has the biggest fitting error over the curve, insert one extreme point so that the sum of the fitting errors is minimum.
  4. Repeat 3) until the error sum over the whole curve meets the specified criterion.

---

There are several aspects to be addressed concerning the above algorithm:

1. Error definition. There are different kinds of definitions. For instance, the maximum distance between the original curve and the fitted curve, the area of difference between the original curve and the the fitted curve, the average distance between the original curve and fitted curve.
2. Only using splitting leads to simpler data structures during the implementation. Yet the solution will be achieved because this is guaranteed by:
  - Good selection of starting points. This is essential. In the normal split-and-merge scheme, the initial break points are assigned randomly along the boundary. Since some of these points are not necessarily at the desired positions, a merging process is required to remove these redundant points. Our approach uses iterative updating to find the best positions for the first two extreme points.
  - Select the interval where the largest error occurs. This step ensures the next extreme point is within this portion.
  - Within each interval, the next point is chosen in such a way that the fitting error is minimized.
3. Advantage of combining the extreme point detection and splitting:
  - The splitting mechanism guarantees the fitting requirement to be met (see the discussion above).

- By detecting candidate extreme points, a higher computational efficiency can be achieved because the number of candidate extreme points is significantly smaller than the number of original points.

### 3.4 Examples

In this section, we apply the principle and algorithms we described in the previous sections to a number of images to illustrate their effectiveness. First of all, the results from the curve fitting are illustrated in Figs.3.2, 3.3, 3.4, where (a) shows the original curve, and (b), (c), (d) show the fitting results with different error specifications. Note that in Fig.3.4 we use a second order polynomial to model the curves between two extreme points, and the fitting results are shown with straight line segments.

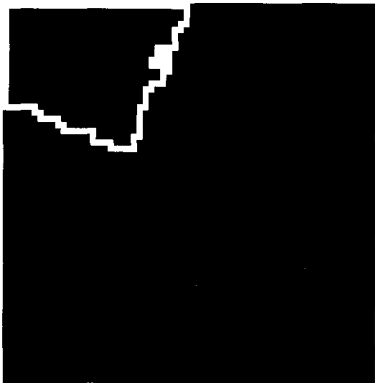
The first image in our experiment of shape-constrained region growing is a simulated ideal image with additive Gaussian noise (see Fig. 3.5). The minimal gray value difference between the regions and the background is 100, and the standard deviation of the Gaussian noise is 50. The filtered image (using Kuwahara filter), segmentation result using a split-and-merge algorithm, segmentation result using MDL based method, segmentation result using shape constraint are shown in (c), (d), (e), (f), respectively.

On another simulated image (Fig.3.6), the minimal gray value difference between regions on the original image is 25. The standard deviation of added noise is 35. The similar results are also shown.

We also show the result of segmenting a remote sensing image (see Fig.3.7).

### 3.5 Concluding remarks

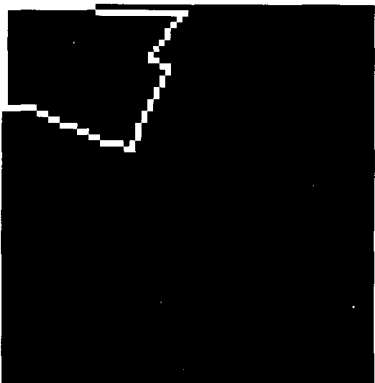
From the above examples, we can see that our fitting algorithm works quite satisfactorily with the test data and using shape constraints can improve the segmentation result when the noise is relatively large and the images contain objects with regular shapes. Therefore, this method is especially suitable for images consisting of man-made objects like buildings, indoor scenes and objects for robotics under severe noise. However, this method takes a quite long time because of the tedious processing of boundary fitting. The extension of this work includes the treatment of complex topological description of object models and integration with other middle-level image analysis tasks, as indicated in [97].



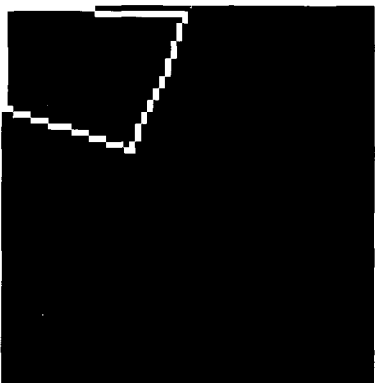
(a) the original curve with 121 points



(b) curve fitting result with 15 points

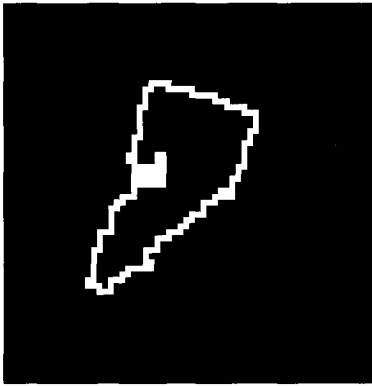


(c) curve fitting result with 11 points

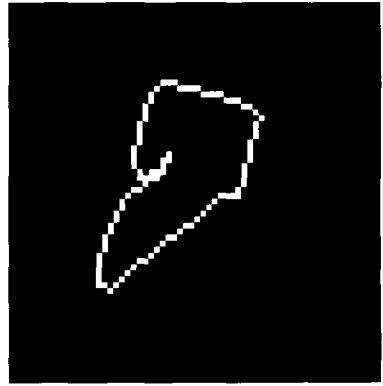


(d) curve fitting result with 7 points

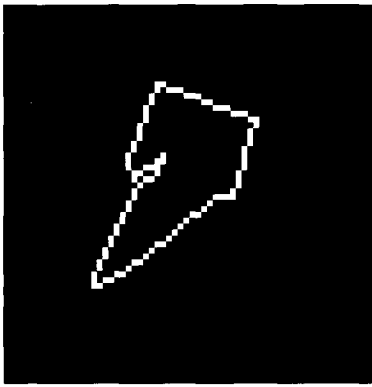
Figure 3.2: First curve image for curve fitting test.



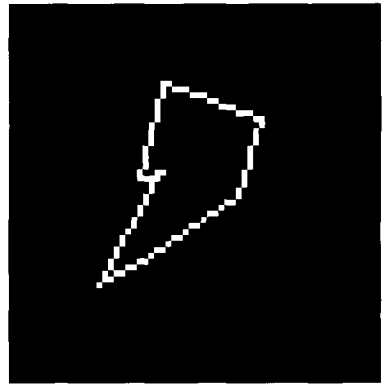
(a) the original curve with 159 points



(b) curve fitting result with 30 points



(c) curve fitting result with 16 points

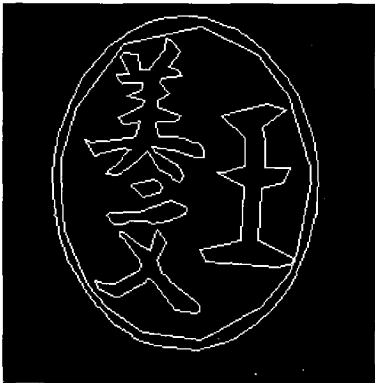


(d) curve fitting result with 9 points

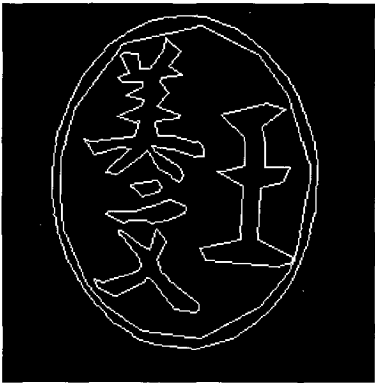
Figure 3.3: Second curve image for curve fitting test.



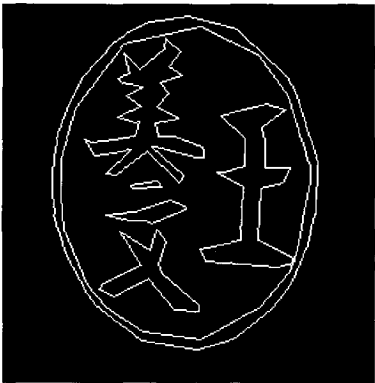
(a) the original curve with 4546 points



(b) curve fitting result with 301 points



(c) curve fitting result with 266 points



(d) curve fitting result with 169 points

Figure 3.4: Third curve image for curve fitting test.

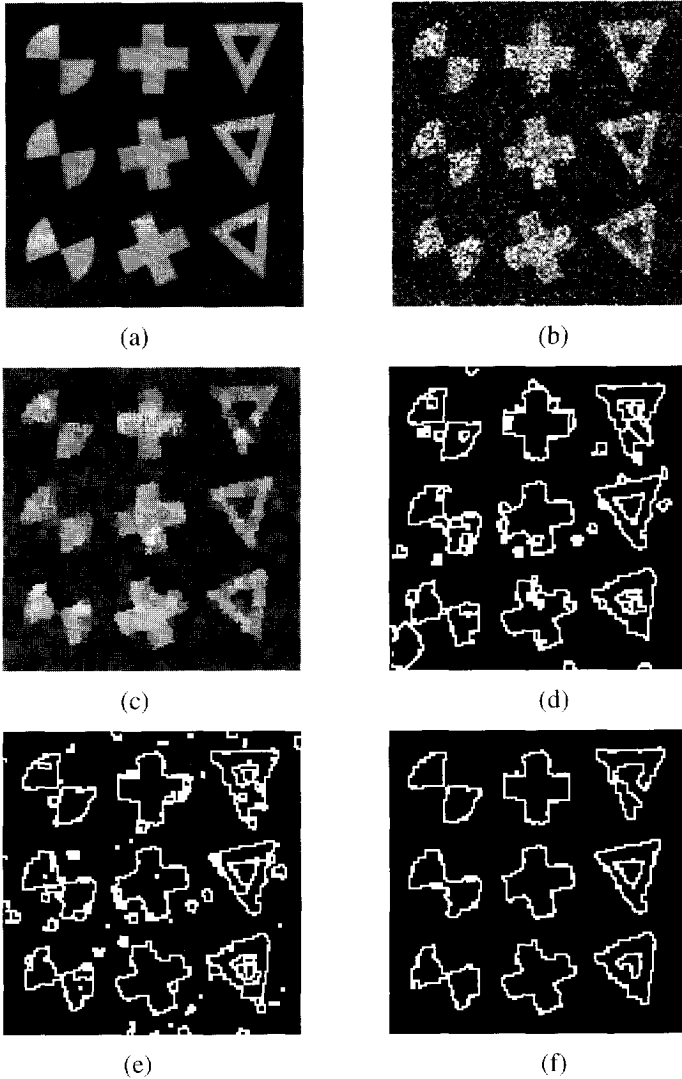


Figure 3.5: A simulated image with added noise ( $\sigma = 50$ ), (a) original image; (b) image with added noise; (c) filtered image; (d) segmentation result using a split-and-merge algorithm; (e) segmentation result using MDL based method; (f) segmentation result using shape constraint.

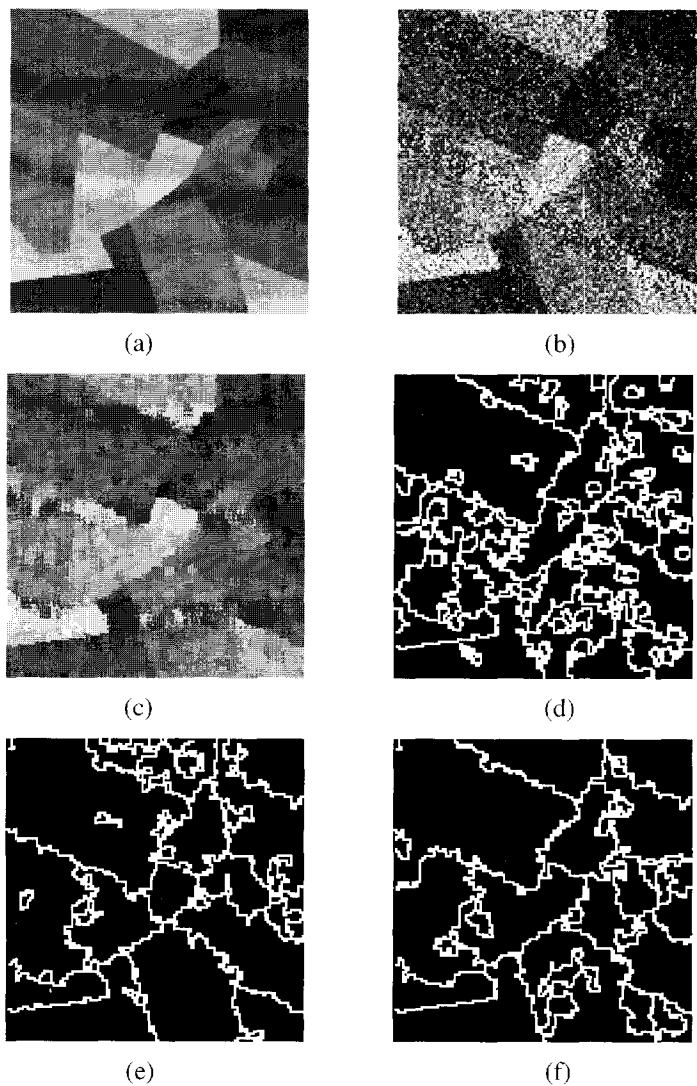


Figure 3.6: Another simulated image with added noise ( $\sigma = 35$ ), (a) original image; (b) image with added noise; (c) filtered image; (d) segmentation result using a split-and-merge algorithm; (e) segmentation result using MDL based method; (f) segmentation result using shape constraint.

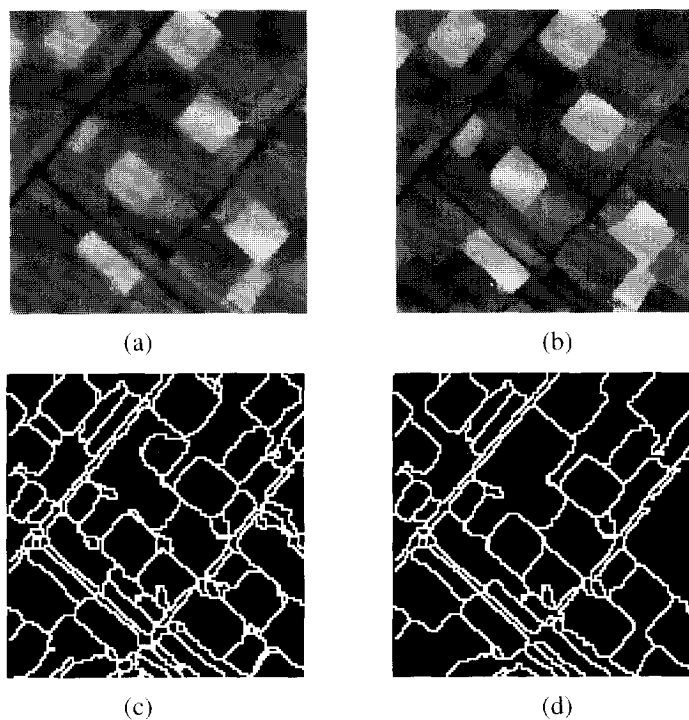


Figure 3.7: A remote sensing image, (a) original image; (b) filtered image; (c) segmentation result using MDL based method; (d) segmentation result using shape constraint.

## Chapter 4

# Curve Matching

### 4.1 Introduction

Region based stereo matching consists of local matching and global matching. This chapter is devoted to local matching, which compares regions on the left image with regions on the right image based on their local attributes. Topological information associated with regions is not a concern within this procedure. Attributes related to regions consists of: geometric attributes, such as region area, shape of region boundary, etc; physical and optical attributes, such as color, gray value information. Our investigation of matching techniques in this chapter concentrates on geometric attributes, mainly on matching the boundaries of stereo regions. A thresholding method is used to compare the intensity difference between stereo regions.

The techniques on curve matching can be classified as local or global in nature. In the global approaches, a set of global numerical features is extracted, and each arc is represented by these features. The matching procedure compares the two arcs in the global feature space. The global features include Fourier descriptors, invariant moments [98], shape vectors [99] [100], and some shape measures such as compactness, elongatedness, etc. [101]. Using global features makes the matching procedure invariant to changes in size, location, and orientation. The main problems with this approach are the sensitivity to noise and the fact that a description of a piece of shape is not simply related to a descriptor for the entire shape, which make the global approach difficult to apply to the segment matching problem.

The other approach to the arc matching problem is that of using local attributes to represent arcs. In [102], an algorithm for matching closed polygonal arcs is presented. The arcs are presented as sequences of 2-tuples  $(a'_i, l_i)$ , where  $a'_i$  is the angle of the  $i$ th line segment and  $l_i$  is its length. The best match is found by using a relaxation

method. In [103], a hierarchical relaxation method for polygon matching is presented. The hierarchical technique is intended to reduce the high computational cost of relaxation methods.

In [104], algorithms for polygon matching are presented with two kinds of mismatch measures: (a) the area of the difference between the two polygons, and (b) the integral of the squared distance between corresponding points on the two polygons. The main problem with this method is the amount of computation needed and the authors present some heuristics to reduce it.

In [105], arcs are represented as attribute strings and symbolic string matching procedures are used. The primitives used are line segments and the attributes are length and direction. There is a cost assigned to each *edit* operation, such as *insert symbol*, *delete symbol*, or *change symbol*, and the matching measure is the minimal-cost sequence of edit operations that transform  $S_1$  into  $S_2$ . The authors introduce a new *edit* operation called *merge symbols* which improves the matching algorithm performance.

In [106], curve segments are represented using B-splines which are piecewise polynomial curves guided by a sequence of points. The B-spline control points found from the boundary points are then used to extract local features of the curve. A Hough-transform-like method is applied to normalize the two curve boundaries using extracted local features. The merit of a match is evaluated using the normalized B-spline control points.

In [107], two arcs of the same length are matched by decomposition into parts of equal length line segments. The distance measure used is the sum of the squared distances between corresponding points on the two arcs. The best match is found by evaluating analytically the relative position and orientation of the arcs that minimize the distance measure.

In [108], contours made up of sequences of adjacent edge points are used as primitives in stereo pair matching. Matching contour segments, rather than the traditional epipolar edge points, can greatly reduce possible ambiguity. This is done by reformulating point matching constraints to apply to contour matching, and by introducing a *unique incremental* matching scheme. Best matched contours are paired first, constraining through neighborhood support their neighboring contours.

In [109], a graph-theoretic optimization method is used to recognize partially occluded objects from a 2-D image through the use of maximal cliques and a weight matching algorithm.

A planar shape distorted by a projective viewing transformation can be recognized under partial occlusion if an invariant description of its boundary is available [110]. Invariant boundary descriptions should be based solely on the local properties of the boundary curve, perhaps relying on further information on the viewing transformation.

Recent research in this area has provided a theory for invariant boundary descriptions based on an interplay of differential, local, and global invariants. A paper by Braunegg [111] describes a new method for matching, validating, and disambiguating features for stereo vision. It is based on the Marr-Poggio-Grimson stereo matching algorithm which uses zero-crossing contours in difference-of-Gaussian stereo matching images as features. The matched contours are represented in disparity space, which makes the information needed for matched contour validation and disambiguation easily accessible. The use of disparity space also makes the algorithm conceptually cleaner than previous implementations of the Marr-Poggio-Grimson algorithm and yields a more efficient matching process. All invariants require high-order derivatives. However, the use of global invariants and point match information on the distorting transformations enables the derivation of invariant signatures for planar shape using lower order derivatives. Trade-off between the highest order derivatives required and the quantity of additional information constraining the distorting viewing transformation are made explicit. Once an invariant is established, recognition of the equivalence of two objects requires only partial function matching. Uses of these invariants include the identification of planar surfaces in varying orientations and resolving the outline of a cluster for planar objects into individual components.

The method described in [112] is based on quadratic approximations to the effect of the slant and of an inverse perspective transformation on angles and lengths. The approximations allow the definition of contour-based properties that are invariant under perspective transformation. The method can be used to recognize partially occluded shapes, as well as shapes that are not exactly related by perspective transformation.

In [113], path-based Gaussian smoothing techniques are applied to the curve to find zeros of curvature at varying levels of detail. The result is the "generalized scale space" image of a planar curve which is invariant under rotation, uniform scaling and translation of the curve. The matching algorithm is a modification of the uniform cost algorithm and finds the lowest cost match of contours in the scale space image.

In [113], arcs are represented as scale space images. These images consists of the curvature of the arcs at points of inflection at varying levels of detail. The curvature of the arc at a specific level of detail is computed by convolving the arc with a two-dimensional Gaussian mask using a value of  $\sigma$  corresponding to the desired level of detail. The matching is done on the scale images. The matching can be performed hierarchically by first matching the less detailed representations.

The chamfer matching technique [114] [115] is a kind of template matching method where the measure of similarity is the distance between a feature point in the template and the nearest feature point in the image. Given two arcs,  $A_1$  and  $A_2$ , in two images to be compared, one first extracts the arcs and transforms the original images into feature arrays.

In a feature array the pixel where features occur get the value 0 and all the other pixels get the value  $\infty$ . One of the two arrays of features (the image array) is then transformed into an array of numbers. Each number is either 0 (at feature points) or an estimate of the distance to the nearest feature point. The mismatch measure is computed by examining the feature points in the template and summing the distance in the corresponding locations in the array of numbers derived from the image. There are some problems with this method as mentioned by the authors. It performs poorly when the image and the template are badly out of alignment; parallel linear feature may cause aliasing; and a single isolated feature can support multiple reference features. Another problem is the asymmetry of the method; the mismatch values depend on which of the two arrays of feature is transformed into an array of numbers.

In this chapter, we present a method to solve the stereo matching problem of general closed planar curves, provided that camera parameters (interior orientation and relative orientation) are known. The stereo curves are assumed to be the output of some segmentation procedures and represent the boundaries of certain objects. Usually, segmentation procedures are not perfect and occlusions may occur. This means that some portions of the stereo curves may be detected incorrectly or may be missing altogether. The general solution to this problem is not available. The existing curve matching methods have the following problems: 1) some methods can only deal with two-dimensional rotation, translation, and scaling, but cannot deal with perspective transformation; 2) some methods are sensitive to noise and cannot tackle errors due to bad segmentation and occlusion; 3) some methods have strict constraints with respect to the mathematic properties of the shape of the curves concerned (e.g. smooth curvature). In this article, we develop a curve matching method which is capable of dealing with perspective transformation and of handling possible errors due to bad segmentation and occlusion without making assumptions on the shape of the curves. The assumptions we do make are: 1) the stereo curves are the perspective projections of 3-D planar curves in an object space; 2) the stereo curves are closed; 3) any curve concerned is a simple curve (if a curve representing a region boundary stems from the output of a region growing algorithm, this requirement is always guaranteed; if the curve is produced by other kinds of segmentation techniques, some operations may be needed in order to meet this requirement). The method we propose in this chapter decomposes parameters related to an object plane, i.e. slant, tilt and scale factor, and uses a histogram technique to estimate these parameters based on disparity information. The point correspondence problem is solved in a dynamic programming style. The final matching is assessed by applying a distance transformation. The issue of dealing with camera parameters is also addressed.

This chapter is organized in the following manner. Section 4.2 describes the formulae

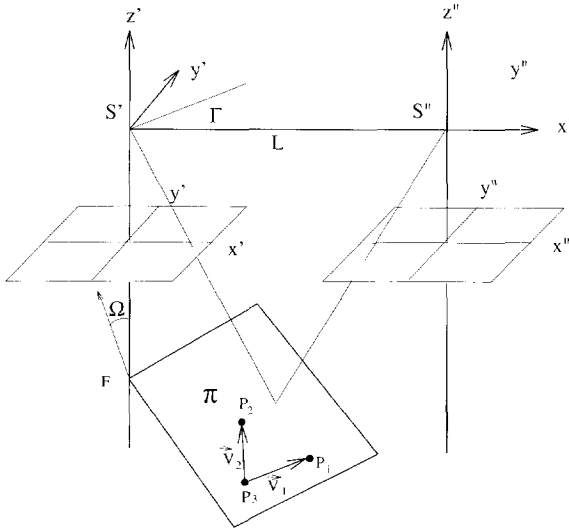


Figure 4.1: Ideal configuration of stereo camera.

for calculating the three parameters slant, tilt, and scale factor under an ideal configuration. The problem of transforming a normal stereo configuration into an ideal stereo configuration is solved in Section 4.3. In Section 4.4, the algorithm for curve matching is described in detail. The parameter estimation procedure using a histogram technique as well as matching evaluation are described in Section 4.5. Section 4.6 describes an experiment based on the method proposed. Section 4.7 contains the conclusions.

## 4.2 Decomposition of slant, tilt and scale factor

The geometric differences of stereo curves are caused by the camera parameters and the parameters related to the object plane concerned. In this section, we consider an ideal configuration for the stereo cameras in which the angle parameters describing relative orientation between stereo cameras are zero. In order to use a histogram technique to estimate the parameters of the object plane, we need first to decompose these parameters. The relevant formulae are described in the following.

### 4.2.1 Projective geometry

We first analyze the projection property of a 3-D planar curve under an ideal stereo camera geometry. In Fig. 4.1,  $S'$  and  $S''$  are the projection centers for the left and right

images, respectively. The two images are on the same plane in this ideal situation. We set the baseline of the stereo camera as the  $x'$  axis of the left photo coordinate system (image coordinates often refer to pixel indices on digital image matrices; for clarity, we use photo coordinates as a real metric measure of point positions on the corresponding photograph) and keep  $x'y'$  parallel to the image plane (in this case  $S' - x'y'z'$  is also used as object coordinate system for the left image). The right photo coordinate system  $S'' - x''y''z''$  is set parallel to  $S' - x'y'z'$ .

The central projection principle for a stereo camera implies the following equations for an object point  $P(X', Y', Z')$ :

for the left image,

$$\begin{cases} x' = -f_c \frac{X'}{Z'} \\ y' = -f_c \frac{Y'}{Z'}, \end{cases} \quad (4.1)$$

and

for the right image,

$$\begin{cases} x'' = -f_c \frac{X''}{Z''} = -f_c \frac{X' - L}{Z'} \\ y'' = -f_c \frac{Y''}{Z''} = -f_c \frac{Y'}{Z'} = y', \end{cases} \quad (4.2)$$

where  $f_c$  is the principal distance of the camera;  $L$  is the length of the baseline;  $(x', y')$  and  $(x'', y'')$  are the photo coordinates of corresponding left and right image points, respectively;  $(X'', Y'', Z'')$  are the object coordinates of point  $P$  in the right object coordinate system  $S'' - X''Y''Z''$ .

The difference between the  $x$  coordinates of corresponding points on the stereo images

$$d = x' - x'' \quad (4.3)$$

is defined as *disparity*. Observing (4.1) and (4.3), we have,

$$d = -f_c \frac{L}{Z'}. \quad (4.4)$$

From (4.1) and (4.4), we can solve the object coordinates:

$$\begin{cases} X' = \frac{x'L}{d} \\ Y' = \frac{y'L}{d} \\ Z' = -\frac{f_c L}{d} \end{cases} \quad (4.5)$$

#### 4.2.2 Determination of slant $\Omega$ and tilt $\Gamma$ of an object plane

In Fig. 4.1, point  $F = (0, 0, C)$  is called the fixation point where  $C$  marks how far an object plane ( $\pi$ ) is from the camera center  $S'$ . The other two parameters of the object plane are determined by the *slant*  $\Omega$  and *tilt*  $\Gamma$  of the object plane, where slant is the angle between the  $z'$  axis and the normal of the plane (we let  $0 \leq \Omega \leq 90^\circ$ ) and tilt is the angle between the  $x'$ -axis and the projection of the normal on the image plane ( $0 \leq \Gamma \leq 360^\circ$ ). The normal vector of the object plane ( $\pi$ ) is determined by three points  $P_1, P_2, P_3$  or two vectors  $\vec{V}_1$  and  $\vec{V}_2$  on the plane. Let  $\vec{n} = \{n_x, n_y, n_z\}$  be the normal vector, then

$$\vec{n} = \vec{V}_1 \times \vec{V}_2 = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ \Delta X'_1 & \Delta Y'_1 & \Delta Z'_1 \\ \Delta X'_2 & \Delta Y'_2 & \Delta Z'_2 \end{vmatrix} \quad (4.6)$$

if

$$\vec{V}_1 \cdot \vec{V}_2 \neq 0 \quad (4.7)$$

where  $\vec{i}, \vec{j}, \vec{k}$  are the unit directional vectors along  $x'y'z'$  axis. In Eq. (4.6),

$$\begin{cases} \Delta X'_i = X'_i - X'_3 = (\frac{x'_i}{d_i} - \frac{x'_3}{d_3})L \\ \Delta Y'_i = Y'_i - Y'_3 = (\frac{y'_i}{d_i} - \frac{y'_3}{d_3})L \\ \Delta Z'_i = Z'_i - Z'_3 = (-\frac{f_c}{d_i} + \frac{f_c}{d_3})L, \end{cases}$$

where  $X'_i, Y'_i, Z'_i (i = 1, 2)$  represent the object coordinates of point  $P_i$  on plane ( $\pi$ ),  $x'_i, y'_i$  are the photo coordinates of the corresponding left image point, and  $d_i$  is the corresponding disparity value.

Thus,

$$\begin{aligned}
 n_x &= \begin{vmatrix} \Delta Y'_1 & \Delta Z'_1 \\ \Delta Y'_2 & \Delta Z'_2 \end{vmatrix} \\
 &= \Delta Y'_1 \Delta Z'_2 - \Delta Y'_2 \Delta Z'_1 \\
 &= \{(y'_1 d_3 - y'_3 d_1)(d_2 - d_3)f_c - (y'_2 d_3 - y'_3 d_2)(d_1 - d_3)f_c\} L/d_1 d_2 d_3^2 \\
 n_y &= \begin{vmatrix} \Delta Z'_1 & \Delta X'_1 \\ \Delta Z'_2 & \Delta X'_2 \end{vmatrix} \\
 &= \Delta X'_2 \Delta Z'_1 - \Delta X'_1 \Delta Z'_2 \\
 &= \{(x'_2 d_3 - x'_3 d_2)(d_1 - d_3)f_c - (x'_1 d_3 - x'_3 d_1)(d_2 - d_3)f_c\} L/d_1 d_2 d_3^2 \\
 n_z &= \begin{vmatrix} \Delta X'_1 & \Delta Y'_1 \\ \Delta X'_2 & \Delta Y'_2 \end{vmatrix} \\
 &= \Delta X'_1 \Delta Y'_2 - \Delta X'_2 \Delta Y'_1 \\
 &= \{(x'_1 d_3 - x'_3 d_1)(y'_2 d_3 - y'_3 d_2) - (x'_2 d_3 - x'_3 d_2)(y'_1 d_3 - y'_3 d_1)\} L/d_1 d_2 d_3^2.
 \end{aligned}$$

$\Omega$  and  $\Gamma$  can be calculated by

$$\Omega = \arccos \left( \frac{|n_z|}{\sqrt{n_x^2 + n_y^2 + n_z^2}} \right) \quad (4.8)$$

$$\Gamma = \arccos \left( \frac{|n_x|}{\sqrt{n_x^2 + n_y^2}} \right). \quad (4.9)$$

With Eq. (4.9) the value of  $\Gamma$  can be determined within  $0 \leq \Gamma \leq 90^\circ$ . The signs of  $n_x, n_y$  are therefore required to decide to which quadrant  $\Gamma$  belongs. It is obvious that  $L$  does not play any role in calculating  $\Omega$  and  $\Gamma$ , so we set  $L = 1$ .

### 4.2.3 Determination of the scale factor

Case of near-horizontal object plane ( $\Omega < 45^\circ$ )

Consider a near-horizontal object plane ( $\pi$ ) when  $\Omega < 45^\circ$  having equation

$$Z' = AX' + BY' + C, \quad (4.10)$$

where  $(X', Y', Z')$  is used to express a point on the object plane. It is obvious that (see Fig. 4.1)

$$A = \tan(\Omega) \cos(\Gamma), \quad B = \tan(\Omega) \sin(\Gamma). \quad (4.11)$$

Substituting the corresponding image coordinates  $x', y'$  into the equation of the object plane using Eq. (4.1), we get

$$Z' = -A \frac{x'Z'}{f_c} - B \frac{y'Z'}{f_c} + C.$$

from which it follows that,

$$Z' = \frac{C}{1 + \frac{Ax'}{f_c} + \frac{By'}{f_c}}.$$

Replacing the above equation into Eq. (4.4) for disparity computing, we have the result (note  $L = 1$ )

$$d = -\frac{1}{C}(f_c + Ax' + By') = -\frac{1}{C}(f_c + \tan(\Omega) \cos(\Gamma)x' + \tan(\Omega) \sin(\Gamma)y'). \quad (4.12)$$

After  $\Omega, \Gamma$  are determined, the remaining unknown variable is  $C$ . In our experience, it is easier to compute  $1/C$  rather than  $C$ . For convenience, we define  $1/C$  as  $S_c$ , a scale factor.

Referring to Eq. (4.12),  $S_c$  is solved in the near-horizontal case by

$$S_c = -\frac{d}{f_c + \tan(\Omega) \cos(\Gamma)x' + \tan(\Omega) \sin(\Gamma)y'} \quad (4.13)$$

Case of near-vertical object plane ( $\Omega \geq 45^\circ$ )

When  $\Omega = 90^\circ$ ,  $\tan(\Omega) = \infty$ , Eq. (4.12) cannot be used as a proper expression. We need to examine other ways for expressing disparity.

Referring to Fig. 4.2, when the object plane is nearly vertical, its orientation may better be represented by  $\Omega'$  and  $\Gamma$ , where  $\Omega'$  is the angle between the normal of the plane and  $x'y'$  plane and  $\Gamma$  has the same meaning as in the case of a non-vertical plane. It is

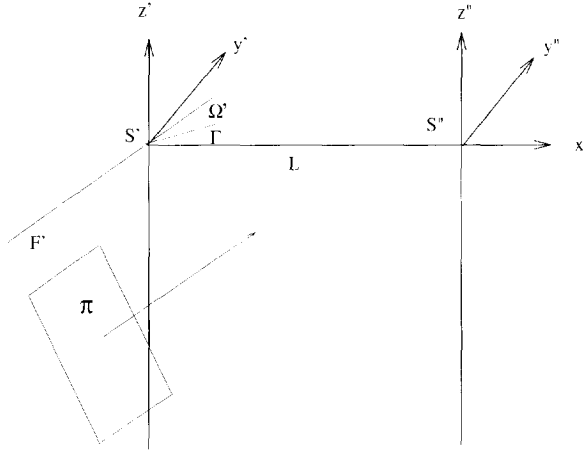


Figure 4.2: Ideal configuration of stereo camera: when object plane is nearly vertical.

obvious that  $\Omega' = 90^\circ - \Omega$ . In the figure,  $F'$  is the intersection point of the object plane ( $\pi$ ) with  $\vec{n}$  (passing through  $S'$ ). The distance between the  $S'$  and  $F'$  is denoted as  $C'$ .

The equation of the object plane in this case can be written as:

$$\cos(\Gamma)X' + \sin(\Gamma)Y' + \tan(\Omega')Z' = C'. \quad (4.14)$$

Substituting disparity into above equation, we get

$$d = -\frac{1}{C'}(\tan(\Omega')f_c - \cos(\Gamma)x' - \sin(\Gamma)y'). \quad (4.15)$$

In this case, the scale factor is defined as  $S'_c = 1/C'$ , and is calculated by

$$S'_c = -\frac{d}{\tan(\Omega')f_c - \cos(\Gamma)x' - \sin(\Gamma)y'} \quad (4.16)$$

### 4.3 Coordinate transformation

In the previous section, all the developments are based on the assumption that the stereo images are in an ideal position, which is usually not true in reality. A procedure called *rectification* is required to transfer the normal situation into the ideal configuration. The method for rectification can be found in photogrammetry books.

In the rectification, the five relative orientation parameters  $\varphi', \kappa', \varphi'', \omega'', \kappa''$  are assumed to be known parameters, which are usually calculated by a relative orientation procedure. The relative orientation procedure uses five or usually more correctly identified pairs of match points derived either automatically or interactively and a least-squares method to solve the five relative orientation parameters.

## 4.4 Curve matching algorithm

It can be seen from previous sections that our method of estimating slant, tilt, and the scale factor is based on disparity information. Thus a point correspondence algorithm is required. The nature of the correspondence problem allows the formulation of several constraints which are helpful in reducing the ambiguity in the number of matching alternatives. These constraints are derived from assumptions which are made about the properties of the image world, as they are reflected in the stereo imagery. Most constraints originally formulated for edge points are modified below in order to be applicable to curves.

*Shape similarity:* Two similar contours, one from each image, are likely to be the projections of the same physical event in the scene.

*Figure continuity:* Connected edges in the image probably represent the projection of a continuous curve in 3-D space. Matched contours must therefore form a continuous disparity curve.

*Ordering:* The left-to-right order of matched points along epipolar lines should be preserved.

*Uniqueness:* The uniqueness constraint requires that an item on one image may correspond to no more than one item on the other image.

*Epipolar:* Rectifying the stereo images enables matching of image points lying on pairs of horizontal lines. As a result, the two-dimensional search required in the general case of image matching, is reduced to a one-dimensional search.

Bearing these constraints in mind, an algorithm for curve matching is proposed as illustrated in Fig. 4.3. Related formulae for coordinate transformation have been described in Section 4.3. In this section, the algorithm for point correspondence and a method of weighting curve direction is discussed. The method for parameter estimation using histogram analysis as well as matching evaluation is addressed in the next section.

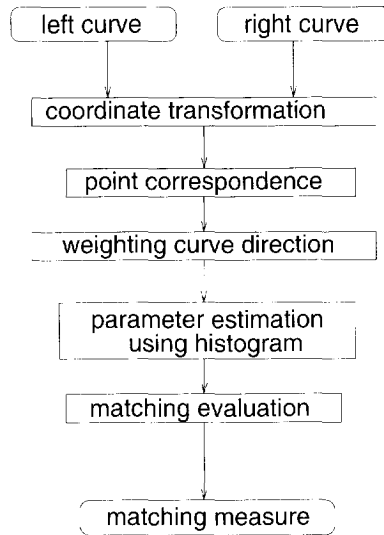


Figure 4.3: Flow chart of curve matching algorithm.

#### 4.4.1 Point corresponding for stereo curves

The general idea for finding point correspondences is summarized in Fig. 4.4. The details of the algorithm are explained below, assuming one curve as the reference curve, another as the searching curve.

1. Compare the lengths of two curves. Such a simple operation can avoid a long curve to be tested against a short curve.
2. Find corner points and common y range. In the algorithm, four corner points are found on stereo curves, namely, "left-most", "right-most", "top-most" and "bottom-most" (see Fig. 4.5). The "left-most" and "right-most" correspond to curve points having the minimum and maximum  $x$  coordinate, respectively. The "top-most" and "bottom-most" correspond to curve points having the minimum and maximum  $y$  coordinate, respectively. The  $y$  coordinates of "top-most" and "bottom-most" points on the stereo curves are used to find the common range (see Fig. 4.5). If the common range is too small, the stereo curve is likely not to be matched. The "left-most" and "right-most" are used to restrict the searching range for finding correspondence points.
3. Produce bitmap. This is a two-valued raster image, which has the same size as the

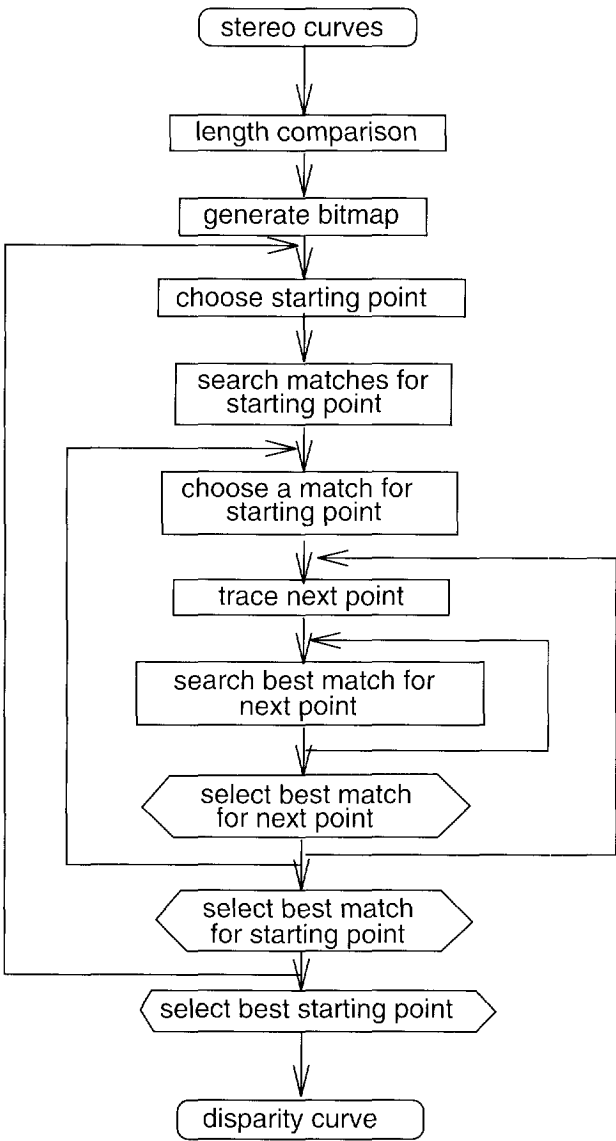


Figure 4.4: Flow chart of point corresponding algorithm.

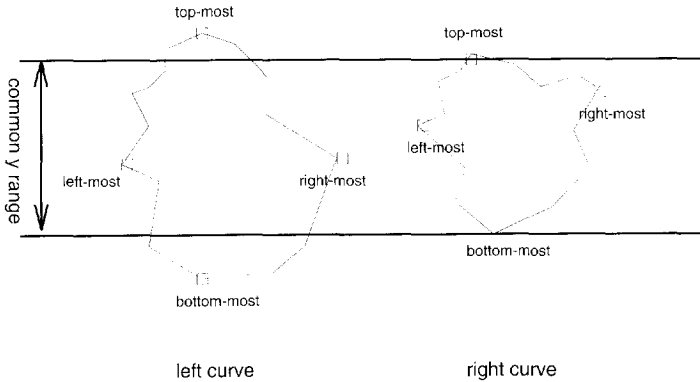


Figure 4.5: Common y range of stereo curves.

stereo images. Pixels with the searching curve on it are assigned 1, otherwise 0. This bitmap is used in the searching procedure.

4. Select a starting point from the reference curve (say the left curve) and find the candidate points on the right curve by searching its bitmap, that is, any pixel with value 1 along the scanline decided by the y coordinate of the reference point, is selected as a candidate point. If there is more than one point found, take one point and calculate the initial disparity value.
5. Trace the reference curve clock-wise. For each point, determine the corresponding points using the principle in the previous paragraph, calculate the disparity value, and choose the point pairs with the disparity value which is most similar to the previous one.
6. After all points on the reference curve have been searched, add the disparity values along the reference curve.
7. If there is more than one corresponding point for the starting point, choose another corresponding point and repeat 5-6.
8. Update the starting point on the reference curve. For each starting point, repeat 4,5,6.
9. Choose the matches which minimize the sum of the disparity values.

### 4.4.2 Weighting the curve direction

The accuracy of the disparity measurement along the curve is quite dependent on the direction of the curve at a specific position. Obviously, the disparities of near-horizontal segments are less reliable than the ones on near-vertical segments. They should be treated differently in the parameter estimation.

In order to estimate the local curve direction, we fit a straight line model to the pixels of a curve within a window, i.e.,

$$y(x) = a_e + b_e x \quad (4.17)$$

with a set of  $N_e$  data points along a curve.

The angle of an edge ( $0 - \frac{\pi}{2}$ ) within the window is then calculated by

$$\theta_e = \arctan(|b_e|). \quad (4.18)$$

Now, we calculate the weight of the curve direction by a soft thresholding function

$$w_e = \frac{1}{1 + e^{-c(\theta_e - \theta_{et})}} \quad (4.19)$$

where  $c$  is a factor controlling the slope of the soft thresholding function, and  $\theta_{et}$  is an angle thresholding value. In our algorithm we choose  $c = 5$  and  $\theta_{et} = \frac{\pi}{8}$ .

## 4.5 Parameter estimation and matching evaluation

### 4.5.1 Three-point configuration

From Section 4.2, it is easily seen that the angle parameter slant and tilt are dependent on a three-point configuration, namely, for each combination of three points on the disparity curve (the reference curve with disparity values), a parameter value can be estimated. Theoretically, for a  $N_e$ -point curve, the total number of such three-point combinations is  $N_e(N_e - 1)(N_e - 2)/6$ , which means that some modification must be made otherwise the tremendous computational load will make this approach unrealistic.

One idea is to use curve fitting or the detection of some crucial or control points which can represent a curve with a much smaller number of points. Such methods could have some difficulties: 1) if stereo curves are fitted individually before point correspondence, the resulting points on one curve may not find corresponding points on the other curve, and if we fit the disparity curve after point correspondence, the correspondence error of a point will propagate to other points, and the derived parameter values will be unreliable;

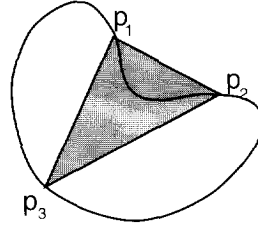


Figure 4.6: Area of three points.

2) when segmentation errors or occlusions are present, using control points only will not be sufficient to obtain correct parameter values.

In our algorithm, we use the following methods to reduce the number of computations:

1. If the direction of the edge at a point is near horizontal, the point will not be considered.
2. Take a sample of points on the disparity curve. The reduction factor will be chosen relatively small when the total length of a curve is small, and will be chosen larger when the curve is long.
3. If the the triangular area decided by the three-point is too small, no computation will be carried out. In this case, the three points will either lie on one straight line or will aggregate near one point, the calculated parameter value in this case is fragile to errors.

In Fig. 4.6, the three points are denoted as  $p_1(x'_1, y'_1)$ ,  $p_2(x'_2, y'_2)$  and  $p_3(x'_3, y'_3)$ , the area of the triangle determined by these points is calculated by

$$A_3 = \frac{1}{2} \begin{vmatrix} x'_1 - x'_3 & y'_1 - y'_3 \\ x'_2 - x'_3 & y'_2 - y'_3 \end{vmatrix} \quad (4.20)$$

### 4.5.2 Histogram analysis

Histogram analysis [101] is a useful technique in image processing to measure the popularity or frequency of a studied signal within a certain range. The technique is used here as a kind of voting mechanism to select the most likely value for the estimated parameter.

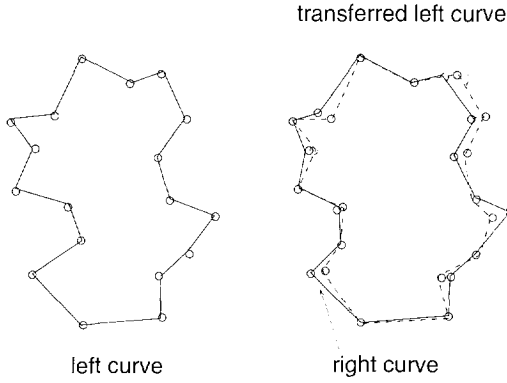


Figure 4.7: Transferring left curve into right image.

In our algorithm, a hierarchical histogram is actually adopted, which means that for each parameter, several levels of histograms are used, with the number of bins at each level fixed. The calculated parameter values are first put into corresponding bins at level 1. After all values have been computed, the peak position at that level is detected. At the next level, the same parameter values are used, but the bin width is reduced. After the computations at several levels have been finished, the peak position at the final level is used to calculate the final parameter value. The reason for using this multi-level histogram is to avoid possible multiple peaks on the histogram and to get a more accurate peak position through the refinement of the levels.

### 4.5.3 Curve normalization

The merit of the overall matching between the stereo curves is judged by the difference between the two normalized curves. By normalization in the stereo case, we mean that one curve is transferred before matching (see Fig.4.7). Suppose we transfer the left curve into the right image and denote the transferred  $x'$  coordinate as  $x^*$  (note that there is no difference between  $y$  coordinates on a correctly matched point pair). Referring to Eqs. (4.3), (4.12) and (4.15), we use the following formula

$$\begin{cases} x^* = x' + S_c(f_c + \tan(\Omega) \cos(\Gamma)x' + \tan(\Omega) \sin(\Gamma)y') & \text{if } \Omega < 45^\circ \\ x^* = x' + S'_c(\tan(\Omega)f_c - \cos(\Gamma)x' - \sin(\Gamma)y') & \text{if } \Omega' \leq 45^\circ. \end{cases} \quad (4.21)$$

#### 4.5.4 Distance transformation

The final step of the algorithm uses the distance transformation and distance measurement techniques to measure the similarity between two shapes. The distance transformation converts the pixels of a curve into a gray-level image where all pixels have a value corresponding to the nearest distance to the curve.

Because of the discrete nature of digital images and the influence of noise on the boundary points, it is unnecessary to compute exact Euclidean distances [116] from the inexact boundary pixels. In most digital image processing applications, it is preferable to use integers to represent distances. Good integer approximations of Euclidean distance can be computed by the process known as the chamfer 3/4 distance [116].

According to Borgefors [116], the chamfer 3/4 distance can be calculated sequentially by a two-pass algorithm. First, a distance image is created such that each pixel of the curve is set to zero and each non-boundary pixel is set to infinity. The forward pass modifies the distance image as follows:

```

for  $i = 2, \dots, \text{rows}$  do
  for  $j = 2, \dots, \text{columns}$  do
 $v_{i,j} = \min(v_{i-1,j-1} + 4, v_{i-1,j} + 3, v_{i-1,j+1} + 4, v_{i,j-1} + 3, v_{i,j})$ ,

```

Similarly, the backward pass operates as follows:

```

for  $i = \text{rows} - 1, \dots, 1$  do
  for  $j = \text{columns} - 1, \dots, 1$ , do
 $v_{i,j} = \min(v_{i,j}, v_{i,j+1} + 3, v_{i+1,j-1} + 4, v_{i+1,j} + 3, v_{i+1,j+1} + 4)$ 

```

where  $v_{i,j}$  is the value of the pixel at position  $(i,j)$ .

#### 4.5.5 Matching measure

In order to get a proper match measure, the searching curve is transferred into a distance image, the left normalized curve is then superimposed on this image. Defining the distance value at point  $j$  on the reference curve as  $D_j$ , the good match points are detected by thresholding the  $D_j$ , i.e.

if

$$D_j < \epsilon_d \quad (4.22)$$

then point  $j$  is a good match point. Here  $\epsilon_d$  is the distance threshold.

Let  $N_c$  be the total number of points on the reference curve,  $N'_c$  be the total number of points having good matching decided by Eq. (4.22), the match measure between the

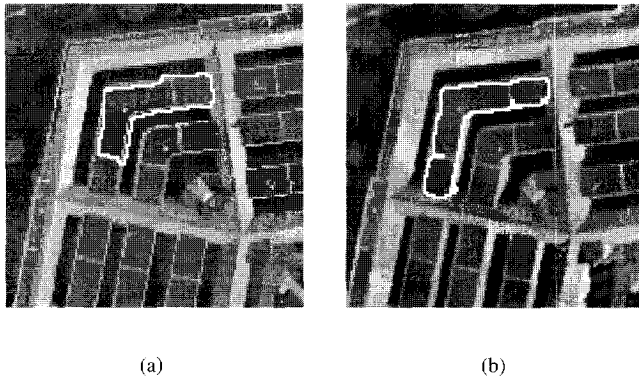


Figure 4.8: Stereo curves on stereo images.

stereo curves are expressed by

$$m_s = \frac{N'_c}{N_c}. \quad (4.23)$$

Another way of defining the matching measure is to consider the average value of  $D_j$ . But if there are some errors caused by bad matches, the average distance value becomes unreliable. The definition by Eq. (4.23) reflects what fraction of the reference curve has been matched reliably.

## 4.6 Experimental results

We first show the results from the procedures carried out on the stereo curves, shown in Fig. 4.8 (the curves are highlighted in the figure). These stereo curves result from region-based segmentation. The images are parts of the Pentagon test images and we assume they have been rectified. Given these two curves (the left curve has 177 pixels and the right one has 231), our task is to match them based on the method described in previous sections. The result for point correspondences is shown in Fig. 4.9 and Fig. 4.10 which indicate that the big errors occur along near-horizontal segments. Consequently, the weight function is calculated based on the curve direction (see Fig. 4.11). To estimate the parameter slant and tilt, a three-level hierarchical histogram analysis was used. At each level, 8 bins were used, the population of a certain value was put into the corresponding bin. A peak was detected and was refined to the next level where the width of bin was reduced by half. The angle ranges of slant and tilt are both  $90^\circ$  (estimated  $\Gamma$  value from

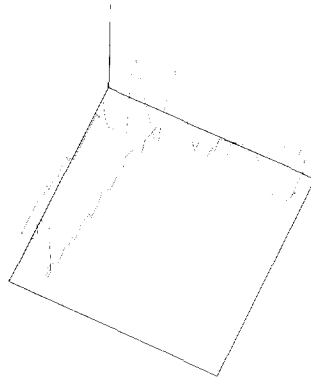


Figure 4.9: 3-D disparity curve – a perspective view.

the histogram was in the first quadrant, and the real quadrant  $\Gamma$  belongs to was decided by the signs of  $n_x, n_y$ , see Section 4.2), the finest bin on the last level of the histogram has the width of  $90/32 = 2.81250^\circ$ . The width of a bin should not be too small otherwise there will be too many local peaks on the histograms. The final position of a peak can be improved by averaging the data positions within the bin of the detected peak. When calculating slant and tilt, a four-point interval was used to sample points on the disparity curve. The calculated slant, tilt and scale factor in this case are  $75.26^\circ$ ,  $71.27^\circ$ , and  $-0.005415$ , respectively. In order to evaluate the match, the left curve was transferred into the right image by formula (4.21) as illustrated in Fig. 4.12. The result of distance transformation was given in Fig. 4.13, in which the transferred left curve was superimposed on this image, the corresponding distance value along the curve was the nearest distance between the transferred left curve and the right curve.

We tested four other examples, as shown in Figs. 4.14, 4.15, 4.16 and 4.17. In each of these figures, (a) and (b) show the left and right curves to be matched, and (c) shows the right curve and the transferred left curve using the estimated parameters. In Fig. 4.15, there are some differences between the stereo curves produced by bad segmentation; in Fig. 4.16, occlusions are present; and in Fig. 4.17, the right curve is partially out of image range. In all these examples, it is shown that our method produces acceptable results.

## 4.7 Conclusions

In this chapter, we have described a complete solution to the general stereo curve matching problem. The experiments have shown that the method can yield the acceptable results.

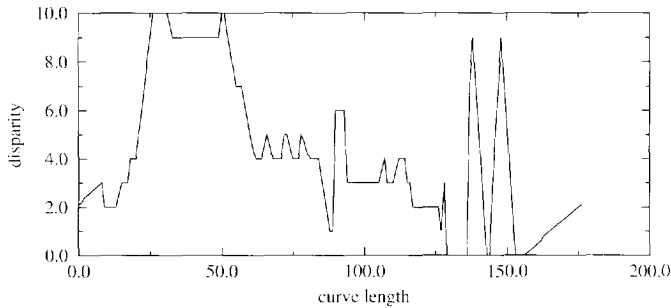


Figure 4.10: Disparity value along curve.

The method has the following advantages: 1) it is able to get a correct matching measure even when occlusions occurs and there are differences between the stereo curves due to the inconsistency of segmentation results over stereo images; 2) no assumptions are made on the mathematical property of the shape of region boundaries. The assumptions we have made are that the curves are planar, closed, and simple. The latter two assumptions (closed and simple) are easy to meet if the curves concerned come from the region based segmentation. The first assumption is met in our current experiments where the images consist of mostly block-like buildings. The extension of this work is to consider other kinds of curves, for example, the curves on polynomial surfaces.

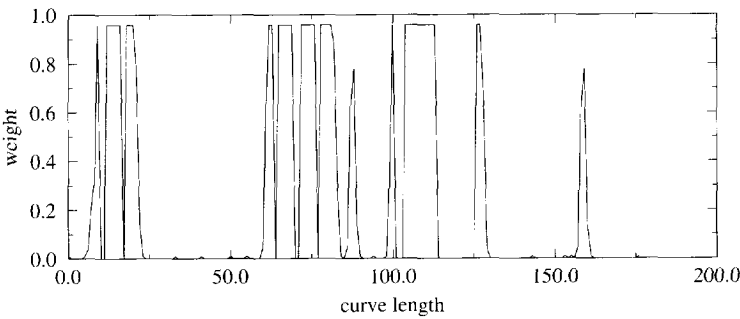


Figure 4.11: Direction weight along curve.

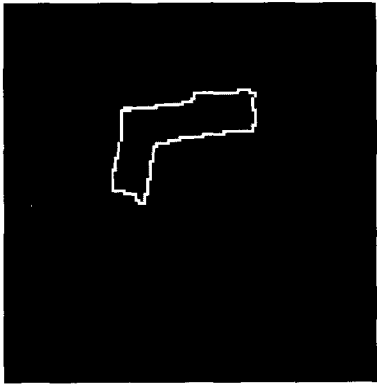


Figure 4.12: Transferred left curve.

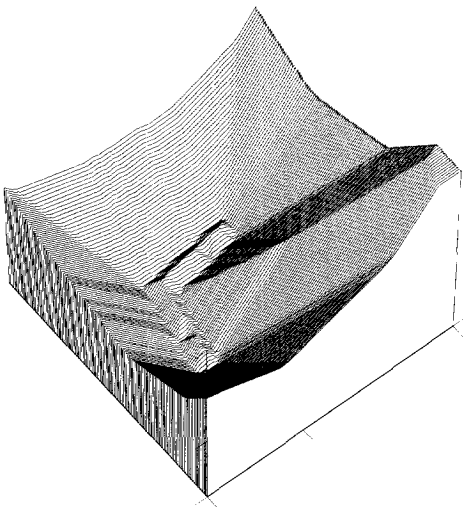
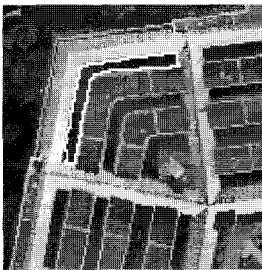
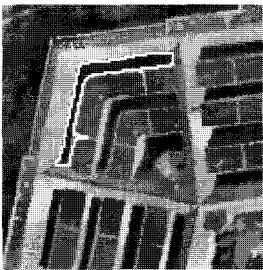


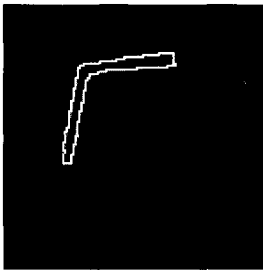
Figure 4.13: A perspective view of distance transformation result.



(a)



(b)



(c)

Figure 4.14: In this example, the left curve has 221 points, and the right curve has 237 points. The final match measure is 0.91.

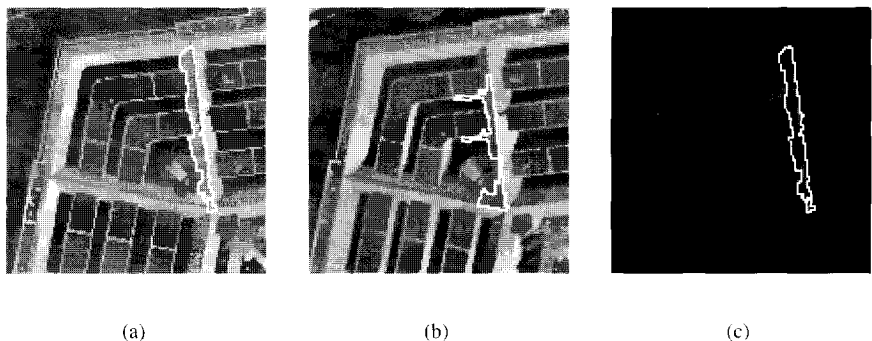


Figure 4.15: In this example, the left curve has 223 points, and the right curve has 259 points. The final match measure is 0.42.

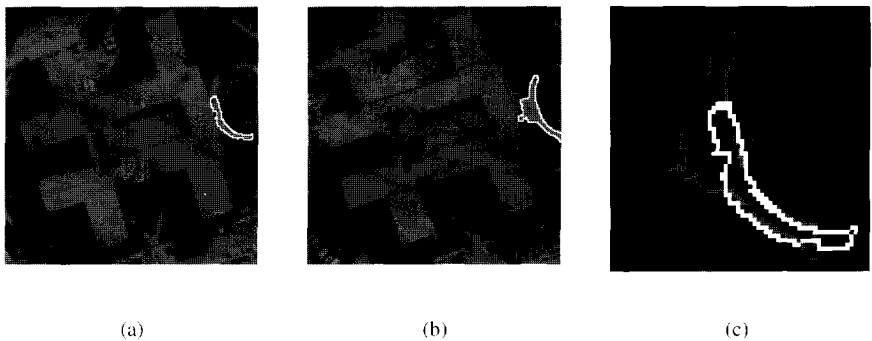


Figure 4.16: In this example, the left curve has 149 points, and the right curve has 199 points. The final match measure is 0.46.

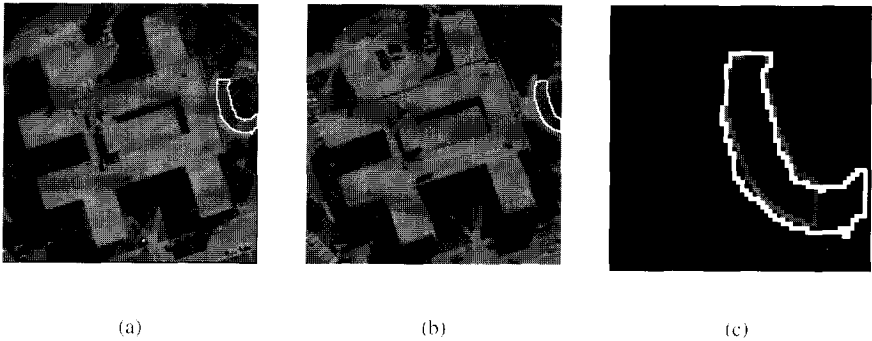


Figure 4.17: In this example, the left curve has 173 points, and the right curve has 127 points. The final match measure is 0.58.



## Chapter 5

# Fusion of Stereo Clues

### 5.1 Introduction

In the field of image processing and computer vision, we often meet the problem of combining several sources or evidences in order to make a unified conclusion. In our application, where the region matching is concerned, it is also required to consider the several features associated with each region, i.e., gray intensity (can be characterized by mean value, for example), area, shape, etc. The problem can be described precisely as follows:

$SM(R', R'')$  is denoted as the similarity measure between a region  $R'$  from one image and the region  $R''$  from another image, and each region is characterized by a number of attributes ( $R' \{x'_1, x'_2, \dots, x'_n\}, R'' \{x''_1, x''_2, \dots, x''_n\}$ ). Between each pair of attributes of two regions from two images, there is a similarity measure, denoted as  $sm(x'_i, x''_i), i = 1, \dots, n$ .  $SM(R', R'')$  can be expressed as the aggregation of  $sm(x'_i, x''_i), i = 1, \dots, n$ , that is,

$$SM(R', R'') = \mathcal{G}(sm(x'_1, x''_1), sm(x'_2, x''_2), \dots, sm(x'_n, x''_n)). \quad (5.1)$$

The objective of this chapter is to find a proper function  $\mathcal{G}$  which best suits our application.

When designing a function  $\mathcal{G}$ , we require that

**Statement 5.1:**

1.  $SM(R', R'') \geq 0$  for all  $R'$  and  $R''$ .
2.  $SM(R', R'') = SM(R'', R')$  for all  $R'$  and  $R''$  (Symmetry).
3. if  $sm_1(x'_i, x''_i) \leq sm_2(x'_i, x''_i)$  for all  $i$ ,

$$SM_1 \leq SM_2.$$

The interpretation of this rule is obvious in that when individual evidence increases, the overall evaluation should gain.

4. if  $n_1 < n_2$ , then

$$\mathcal{G}(sm(x'_1, x''_1), sm(x'_2, x''_2), \dots, sm(x'_{n_1}, x''_{n_1})) \leq \mathcal{G}(sm(x'_1, x''_1), sm(x'_2, x''_2), \dots, sm(x'_{n_2}, x''_{n_2})),$$

which says that when extra sources are considered, the final evaluation should somehow be improved.

□

The problem of finding function  $\mathcal{G}$  is related to the aggregation and fusion of visual information (in other engineering fields, the problem is usually treated as data fusion or sensor fusion). Several methods of combining evidence produced by multiple information sources have been studied by different researchers; these include Bayesian reasoning, Mycin style measures of belief and disbelief, the Dempster-Shafer structure, fuzzy logic, etc. All these techniques have the ability to deal with conflicts in evidence and to make predictions in the presence of conflicting evidence.

When choosing a proper fusion method, there are two aspects to be considered: 1) under which paradigm do we represent our problems? for example, probability theory, fuzzy logic, etc.; 2) under a particular paradigm, what kind of formula or algorithm is most suitable for the problem at hand? We can't offer a general solution, but we can offer some rules of thumb.

- The representation under a paradigm should as closely as possibly describe the characteristics of the concerned problem.
- The fusion process should reflect the structure of the real situation.
- The output of the fusion process should be easily interpreted and used in later decision making.

Bayesian inference has been widely used in engineering fields, including image processing and computer vision. Although it is simple and has been well studied, it has several major drawbacks,

- Difficulty in defining a priori likelihood.
- Complexity when there multiple potential hypotheses and multiple conditionally dependent events.

- Requirement that competing hypotheses be mutually exclusive.
- Lack of an ability to assign general uncertainty.

Dempster evidential reasoning is an extension to the Bayesian approach that makes explicit any lack of information concerning a proposition's probability by separating firm support for the proposition from just its plausibility. The rule for combining evidence is to assume independence whenever possible. Despite the approximations used, the Dempster-Shafer theory is not necessarily simple and it is not clear what a solution is according to this theory, does one maximize upper probability, lower probability or work with some more complex constraints?

The theory of fuzzy sets, with its keen ability to capture human cognition, has been a powerful tool in the process of formally representing the sophisticated relationships between the criteria inherent in multiple-criteria decision making. In our research we use the ideas of the fuzzy measure and fuzzy integral introduced by Sugeno [57] [58] for the representation and development of multiple-criteria aggregation functions. The reasons for using the fuzzy measure and fuzzy integral in our application are: 1) there is a solid theoretic background for fuzzy measure and fuzzy integral, which provides a general framework for representing and aggregating multiple criteria (it is worth mentioning that the probability measure is a special case of the fuzzy measure); 2) there is no assumption required for independent events. Tahani and Keller [117] have used the fuzzy measure and fuzzy integral in the field of computer vision with application in object recognition in forward-looking infrared imagery. In this chapter, we extend and generalize the fuzzy integral using Yager's idea [59] [60] to get different aggregation functions.

One difficulty associated with the use of fuzzy measure and fuzzy integral is that we must find a proper way to assign the fuzzy measures to the information sources. In this chapter we discuss two approaches: one is to normalize the subjectively given weights, which is called the identification of fuzzy measure (Section 5.4); another is to learn the fuzzy measure from training data (Section 5.5).

The rest of the chapter is arranged as follows. The introduction to fuzzy measure, and fuzzy integral is in Section 5.2. The identification and learning aspects of the fuzzy measure are given in Section 5.4 and Section 5.5, respectively. Section 5.6 focuses on the application of the fuzzy integral to the stereo matching problem.

## 5.2 Fuzzy measure and fuzzy integral

According to Kruse [118], the mathematical concept of fuzziness, which is a modality of uncertainty related to the subjective fuzziness of a human being, may be classified into two

categories, one is the fuzzy set proposed by Zadeh [119] and the other the fuzzy measure proposed by Sugeno [58].

Underlying the concept of the fuzzy integral is that of a fuzzy measure. In this section we describe fuzzy measures and fuzzy integrals and their properties as they relate to the information fuzzy problem.

### 5.2.1 Fuzzy measures

Fuzzy measures are the generalization of such elementary notions as the length of line segment, the area of a rectangle. A more general concept of a measure in an arbitrary abstract set can be defined [120], [121].

**Definition 5.1:** By a measurable space we mean a pair  $(X, \mathcal{W})$  consisting of a set  $X$  and a  $\sigma$ -algebra of subsets of  $X$ . A subset  $A$  of  $X$  is called measurable (or measurable with respect to  $\mathcal{W}$ ) if  $A \in \mathcal{W}$ .

□

**Definition 5.2:** A measure  $\mu$  on a measurable space  $(X, \mathcal{W})$  is a real nonnegative set function defined for all sets of  $\mathcal{W}$  such that  $\mu(\emptyset) = 0$ , and if  $\{A_i\}_{i=1}^{\infty}$  is a disjoint family of sets with  $A_i \in \mathcal{W}, i \geq 1$ , then

$$\mu\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} \mu(A_i).$$

□

It can be shown that a measure  $\mu$  has the follow properties [120]:

1.  $\mu(A) \leq \mu(B)$  if  $A \subset B$ .
2. if  $\{A_i\}_{i=1}^{\infty}$  is an increasing sequence of measurable sets, then

$$\lim_{i \rightarrow \infty} \mu(A_i) = \mu\left(\lim_{i \rightarrow \infty} A_i\right).$$

An important example of such a measure is the probability measure,  $P$ , where  $P(X) = 1$ .

Without the framework of a human reason model, the additivity hypothesis of this measure is too restrictive. In the seventies, alternative models were proposed by several researchers [122] [123] [58] [124] who all share the flowing intuitively reasonable axioms.

**Definition 5.3:** Let  $g : \mathcal{W} \rightarrow [0, 1]$  be a set function with

1.  $g(\emptyset) = 0, g(x) = 1,$
2.  $g(A) \leq g(B)$  if  $A \subset B.$
3. if  $\{A_i\}_{i=1}^{\infty}$  is an increasing sequence of measurable sets, then

$$\lim_{i \rightarrow \infty} g(A_i) = g(\lim_{i \rightarrow \infty} A_i).$$

Such a function is called a *fuzzy measure* by Sugeno [58].

□

Note that  $g$  is not necessarily additive. Axiom 3 (monotonicity) is substituted for the additivity axiom of the measure. All belief and plausibility functions (which include probability measures) are examples of fuzzy measure.

A number of interesting special families of these fuzzy measures have been introduced [59]. One family of particular interest are those based upon triangular norms [125].

**Definition 5.4:** A mapping  $T$ ,

$$T : [0, 1] \times [0, 1] \rightarrow [0, 1]$$

such that

1.  $T(a, b) = T(b, a)$
2.  $T(a, b) \geq T(c, d)$  if  $a \geq c$  and  $b \geq d$
3.  $T(T(a, b), c) = T(a, T(b, c))$
4.  $T(a, 1) = a$

is called a *t-norm*.

□

**Definition 5.5:**

A *t-conorm*  $S$  is a mapping,  $S : [0, 1] \times [0, 1] \rightarrow [0, 1]$ , which satisfies:

1.  $S(a, b) = S(b, a)$
2.  $S(a, b) \geq S(c, d)$  if  $a \geq c$  and  $b \geq d$
3.  $S(S(a, b), c) = S(a, S(b, c))$

$$4. S(a, 0) = a$$

□

We note that some examples of  $t$ -norm are

$$1. T(a, b) = \min(a, b) = a \wedge b \quad \text{min}$$

$$2. T(a, b) = ab \quad \text{Product}$$

$$3. T(a, b) = 1 - \min(1, 1 - a + 1 - b) \quad \text{Quasi-linear.}$$

Some examples of  $t$ -conorms are

$$1. S(a, b) = \max(a, b) = a \vee b \quad \text{max}$$

$$2. S(a, b) = a + b - ab \quad \text{Probabilistic sum}$$

$$3. S(a, b) = \min(1, a + b) \quad \text{Bounded sum.}$$

### 5.2.2 $g_\lambda$ – Fuzzy measures

By the nature of the definition of a fuzzy measure  $g$ , the measure of union of two disjoint subsets cannot be directly computed from the component measures. In the light of this, Sugeno [58] introduced the  $g_\lambda$ -fuzzy measures satisfying the following additional property: for all  $A, B \subset X$  and  $A \cup B = \emptyset$ ,

$$g(A \cup B) = g(A) + g(B) + \lambda g(A)g(B), \text{ for some } \lambda > -1 \quad (5.2)$$

A  $g_\lambda$ -fuzzy measure is indeed a fuzzy measure, and the  $g_\lambda$ -fuzzy measure for  $\lambda = 0$  is a probability measure.

#### Properties of the $g_\lambda$ -fuzzy measures

Let  $X = x_1, \dots, x_n$  be a finite set and let  $g^i = g(\{x_i\})$ . The mapping  $x_i \mapsto g^i$  is called a fuzzy density function. Suppose

$$A = \{x_{i_1}, \dots, x_{i_m}\} \subseteq X.$$

Then we can write [126],

$$\begin{aligned} g(A) &= \sum_{j=1}^m g^{i_j} + \lambda \sum_{j=1}^{m-1} \sum_{k=j+1}^m g^{i_j} g^{i_k} + \dots + \lambda^{m-1} g^{i_1} \dots g^{i_m} \\ &= \left[ \prod_{x_i \in A} (1 + \lambda g^i) - 1 \right] / \lambda, \quad \lambda \neq 0. \end{aligned} \quad (5.3)$$

Thus the value of  $\lambda$  can be found from the equation

$$g(X) = 1. \quad (5.4)$$

This is equivalent to solving the equation

$$\lambda + 1 = \prod_{i=1}^n (1 + \lambda g^i). \quad (5.5)$$

Hence if we know the fuzzy densities,  $g^i, i = 1, \dots, n$ , we can construct the  $g_\lambda$ -fuzzy measure. For the information fusion problem, we interpret the fuzzy density value  $g^i$  as the degree of importance of some source  $x$  towards the final evaluation. The measure of an arbitrary set  $A$  represents the importance degree of the set of sources denoted by  $A$  towards a final decision.

**Lemma 5.1:** For the fixed set of  $\{g^i\}, 0 < g^i < 1$ , there exists a unique  $\lambda \in (-1, +\infty)$ , and  $\lambda \neq 0$ , which satisfies (5.5) [117].

□

This lemma makes the calculation of the parameter  $\lambda$  much easier. In fact, one needs only to solve an  $(n - 1)$ st degree polynomial and find the unique root greater than  $-1$ .

**Lemma 5.2:** Let  $g_1$  and  $g_2$  be two  $g_\lambda$ -fuzzy measures with parameters  $\lambda_1$  and  $\lambda_2$  respectively. Suppose  $g_1^i \geq g_2^i$  for all  $i$ . Then  $\lambda_1 < \lambda_2$  [117].

□

**Proposition 5.1:** Let  $g_1$  and  $g_2$  be two  $g_\lambda$ -fuzzy measures with fuzzy densities  $\{g_k^i : i = 1, 2, \dots, n\}, k = 1, 2$ . Suppose  $g_1^i > g_2^i$  for  $x_j \in A \subseteq X$  and  $g_1^i = g_2^i, x_i \notin A$ . Then [117]

$$g_1(B) < g_2(B), \text{ if } B \cup A = \emptyset.$$

□

### 5.2.3 Fuzzy integrals

Using the notions of fuzzy measures, Sugeno [58] defined the concept of the fuzzy integral. Fuzzy integrals are non-linear functionals very similar to Lebesgue integrals, where the integral is defined over measurable sets [120] [121].

**Definition 5.6:** Let  $(X, \mathcal{W})$  be a measurable space and let  $h : X \rightarrow [0, 1]$  be a measurable function. The fuzzy integral over  $A \subseteq X$  of the function  $h$  with respect to a fuzzy measure  $g$  is defined by

$$\begin{aligned} \mathcal{E}(A) &= \int_A h(x) \circ g(\cdot) = \sup_{E \in \mathcal{X}} [\min(\min_{X \subseteq E} h(x), g(A \cap E))] \\ &= \sup_{\alpha \in [0, 1]} [\min(\alpha, g(A \cap F_\alpha))] \end{aligned}$$

where

$$F_\alpha = \{x : h(x) \geq \alpha\}$$

□

Let us denote

$$W(E) = \min_{x \in E} h(x)$$

Thus  $W(A)$  indicates the degree to which the alternative fulfils the criteria in the subset  $E$  of criteria. Using this substitution we get

$$\mathcal{E}(A) = \sup_{E \in \mathcal{X}} [W(E) \wedge g(A \cap E)]$$

Further we can denote

$$\mathcal{E}_E(A) = W(E) \wedge g(A \cap E),$$

with  $g(A \cap E)$  measuring the significance of the subset  $E$  of criteria.  $\mathcal{E}_E(A)$  can be seen to formalize the concept

*$E$  is a significant subset of criteria and is satisfied by alternative  $A$ .*

Finally we can write  $\mathcal{E}$  as

$$\mathcal{E}(A) = \max_E (\mathcal{E}_E(A)).$$

We see that the formulation used above formalizes the following semantic imperative

*An alternative is a good one if it satisfies a significant subset of criteria.*

One can interpret the fuzzy integral as finding the degree to which the following proposition is satisfied.

**Proposition 5.2:** There exists at least one subset of  $X$ , which satisfies the condition  $g$  and all its elements satisfy  $h$ .

□

The following properties of fuzzy integral are easy to prove [127].

1. if  $h(x) = c$ , for all  $x \in X, 0 \leq c \leq 1$ , then

$$\bigvee_X h(x) \circ g(\cdot) = c.$$

2. if  $h_1(x) \leq h_2(x)$  for all  $x \in X$  then

$$\bigvee_X h_1(x) \circ g(\cdot) \leq \bigvee_X h_2(x) \circ g(\cdot).$$

3. if  $A \subset B$ , then

$$\bigvee_A h(x) \circ g(\cdot) \leq \bigvee_B h(x) \circ g(\cdot).$$

In addition, we have established the following that is also easily proved. Let  $\{A_i : i = 1, \dots, n\}$  be a partition of the set  $X$ . Then

$$\bigvee_X h(x) \circ g(\cdot) \geq \max(\mathcal{E}_1, \dots, \mathcal{E}_n),$$

where  $\mathcal{E}_i$  is the fuzzy integral of  $h$  with respect to  $g$  over  $A_i$ .

The calculation of the fuzzy integral when  $X$  is a finite set is easily given [127]. Let  $X = \{x_1, x_2, \dots, x_n\}$  be a finite set and let  $h : X \rightarrow [0, 1]$  be a function. Suppose  $h(x_1) \geq h(x_2) \geq \dots \geq h(x_n)$ , (If not,  $X$  is rearranged so that this relation holds). Then a fuzzy integral,  $\mathcal{E}$ , with respect to a fuzzy measure  $g$  over  $X$  can be computed by

$$\mathcal{E} = \max_{i=1}^n [\min(h(x_i), g(A_i))] \quad (5.6)$$

where  $A_i = \{x_1, \dots, x_i\}$ .

Note that when  $g$  is a  $g_\lambda$ -fuzzy measure, the value of  $g(A_i)$  can be determined recursively as

$$g(A_1) = g(\{x_1\}) = g^1 \quad (5.7)$$

$$g(A_i) = g^i + g(A_{i-1}) + \lambda g^i g(A_{i-1}), \text{ for } 1 < i \leq n. \quad (5.8)$$

Thus the calculation of the fuzzy integral with respect to a  $g_\lambda$ -fuzzy measure would only require the knowledge of the density function where  $i$ th density,  $g^i$ , is interpreted as the degree of importance of the source  $x_i$ , for  $i = 1, 2, \dots, n$ .

### 5.3 Extension and generalization of the fuzzy integral

The interpretation of the fuzzy integral as the degree of satisfaction of Proposition 5.2, as stated previously, allows us to introduce alternative operations for the calculations involved.

One natural direction for the extension of the structure of the fuzzy integral is to use  $t$ -norm and  $t$ -conorm operators. In particular we see three places where operations are performed in the fuzzy integral that can be generalized using these operators.

1. The calculation of  $W(E)$  as  $\min_{x \in E} h(x)$ . We note that  $\min$  is an example of an *and* operator which emulates *all*. It is natural to consider using other  $t$ -norm operators for this calculation, thus

$$W(E, T) = T_{x \in E}(h(x)) \quad (5.9)$$

where  $T$  is  $t$ -norm.

2. Again in the calculation  $W(E) \wedge g(A \cap E)$  we can use any  $t$ -norm operator instead of  $\wedge$ . Therefore we have

$$\mathcal{E}_E(A, T) = T(W(E, T), g(A \cap E)). \quad (5.10)$$

3. Finally, in the aggregation over  $E$ , the  $\max$  operation, which emulates *at least one* or *or*, can be replaced by any  $t$ -conorm, that is,

$$\mathcal{E}(A, S) = S_E(\mathcal{E}_E(A, T)) \quad (5.11)$$

#### Results of using OWA operators

As appointed out by Yager [59], the replacement of  $\min$  by any other  $t$ -norm still is a formulation of the condition *all*  $x \in E$  satisfy  $h$ . As a matter of fact, the use of any other  $t$ -norm implements a stricter definition of *all* than  $\min$ . The alternative approach is to soften the requirements. In particular, we can change the requirement of  $W(E)$  to the proposition *most*  $x \in E$  satisfy  $h$ , where *most* is an example of a linguistic quantifier of the type introduced by Zadeh [128]. In [129] Yager introduced the OWA operators which provide a formal mechanism for implementing these kinds of quantifier guided aggregations. Yager and Filv introduced two special families of OWA operators which are useful for extending the fuzzy integral. The first family is called the S-OWA-AND operators, defined as

$$F_\alpha(a_1, a_2, \dots, a_n) = \frac{1 - \alpha}{n} \sum_i a_i + \alpha \min_i [a_i]. \quad (5.12)$$

These S-OWA-AND operators provide for *and-like* aggregations. The parameter  $\alpha$  lies in the unit interval. The closer  $\alpha$  is to one the more *and* like the aggregation. We now see that in the formulation for the fuzzy integral we can replace  $W(E)$  by

$$W_\alpha(E) = \frac{1 - \alpha}{\text{Card}E} \sum_{x \in E} h(x) + \alpha \min_{x \in E} h(x). \quad (5.13)$$

The second operator introduced by Yager and Filv is called the S-OWA-OR operator. This operator is defined such that

$$\tilde{F}_\beta(a_1, a_2, \dots, a_n) = \frac{1 - \beta}{n} \sum_i a_i + \beta \max_i [a_i]. \quad (5.14)$$

This provides for an *or-like* aggregation. Here again the parameter  $\beta$  lies in the unit interval and the closer  $\beta$  is to 1 more like a pure *or* the operation. These S-OWA-OR operators can be used to provide a further generalization of the fuzzy integral. With  $N$  the cardinality of  $X$  we can change the aggregation to

$$\tilde{\mathcal{E}} = \frac{1 - \beta}{2^N} \sum_{E \subset X} \mathcal{E}_E + \beta \max_{E \subset X} \mathcal{E}_E. \quad (5.15)$$

## 5.4 Identification of fuzzy measure

To be able to use fuzzy measures in practice we should learn to identify them. By identification we mean a procedure of fitting the subjectively specified measure data to the conditions of a fuzzy measure. Here we concentrate on a  $g_\lambda$ -measure, this procedure may be stated as follows. Let  $X = \{x_1, x_2, \dots, x_n\}$  and let  $w(E), E \subset X$ , be the estimates of a  $g_\lambda$ -measure (subjectively given). We are looking for fuzzy densities  $g_i = g(\{x_i\}), i = 1, 2, \dots, n$  with corresponding  $\lambda$  such that the following index is minimal:

$$\mathcal{J}^2 = \sum_{E \subset X} (w(E) - g(E))^2. \quad (5.16)$$

Here the values of  $g(E)$  are calculated from the  $g_i$ 's by using the  $\lambda$ -rule.

The first attempt to solve this problem was made by Sekita and Tabata in [130] where the authors proposed the SUMT method. In [131], Wierchcon proposed another method that uses the relation between probability measure and  $g_\lambda$ -measure. In order to introduce his algorithm, let us list the symbols in uses.

- $X$  - nonempty, finite set;
- $\mathcal{P}(X)$  - power set of  $X$  without set  $\emptyset$  and  $X$ ;

$n$	- cardinality of $X$ ;
$m$	- cardinality of $\mathcal{P}(X)$ ;
$x_i$	- member of $X, i = 1, 2, \dots, n$ ;
$X_j$	- member of $\mathcal{P}(X), j = 1, 2, \dots, m$ ;
$k_j$	- cardinality of $X_j$ ;
$g^i = g(\{x_i\})$	- $g_\lambda$ density function;
$G_j$	- $g_\lambda$ -measure of a subset $X_j$ ;
$p^i$	- probability mass function;
$P_j$	- probability measure of subset $X_j$ ;
$w_j$	- subjective measure of a subset $X_j$ ;
$d_{ij}$	- characteristic function equal to 1 iff $x_i \in X_j$ and equal to 0 iff $x_i \notin X_j$ .

We assume the following convention:  $w_j \in (0, 1), j = 1, \dots, m$  and if  $X_j \subset X_l$  then  $w_j \leq w_l$ . Moreover

$$G_j = \frac{1}{\lambda} \left[ \prod_{x_i \in X_j} (1 + \lambda g^i) - 1 \right], \quad P_j = \sum_{x_i \in X_j} p^i.$$

---

**Algorithm 5.1** : the program for the identification of fuzzy measure consists of the following steps:

1. read the data, i.e. cardinality  $n$  of a set  $X$  and subjective weights  $w_j, j = 1, 2, \dots, 2^n - 2$ ;
2. find  $\lambda$  solving the following equation (see the discussion following this algorithm) :

$$\prod_{j=1}^m (1 + \lambda w_j)^{k_j} = c^{[2^{n-2}(n+1)] - n}. \quad (5.17)$$

3. compute the values  $z_i$  using the following equation:

$$z_i = \log_c \prod_{j=1}^m (1 + \lambda w_j d_{ij}). \quad (5.18)$$

where  $c = \lambda + 1$

4. compute the values  $p^i$  based on the following equation:

$$p^i = 2^{2^{-n}(z_i + 1)} - 1 \quad (5.19)$$

5. compute the values  $g^i$  using the following equation:

$$g^i = \frac{1}{\lambda}(c^{p^i} - 1), \quad c > 0, c \neq 1. \quad (5.20)$$

In above algorithm, we need to solve the equation (5.17) for  $\lambda$ , or in other words to get the roots for

$$F_\lambda(\lambda) = \prod_{j=1}^m (1 + \lambda w_j)^{k_j} - c^{[2^{n-2}(n+1)]-n} = 0. \quad (5.21)$$

Lemma 5.2 gives a secure indication that a unique  $\lambda$  exists. We solve the  $\lambda$  by the *bisection method* [91]. The ideal is simple. Over some interval the function is known to pass through zero because it changes sign. Evaluate the function at the interval's midpoint and examine its sign. Use the midpoint to replace whichever limit has the same sign. After each iteration, the bounds containing the root decrease by a factor of two. If after  $n$  iterations the root is known to within an interval of size

$$\epsilon_{n+1} = \epsilon_n/2 \quad (5.22)$$

neither more nor less. Thus, we know in advance the number of iterations required to achieve a given tolerance in the solution

$$n = \log_2 \frac{\epsilon_0}{\epsilon} \quad (5.23)$$

where  $\epsilon_0$  is the size of the initially bracketing interval,  $\epsilon$  is the desired ending tolerance. In order to decide the original interval for  $F_\lambda(\lambda)$  function, we do a search on two sections, i.e.  $(-1,0)$  and  $(0, \infty)$ , skipping over zero.

To obtain the power set  $\mathcal{P}(X)$  we use  $m$  times subroutine *int2bin(num, n, IBIN)* which transforms an integer number *num* into its binary representation *IBIN* with dimension  $n$ . The results are stored (reversely) in the matrix *IBI* of dimension  $m \times n$ . Note that the value *IBI[j][i]* is nothing but the value of characteristic function  $d_{ij}$ . The values  $k_j = \text{card}(X_j)$  are stored in an array *IK* and its  $j$ th elements is found as

$$IK(j) = \sum_{i=1}^n IBI[j][i]. \quad (5.24)$$

Let's gives a numerical example. In Table 5.1 subjective weights are given (modified from [131], Table 1). Using the algorithms we obtain the solution given in Table 5.2 and Table 5.3. The calculated *IBI* matrix is shown in Table 5.4.

$X_j \in X$	$w(X_j)$	$X_j \in X$	$w(X_j)$
1	0.57	5	0.06
2	0.26	15	0.23
12	0.44	25	0.32
3	0.22	125	0.91
13	0.40	35	0.28
23	0.49	135	0.46
123	0.68	235	0.96
4	0.21	1235	0.75
14	0.39	45	0.28
24	0.48	145	0.46
124	0.67	245	0.55
34	0.44	1245	0.74
134	0.63	345	0.51
234	0.72	1345	0.70
1234	0.93	2345	0.80

Table 5.1: Input data (subjective weights);  $2345 = \{x_2, x_3, x_4, x_5\}$ .

	$\{x_1\}$	$\{x_2\}$	$\{x_3\}$	$\{x_4\}$	$\{x_5\}$	Quality index
$p_i$	0.228	0.327	0.231	0.168	0.107	0.012

Table 5.2: Output data (identified densities);  $\lambda = -0.1451$ .

$X_j \in X$	$w(X_j)$	$X_j \in X$	$w(X_j)$
1	0.23	5	0.11
2	0.33	15	0.33
12	0.54	25	0.43
3	0.23	125	0.64
13	0.45	35	0.33
23	0.54	135	0.55
123	0.75	235	0.64
4	0.16	1235	0.85
14	0.39	45	0.27
24	0.48	145	0.49
124	0.70	245	0.59
34	0.39	1245	0.80
134	0.61	345	0.49
234	0.70	1345	0.71
1234	0.91	2345	0.80

Table 5.3: Identified fuzzy measures.

<i>IBI</i>	$k_j$	<i>IBI</i>	$k_j$
10000	1	00001	1
01000	1	10001	2
11000	2	01001	2
00100	1	11001	3
10100	2	00101	2
01100	2	10101	3
11100	3	01101	3
00010	1	11101	4
10010	2	00011	2
01010	2	10011	3
11010	3	01011	3
00110	2	11011	4
10110	3	00111	3
01110	3	10111	4
11110	4	01111	4

Table 5.4: *IBI* matrix.

## 5.5 Learning of fuzzy measure

It is obvious that the accuracy of the fuzzy measure plays an important role in the success of fusion process. In the following, we design a method to get a more accurate fuzzy measure  $g(\cdot)$  from training data. More precisely, the problem can be formulated as:

Assume that there are  $M$  examples in the training data. Let  $h_j(x)$ , ( $1 \leq j \leq M, x \in X$ ) be a partial evaluation of  $j$  example to source  $x$ , and  $\mathcal{E}_j$  the overall one obtained from equation (5.11). Showing a person the sources and their partial evaluations, ask him/her to express his/her subjective overall evaluation  $\mathcal{D}_j$ 's for  $1 \leq j \leq M$  in  $[0,1]$ . Define  $\mathcal{E}^+ = \max\{\mathcal{E}_j\}$ ,  $\mathcal{E}_- = \min\{\mathcal{E}_j\}$  and similarly  $\mathcal{D}^+, \mathcal{D}_-$ . By normalizing  $\mathcal{E}_j$  so that  $\mathcal{E}^+ = \mathcal{D}^+$  and  $\mathcal{E}_- = \mathcal{D}_-$ , we have for  $1 \leq j \leq M$

$$\bar{\mathcal{E}}_j = \frac{\mathcal{D}^+ - \mathcal{D}_-}{\mathcal{E}^+ - \mathcal{E}_-} \mathcal{E}_j + \frac{\mathcal{D}_- \mathcal{E}^+ - \mathcal{D}^+ \mathcal{E}_-}{\mathcal{E}^+ - \mathcal{E}_-} \quad (5.25)$$

Define

$$\kappa^2 = \frac{1}{M} \sum_{j=1}^M w_j (\mathcal{D}_j - \bar{\mathcal{E}}_j)^2 \quad (5.26)$$

where  $w_j$  is a weight factor.

We can calculate a person's fuzzy measure by finding  $g$  which minimizes  $\kappa^2$ .

Generally, an analytical solution for minimizing  $\kappa^2$  is very difficult because the

complexity of the fuzzy integral. Instead, we treat this problem under the paradigm of optimization, which is shown in the next section.

## 5.6 Fusion of stereo clues

With previously discussed theories and methods, we can now treat our problem of fusing stereo clues. We use the following attributes: 1) gray mean difference; 2) area difference; 3) shape difference. We describe the fusion process in terms of the fuzzy measure and fuzzy integral as follows.

Let  $X = \{x_g, x_a, x_s\}$  denote sources ( $x_g$  for gray mean difference;  $x_a$  for area difference;  $x_s$  for shape difference). Let  $sm(x) \in [0, 1]$  denote the partial evaluation for the matching when source  $x \in X$  is considered (i.e.  $sm(x_g)$  for gray mean difference;  $sm(x_a)$  for area difference; and  $sm(x_s)$  for shape difference) and let  $g(\{x\})$  denote the degree of importance of this source. Then the fusion of stereo clues can be expressed by a fuzzy integral

$$SM = G = \int_X sm(x) \circ g(\cdot). \quad (5.27)$$

In order to get a match measure for combining the three sources, we have to specify the  $sm(x)$  and  $g(\{x\})$ .

First, we use the concept of fuzzy sets to quantify  $sm(x)$ . For example, for the attribute of gray mean, the following formula is used to calculate  $sm(x_g)$ :

$$DF_g = |GM_l - GM_r|, \quad (5.28)$$

and

$$sm(x_g) = \frac{1}{1 + e^{c_g(DF_g - TH_g)}} \quad (5.29)$$

where  $GM_l, GM_r$  represent the left and right gray mean value, respectively,  $DF_g, TH_g, c_g$  stand for gray mean difference, threshold of gray mean difference, and coefficient, respectively. In Fig.5.1, the matching function based on the gray mean difference of stereo regions is illustrated with  $c_g = 0.5, TH_g = 25.0$ .

We have similar equations for  $SM(x_a)$  and  $SM(x_s)$ ,

$$DF_a = |AR_l - AR_r| / \max(AR_l, AR_r), \quad (5.30)$$

and

$$sm(x_a) = \frac{1}{1 + e^{c_a(DF_a - TH_a)}} \quad (5.31)$$

where  $AR_l, AR_r$  represent left and right area value, respectively, and  $DF_a, TH_a, c_a$  stand

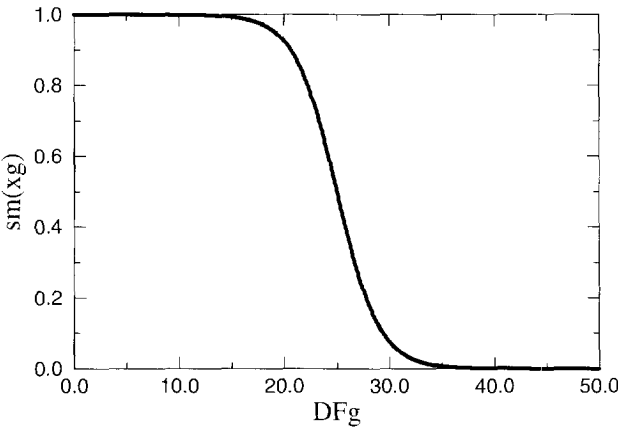


Figure 5.1: Matching function for the difference of region gray means.

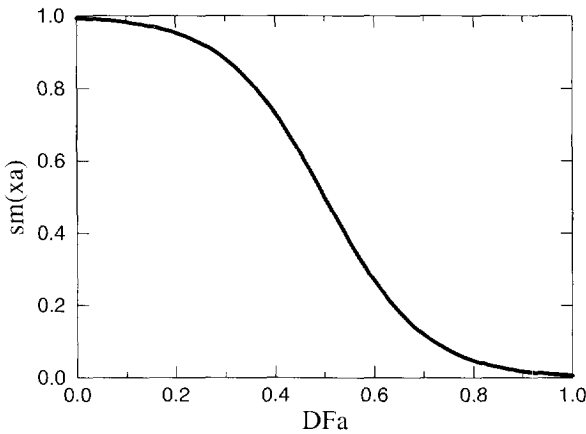


Figure 5.2: Matching function for the difference of region areas.

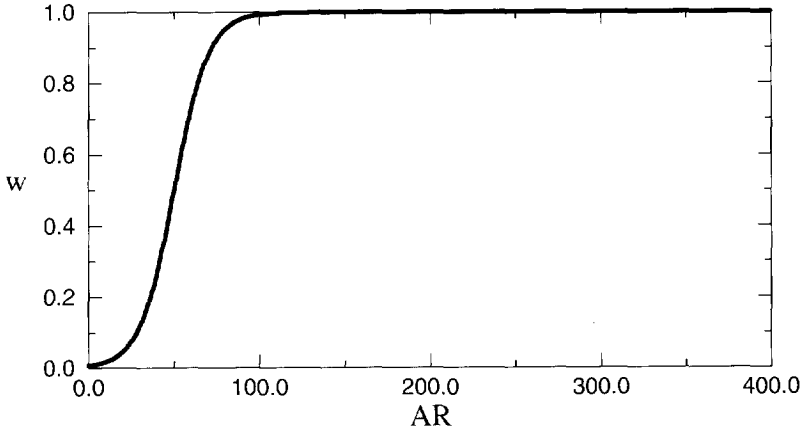


Figure 5.3: Weight function based on region area.

for area difference, threshold of area difference, and coefficient, respectively. In Fig.5.2, the matching function based on the area difference of stereo regions is illustrated with  $c_a = 10.0$ ,  $TH_a = 0.5$ .

For the shape attribute, we have

$$DF_s = 1 - m_s \quad (5.32)$$

and

$$sm(x_s) = \frac{1}{1 + e^{c_s(DF_s - TH_s)}} \quad (5.33)$$

where  $DF_s$ ,  $TH_s$ ,  $c_s$  stand for shape matching measure, threshold, and coefficient, respectively, and the method for shape matching is described in the previous chapter, and  $m_s$  is calculated by (4.23) where  $c_s = 10.0$ ,  $TH_s = 0.5$ .

In Eq.(5.26),  $w_j$  is used to assign different weights to each training example. In the case of stereo matching, this weight is assigned to the areas of the stereo regions, and is calculated by

$$w = \frac{1}{1 + e^{-c_w(AR - TH_{a1})}} \quad (5.34)$$

where

$$AR = \frac{|AR_l + AR_r|}{2} \quad (5.35)$$

The weight function with  $c_w = 0.1$ ,  $TH_{a1} = 50$  in Eq.(5.34) is shown in Fig.5.3.

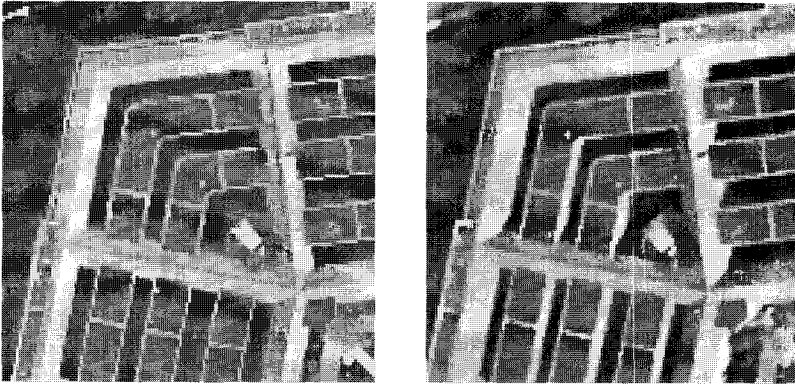


Figure 5.4: First training stereo images.



Figure 5.5: Second training stereo images.

There are two ways to get the fuzzy measures for individual information source: 1) subjectively assign a weight to each source and use identification procedure to calculate the fuzzy measures; 2) use a learning procedure to get more accurate fuzzy measures from training examples. The latter method is used for fusing the stereo clues. In order to get training examples, we first perform segmentation over stereo images, for a selected pair of stereo regions, assign a subjective matching metric. The selected stereo regions, their local properties, and the subjective matching metric are used in the learning process. When we collect the training data sets, we consider several aspects: 1) the data sets are from several different stereo images; 2) only individual regions are displayed to exclude the topological information when asking a person to assign a subjective matching value.

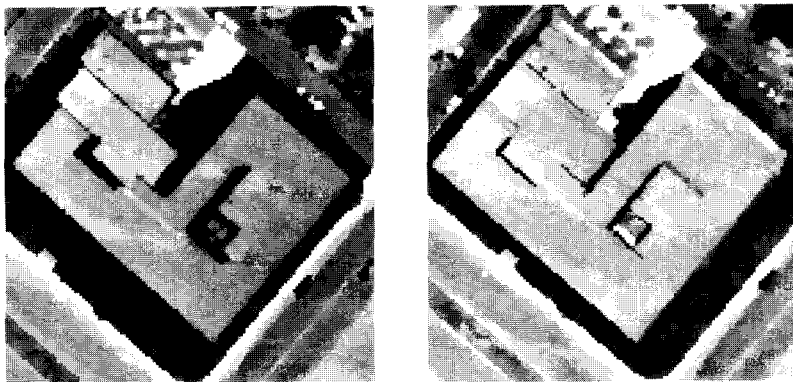


Figure 5.6: Third training stereo images.

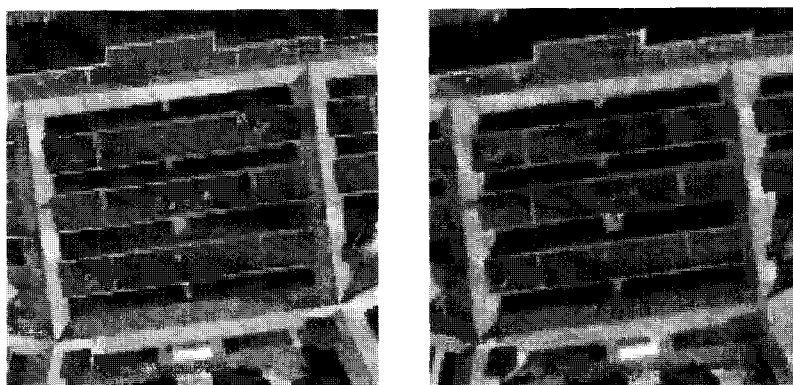


Figure 5.7: Stereo images for testing.

$T1$	$T2$	$S1$	$\alpha_1$	$\alpha_2$	$\beta$	$g^1$	$g^2$	$g^3$	$\lambda$	fitting error	test error
1	1	0	1.0	1.0	1.0	0.10	0.22	0.86	-0.6309	0.0497	0.0416
1	1	1	1.0	1.0	1.0	0.02	0.20	0.48	2.6172	0.0507	0.0400
2	1	1	1.0	1.0	1.0	0.02	0.20	0.40	3.8700	0.0479	0.0388
1	2	1	1.0	1.0	1.0	0.02	0.04	0.06	97.1142	0.0422	0.0321
1	1	2	1.0	1.0	1.0	0.02	0.98	0.04	-0.6771	0.1458	0.1247
3	1	1	1.0	1.0	1.0	0.02	0.18	0.98	-0.9166	0.0545	0.0399
1	3	1	1.0	1.0	1.0	0.06	0.06	0.02	75.2972	0.6153	0.6209
1	1	3	1.0	1.0	1.0	0.02	0.06	0.10	52.7862	0.0815	0.0847
1	1	1	1.0	1.0	0.5	0.12	0.18	0.98	-0.9473	0.0488	0.0385
1	1	1	1.0	0.5	1.0	0.02	0.28	0.40	2.2946	0.0474	0.0415
1	1	1	0.5	1.0	1.0	0.02	0.18	0.78	0.1250	0.0616	0.0506
1	2	1	1.0	1.0	0.0	0.02	0.22	0.16	11.7483	0.0464	0.0435
1	2	1	0.0	1.0	1.0	0.02	0.16	0.16	16.3486	0.0644	0.0623

Table 5.5: The result of the learning process.

## 5.7 Experimental results

In our current experiment, we use three stereo images to collect the training examples and other stereo images to get testing data. They are shown in Figs.(5.4), (5.5), (5.6), and (5.7). The learning of fuzzy measures or minimizing function (5.26) is solved in a simple way. If the number of information sources is relatively small, we can select a number of values for each fuzzy measure, then perform an exhaustive searching to find the best combination. The main advantage of this method is that it is very simple and may need only a few lines of C code to carry out the task. Another merit is that the found values are guaranteed to be near the global minimum. The disadvantage is also very obvious that the computation process is slow.

In the experiment, we collected 368 pairs of stereo regions for learning and 50 pairs for testing. The result of the learning process is listed in Table 5.5, where  $T1$  indicates the type of  $t$ -norm used in Eq.(5.9),  $T2$  indicates the type of  $t$ -norm used in Eq.(5.10), and  $S1$  is the type of  $t$ -conorm used in Eq.(5.11). The interpretation for different types of  $t$ -norm and  $t$ -conorm can be found on page 82.  $S1 = 0$  means a linear function for fusion as follows:

$$SM = g^1 h(x_g) + g^2 h(x_a) + g^3 h(x_s). \quad (5.36)$$

$\alpha_1$  is the  $\alpha$  value used in the S-OWA-AND operator (Eq.(5.13)) for Eq.(5.9), and  $\alpha_2$  is the  $\alpha$  value used in the S-OWA-AND operator for Eq.(5.10),  $\beta$  is the value used in the S-OWA-OR operator (Eq.(5.13)) for Eq.(5.11).  $g^1, g^2, g^3, \lambda$  are the calculated fuzzy measures and corresponding  $\lambda$  value. The fitting error and testing error in Table 5.5

represent the  $K^2$  in Eq. (5.26) for training and testing data, respectively.

## 5.8 Conclusions

The experiment result in Table 5.5 reveals several things:

- For both training and testing data,  $T1 = 1, T2 = 2, S1 = 1, \alpha_1 = 1.0, \beta = 1.0$  gives the best result. These parameters are used in our matching algorithm.
- Shape is the most important attribute in the matching.
- From Eq.(5.3), we can see that when  $\lambda = 0$ ,  $g(A) = \sum_{j=1}^m g^j$ , which can be interpreted that the several sources are not independent with each other. In Table 5.5, the calculated  $\lambda$  value are not near 0 in all cases, which can be interpreted as that the stereo clues (i.e. gray mean difference, area difference, and shape difference) are not independent of each other.

## Chapter 6

# Integration of Segmentation and Stereo Matching

### 6.1 Introduction

There is a standard way to design large and complex information systems as research in computational fields has shown. First we divide the system into functional components which break down the overall task into autonomous parts, and analyze these components. Then we must choose the representation of information within the subsystems and the language of communication among them. After this, the details of the systems are tested individually, in parts, and all together [132]. At least the above principle should be used in analyzing a complex information system, such as a visual system. With respect to research in vision, this principle was apparently first realized in the mid-70s at the Massachusetts Institute of Technology through pioneering research under the leadership of David Marr [2]. It was then that the foundations of modern computer vision were set with attention shifting from restrictions on the domain of application of a vision system to restrictions on visual abilities (autonomous parts). Following the lead of David Marr, much progress has been made in understanding and developing computational theories of individual modules, but we still have difficulty in analyzing a complex moving real-world scene. Constraints used in one particular module are not always reliable and even when they are reliable, they provide limited information about the scene. If we want to build a vision system that will be effective in a complex, cluttered, dynamic environment, we will need to know how to integrate visual modules. How to achieve this integration while retaining computational tractability is far from obvious. Therefore, following the Marr paradigm, we should first obtain an understanding of the computational theory that

underlies any non-ad-hoc integration [1].

The existing control mechanisms in computer vision can be summarized as follows [133]:

### Bottom-Up control

The general outline for bottom-up vision processing is:

1. Preprocess. Convert raw data into more usable intrinsic forms, to be interpreted by next level. This processing is automatic and domain-independent.
2. Segment. Find visually meaningful image objects perhaps corresponding to world objects or their parts. This process is often but not always broken up into (a) the extraction of meaningful visual primitives, such as lines or regions of homogeneous composition (based on their local characteristics); and (b) the agglomeration of local image features into larger segments.
3. Understand. Relate the image objects to the domain from which the image arose. For instance, identify or classify the objects. As a step in this process, or indeed as the final step in the computer vision program, the objects and the relations between them may be described.

In pure bottom-up organization each stage yields data for the next. The progression from raw data to interpreted scene may actually proceed in many steps; the different representations at each step allow us to separate the process into the main steps mentioned above.

Bottom-up control is practical if the input data are accurate and yield reliable and unambiguous information for the higher-level visual processes. For example, the binary images that result from engineering maps and input thresholding can often be processed quite reliably and quickly in a bottom-up mode. If the data are less reliable, bottom-up styles may still work if they make only tolerably few errors on each pass.

### Top-Down control

The mode to keep in mind in top-down perception is that of goal-directed processing. A high-level goal spawns subgoals which are attacked, again perhaps yielding sub-subgoals, and so on, until the goals are simple enough to be solved directly. A common top-down technique is “hypothesize-and-verify”, here an internal modeling process makes predictions about the way objects will act and appear. Perception becomes the verifying of predictions or hypotheses that flow from the model, and the updating of the model based

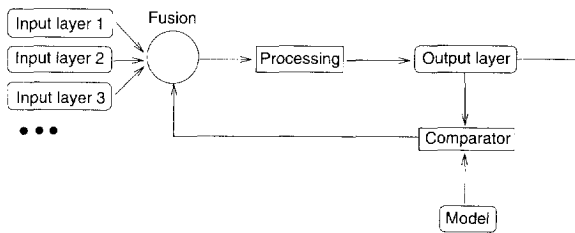


Figure 6.1: A simple feedback paradigm for computer vision.

on such probes into the perceptual environment. Of course, our goal-driven processes may be interpreted and resources diverted to respond to the interrupt. Normally, however, the hypothesis verification paradigm requires relatively little information from the lower levels and in principle it can control the low-level computations.

### Mixed Top-Down and Bottom-Up control

In actual computer vision practice, a judicious mixture of data-driven analysis and model-driven prediction often seems to perform better than either style in isolation. This meld of control styles can sometimes be implemented in a complex hierarchy with a simple pass-oriented control structure.

In our work, the feedback theory of system control has been adopted to integrate segmentation and stereo matching. The complete system for integrating segmentation and stereo matching using the feedback mechanism is described in Section 6.2. The associated algorithms are presented in Section 6.3. The experimental results are shown in Section 6.4.

## 6.2 A feedback system for integrating segmentation and matching

Most of the existing computer vision and image processing systems in the literature have the following problems:

- many systems are sensitive to the choice of the parameters;
- many systems work only for specific images and specific tasks.

We can have numerous explanations for the above problems. The several main reasons as Aloinomos pointed out [134] can be quoted here:

1. Visual objects are hard to define. Object modeling techniques have been developed in artificial intelligence and computer graphics to represent the 3-D objects, but it is still very difficult to use these techniques to describe a variety of natural objects.
2. During the image formation process the three-dimensional world is mapped into two dimensions, and one dimension is lost. This creates many problems when we try to solve the inverse (ill-posed) problem of recovering the world from the image.
3. Even well-posed (or regularized) visual computations are often numerically unstable, if noise is present in both the scene and the image. Therefore, many problems which theoretically have unique solutions become very unstable in the presence of input noise.

From our perspective, the existing vision systems are generally open-loop systems. Based on the discussion in the previous section, we can say that such systems are sensitive to the parameter changes. In our work, the concept of feedback is used to improve the performance of our processing system. In order to discuss the idea of using feedback control, in Fig. 6.1, a simple feedback architecture for image analysis is illustrated, it consists of several basic component of a visual system: 1) *input layers* of one or more than one input images (stereo, motion, multi-spectral, etc.); 2) *processing* which is the main component of a computer vision system to extract or understand desired visual information from input images; 3) *models* which explicitly express the available knowledge on input images, camera geometry, and object properties; 4) *comparator* which compares the output with the expectations or models; 5) *fusion* integrating information from various sources, including the feedback result, to improve the outcome of the visual process.

Using the above scheme, we designed a system to integrate segmentation and stereo matching as illustrated in Fig. 6.2. The system consists of three blocks, with two of them performing segmentation tasks, and one for stereo matching. In the following, the system is explained in detail.

### 6.2.1 Representation of each layer

- original image. Raster images are the most common input for the image analysis, which can be in the format of binary (2-valued), gray, or in multispectral forms. In our research, we only deal with 2-D images, not with 3-D images, such as medical images. Each pixel on a 2-D image is indexed from left to right and from top to bottom.
- segmented image. In order to interpret a 2-D image, the image is first partitioned into regions, and each region is uniform and homogeneous with respect to some criteria.

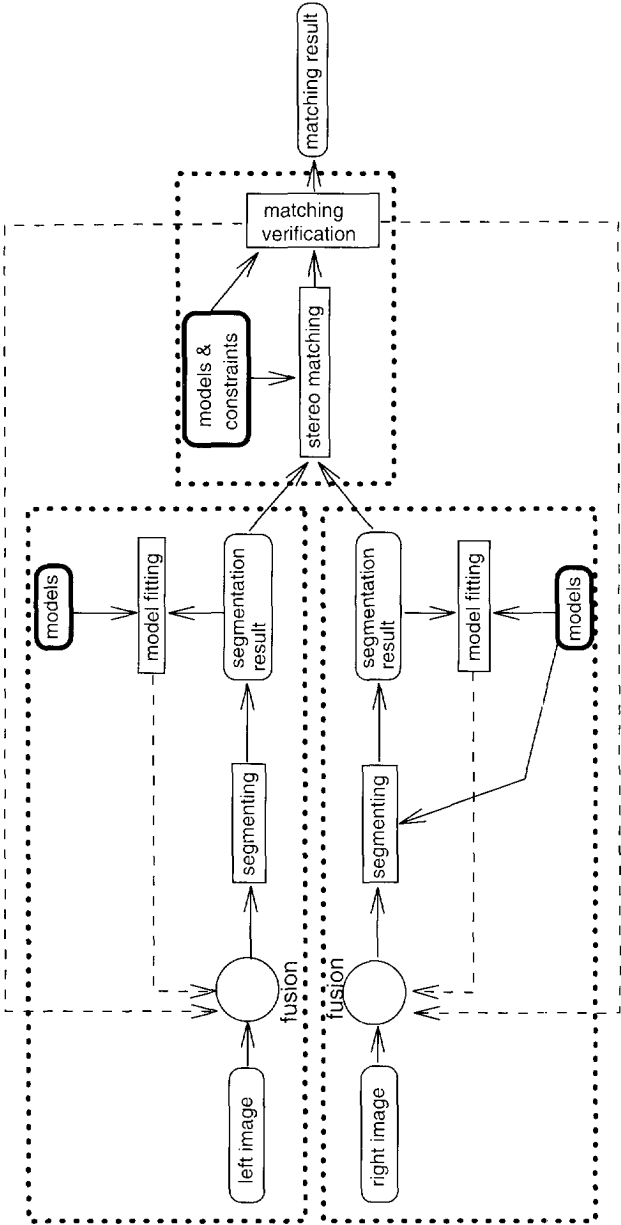


Figure 6.2: A feedback paradigm for integrating segmentation and matching.

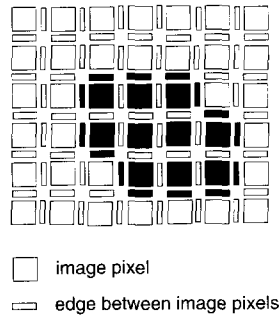


Figure 6.3: Boundary representation of a region.

For the purpose of incorporating the high-level knowledge into the segmentation, an adequate data structure should be designed to represent the segmented image, which should fulfill the requirements. 1) It should be able to be used as the linkage between the original raster image and the vector representation of objects implicitly contained on the image, which means that the data structure should represent each region directly and it should be easy to calculate every kind of properties associated with regions such as the region boundary list, area of region, intensity mean of region, etc.; 2) The data structure should be organized in a hierarchic fashion. Such a requirement is based on the observation that segmentation is an evolving procedure which usually starts from the original raster image and gradually groups small regions into more meaningful regions. During such evolution, some grouping or decision making may go wrong for a variety of reasons. Therefore it should be possible to return to a more primitive status and make a new decision. Bearing these requirements in mind, a "N-node tree" has been developed (See Chapter 2).

- vector representation. Vector representation is the critical step in shape analysis. Shape is a function of the position and direction of a simply connected curve defined within a two-dimensional field. A simply connected curve is one in which any point on the curve has at most two neighbors which lie on the curve. The coding of shape may involve a description of a closed boundary or the pixels which lie on it. We represent the region boundary using the edges between the region boundary pixels instead of the pixels themselves (See Fig 6.3).
- 2-D structural description. The individual description of each object region is often not sufficient for the final goals of many applications, because such description may be too vague for the purpose of high-level analysis such as matching and object recognition. Structure is used to represent the interrelationship of the boundaries,

which provides the basis for many kinds of tasks in computer vision, specially for matching, and object recognition purposes.

### 6.2.2 Operations on one layer

By "operations on one layer", we mean that the input and output of the operation are in the same format, e.g., from raster image to raster image, from vector data to vector data, etc. During the processing, the content of the representation may change.

- operations on the original raster images. These operations may include 1) calculating the characteristics of the image (e.g. histogram transformation, etc); 2) image quality improvement (e.g. enhancement, noise suppression by filtering, etc).
- operations on the segmented images. Merging is the main mechanism in the segmentation procedure, which merges small regions into more meaningful big regions.
- operations on the vector data. This refers to the line fitting or curve fitting algorithm, which reduces the data needed while keeping the result as close as possible to the original data. To be useful for high-level analysis, these vector data must be approximated so as to overcome local noise, and be represented in a more manageable form. The more comprehensive that representation is, the better the performance of the analysis will be.

### 6.2.3 Operations between the layers

The operations in this category change or transfer the data from one representation to another representation which are the essential parts of image analysis.

- operations between the original image and segmented image. These operations are generally called segmentation which is usually encountered on one of two forms: a) edge detection and line following. This category of techniques involves the study of various operators applied to raw images, which yield primitive edge elements, followed by a concatenating procedure to make a coherent one-dimensional feature from many local edge elements; b) Region-based methods, which depend on pixel statistics over localized areas of the image. Regions of an image segmentation should be uniform and homogeneous with respect to some characteristic such as gray tone or texture. Region interiors should be simple and without many small holes. Adjacent regions of a segmentation should have significantly different values with respect to the characteristic on which they are uniform. The boundary of each segment should be simple, not ragged, and must be spatially accurate [61].

- operations between the segmented image and vector data. This “vectorization” procedure traces along each region boundary to get the boundary position and the position is represented in chain code, which is used later in shape analysis. However, in order to integrate shape constraint into segmentation, there is another information flow which transfers the result of curve fitting into the region growing. The principle of encoding shape is described in Section 3.2.
- operations between the vector data and 2-D structural description. These operations build the structural description by performing a geometric analysis of the vector data.
- operations between the 2-D and 3-D structural descriptions. Matching two or more than two images of the same scene from different viewing positions in order to recover the three-dimensional geometry of the scene is the correspondence (stereo matching) problem.

#### 6.2.4 Data fusion

In a visual system, the comparator between the input data and feedbacked data is usually complex. In the paradigm we have proposed before, different layers have different representations, the feedback mechanism sends back certain information from high layers into lower layers. In most of cases, it is difficult to directly compare the information from different representations. The fusion process in our application consists of two goals: 1) during the segmentation stage, intensity and shape information have to be fused; 2) after stereo matching, the information from two images (or more) have to be integrated. The first fusion problem was discussed in Chapter 3 and the second problem was addressed in Chapter 5.

#### 6.2.5 Domain models

The models used in the segmentation stage were described in Chapter 2 and Chapter 3, are summarized below:

- Intensity model for regions:  
polynomial surface + Gaussian noise + outliers
- Shape model for region boundaries:  
smooth segments broken at a number of points

The models used in stereo matching can be found in Chapter 4 and Chapter 7. The summary of the models is given in the following:

- Models used in curve matching:

The curves concerned are planar, closed and simple.

- Model for intensity difference: within a certain threshold.
- Model for area difference: within a certain threshold.
- Model for position difference: obey epipolar geometry.
- Model for topology: a true surface should not be isolated. This property is used in Chapter 7 for global matching.

### 6.2.6 Computation mechanism

In *parallel computation*, several computations are done at the same time. For example, different parts of an image may be processed simultaneously. One issue in parallel processing is synchronization: Is the computation such that the different parts can be done at different rates, or must they be kept in step with each other? Usually, the answer is that synchronization is important. Another issue in parallel processing is its implementation. Animal vision systems possess the architecture to do parallel processing, whereas most computer systems are serial(although developing computer technologies may allow the practical realization of some parallel processing). On a serial computer, parallelism must be simulated where it is not always straightforward.

In *serial computation*, operations are performed sequentially in time whether or not they depend on one another. The implied sequential control mechanism is more closely matched to a (traditional) serial computer than is a parallel mechanism. Sequential algorithms must be stingy with their resources. This fact has had many effects in computer vision. It has led to the development of mechanisms for efficient data access, such as multiple-resolution representations. It has also let some to emphasize cognitive alternatives for low-level visual processing, in the hope that the massive parallel computations performed in biological vision systems could be circumvented.

## 6.3 Algorithms

The paradigm proposed in the previous section can be implemented in several ways such as parallel, serial, etc. In this section, we describe the approach to integrate segmentation and stereo matching in a sequential way. The detailed implementation for segmentation can be found in Chapter 2. Fig. 6.4 describes the procedure to integrate segmentation and stereo matching. The individual step is explained below.

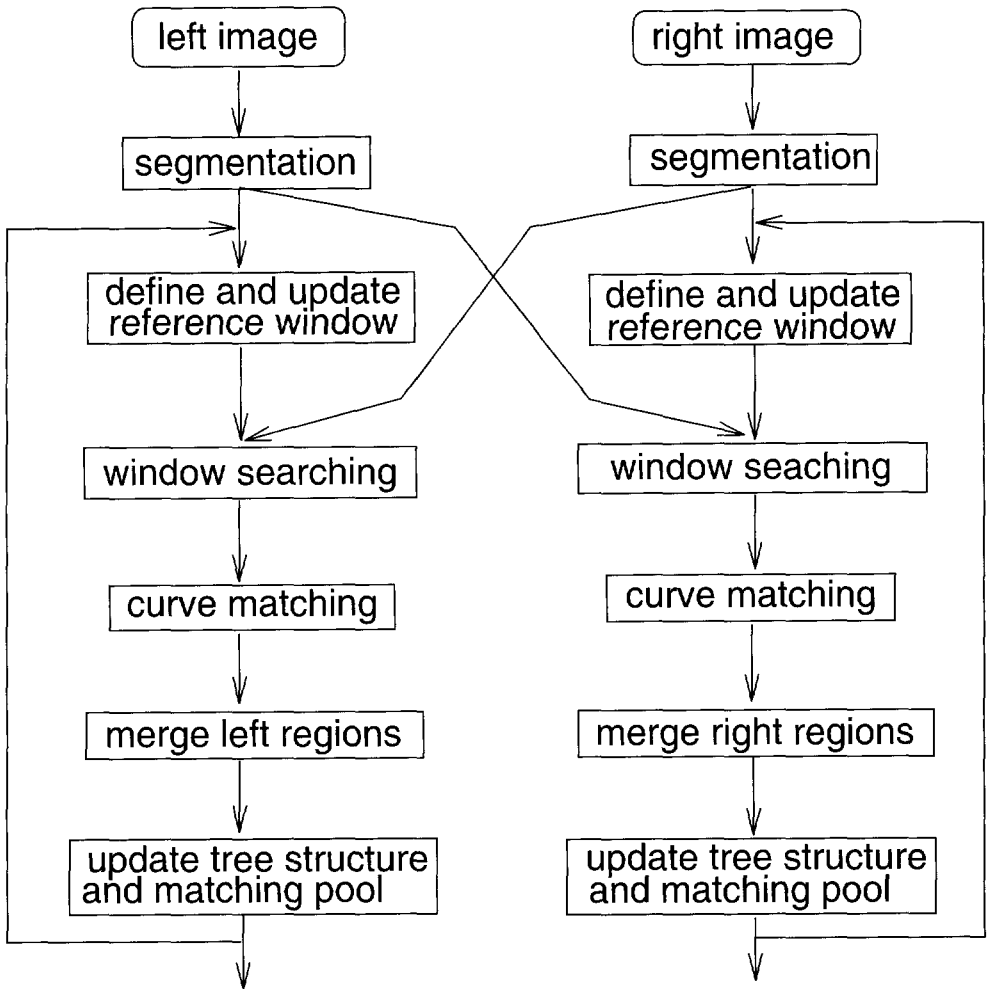


Figure 6.4: Integration of segmentation and matching.

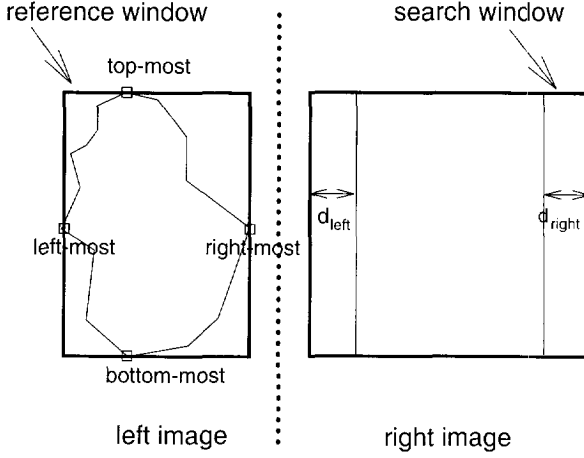


Figure 6.5: Reference window and search window.

## Segmentation

For details, see Chapter 2 and Chapter 3.

### Define and update reference window

It would be very time consuming and it is not necessary to compare each region on the left image with all the regions on the right image. The epipolar constraint and a priori knowledge on disparity will greatly reduce the search range. In Fig.6.5, for a given region, a rectangular window called *reference window* is defined by the four corner points, or in other words, by the minimum and maximum  $x', y'$  values of the region (for convenience, they are denoted as  $x'_{\min}, x'_{\max}, y'_{\min}, y'_{\max}$ , respectively). The search window is determined by  $x''_{\min}, x''_{\max}, y''_{\min}, y''_{\max}$ , which are decided by the following equations<sup>1</sup>:

$$\begin{cases} x''_{\min} = x'_{\min} - d_{\text{left}} \\ x''_{\max} = x'_{\max} + d_{\text{right}} \end{cases} \quad (6.1)$$

and

$$\begin{cases} y''_{\min} = y'_{\min} - \epsilon_{\text{top}} \\ y''_{\max} = y'_{\max} + \epsilon_{\text{bottom}} \end{cases} \quad (6.2)$$

$d_{\text{left}}, d_{\text{right}}$  are a kind of a priori values on the disparity to control the width of the search window.  $\epsilon_{\text{top}}, \epsilon_{\text{bottom}}$  are the small values to overcome possible errors in relative orientation.

<sup>1</sup>We assume that the stereo images are rectified.

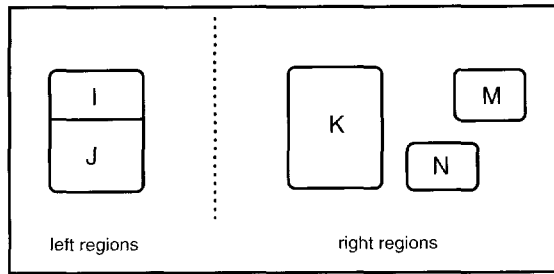


Figure 6.6: Decision on region merging based on matching.

## Window searching

The regions completely contained within the search window are selected as candidate regions. The next steps are performed to decide possible matches.

## Curve matching

The detail of the method and algorithms are described in Chapter 4.

## Merge regions using stereo information

After initial segmentation and region matching, one region in the left image may have more than one corresponding region in the right image, similarly to the regions in the right image. Before we go to the global matching to resolve the ambiguity, it is necessary to do the resegmentation. During the resegmentation, stereo information is included, that is, in considering the merging of one region with its neighboring region, the corresponding regions in the candidate pools are extracted and a unified measurement is calculated which integrates the several sources from both images. Generally, this procedure is an updating one (see Fig. 6.4), which means that when merging regions on the left image, the right image has to be static as a reference image. In the next round, the left image is used as a reference image to update the regions on the right image. After both images are updated, the new corner points have to be recalculated because some regions have changed. The candidate matching pools must also be refreshed. This procedure is updated until there are no more regions to be merged.

We now describe the integration with reference to Fig. 6.6 (we assume updating left image, keeping right one static):

1. Search candidate merging regions

For each region  $I$  on the left image, the candidate merging region  $J$  is selected based on several conditions: 1)  $I$  and  $J$  are neighboring regions; 2) their gray level difference is within a thresholding value; 3) they have same corresponding right region  $K$ ; 4) if  $I$  and  $J$  are merged, the new matching measure is largest among the other possibilities.

## 2. Merging regions

For a left region  $I$ ,  $J$  is selected candidate region to be merged, and  $K$  is their common corresponding right region.  $M$  is another right region which is a candidate match for  $I$ , similarly  $N$  for  $J$ .

$I$  and  $J$  is allowed to be merged if

$$\begin{aligned} SM(I \cup J, K) &> SM(I, \cdot) \text{ and} \\ SM(I \cup J, K) &> SM(J, \cdot) \text{ and} \\ SM(I \cup J, K) &> SM(\cdot, K) \end{aligned}$$

where  $SM(I \cup J, K)$  is the matching measure between  $K$  and the merging result of  $I, J$ ,  $SM(I, \cdot)$  is the matching measure of all possible matches for  $I$ , similar notations for  $SM(J, \cdot)$  and  $SM(\cdot, K)$ .  $SM$  is calculated by Eq.(5.27) in Chapter 5.

## Update segmented image, matching pools

Since some regions are changed, labels on the segmented image, the matching pools have to be updated. It is also necessary that the matching measures associated with match pools to be changed.

## 6.4 Experimental results

We implemented the principle and algorithms described in the previous sections. The experimental results are shown in this section. The first result is on a part of the building images as shown in Figs.6.7a and 6.7b. The segmentation result is presented in Figs.6.7c and 6.7d, and the final segmentation after improvement using stereo information is given in Figs.6.7e and 6.7f. In Figs.6.8, 6.9, 6.10, other results are shown. It can be easily seen that in the case of images shown in Fig.6.7, the segmentation has been quite significantly improved after using the integration mechanism. In other cases (e.g. Fig.6.9), the improvement is minor because the initial segmentations are quite good (in the sense that the segmentation results over the stereo images are consistent). The extent of the improvement is dependent on the difference of segmentation results over stereo images.

## 6.5 Conclusions

The performance of the region-based matching method are strongly dependent on the results of segmentation algorithms. The stereo images are the results of capturing 3-D scenes from different positions and angles, the segmentation results over the stereo images are usually inconsistent. It is very difficult and may be impossible to improve the segmentation methods for perfect segmentation results (here "perfect" means consistent segmentation results over stereo images when stereo matching is concerned). In this chapter, the method for integration of segmentation and stereo matching is proposed. The experiments have shown that the segmentation results have been improved after such integration and consequently the matching results have been improved. The extent of improvement is dependent on the difference of the segmentation results over the stereo images. It should be pointed out that the mechanism we have used in this thesis is not very complex. In order to handle complex situations, a rule-based system can be used. Nevertheless, the results show the success of the method and demonstrate that integrating different vision modules in a unified framework is a good direction to solve difficult vision problems and has potentially many practical applications.

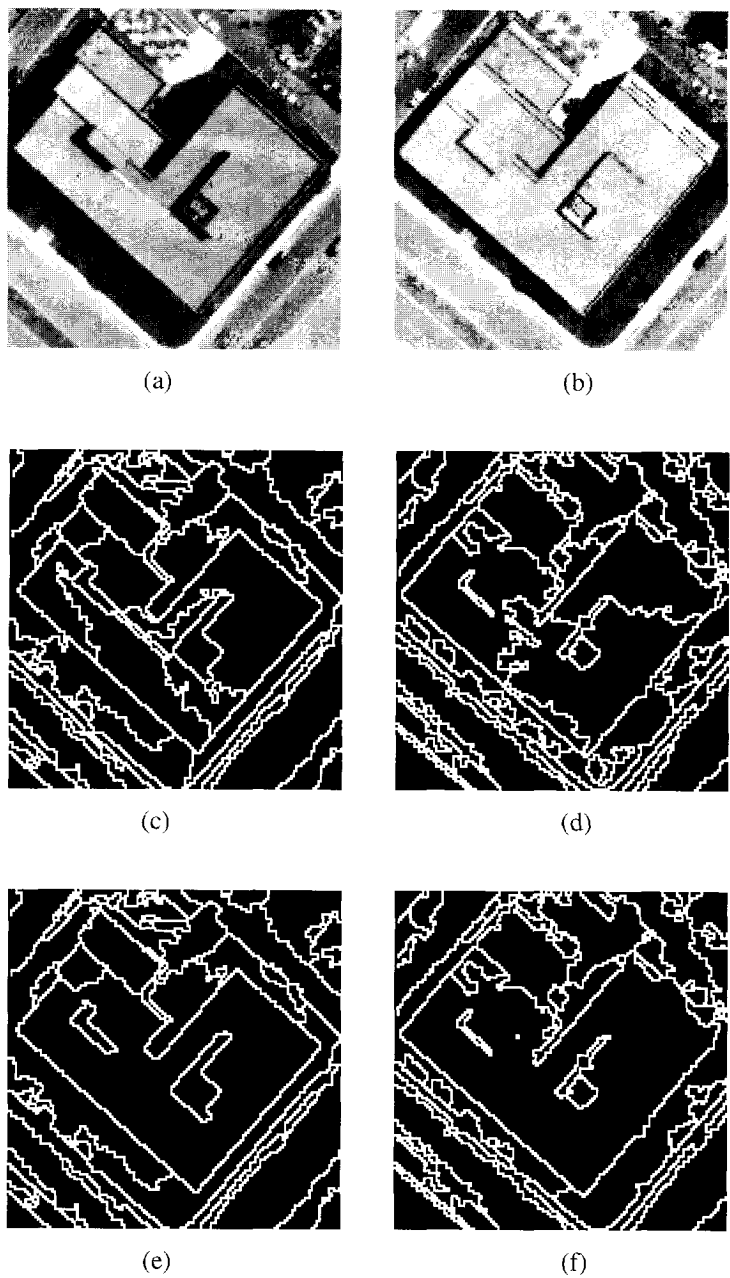
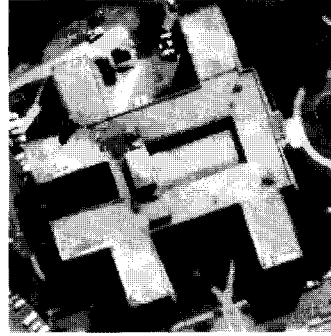


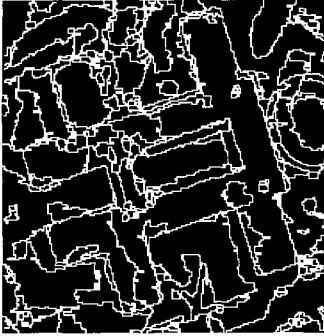
Figure 6.7: Result on a stereo image of a building, (a) and (b) are the original images; (c) and (d) are the initial segementation results; (e) and (f) are the improved segmentation results.



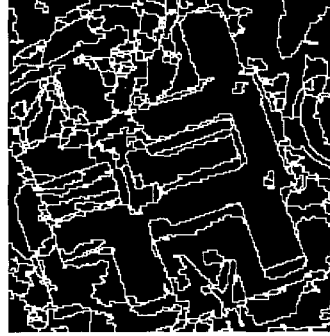
(a)



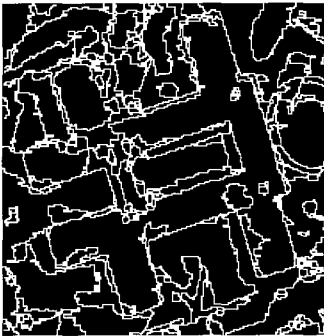
(b)



(c)



(d)



(e)



(f)

Figure 6.8: Result on a stereo image of a building, (a) and (b) are the original images; (c) and (d) are the initial segmentation results; (e) and (f) are the improved segmentation results.

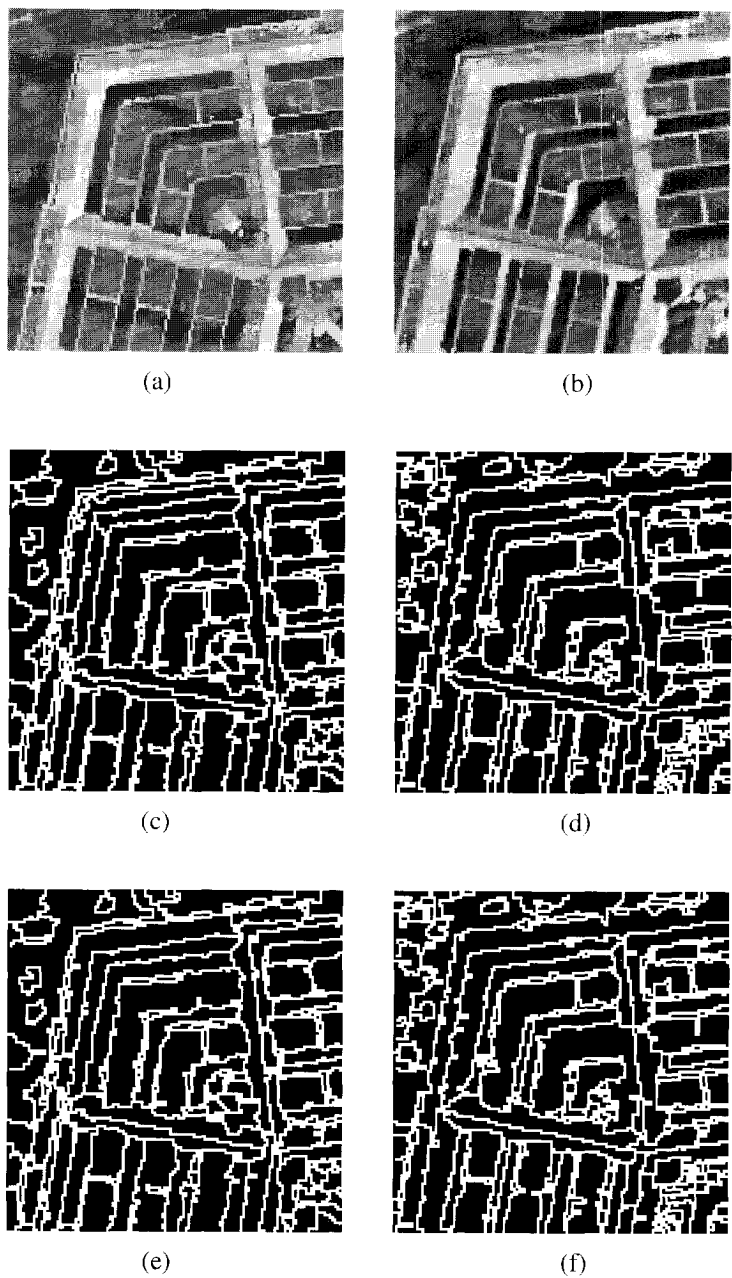


Figure 6.9: Result on a part of the Pentagon images, (a) and (b) are the original images; (c) and (d) are the initial segementation results; (e) and (f) are the improved segmentation results.

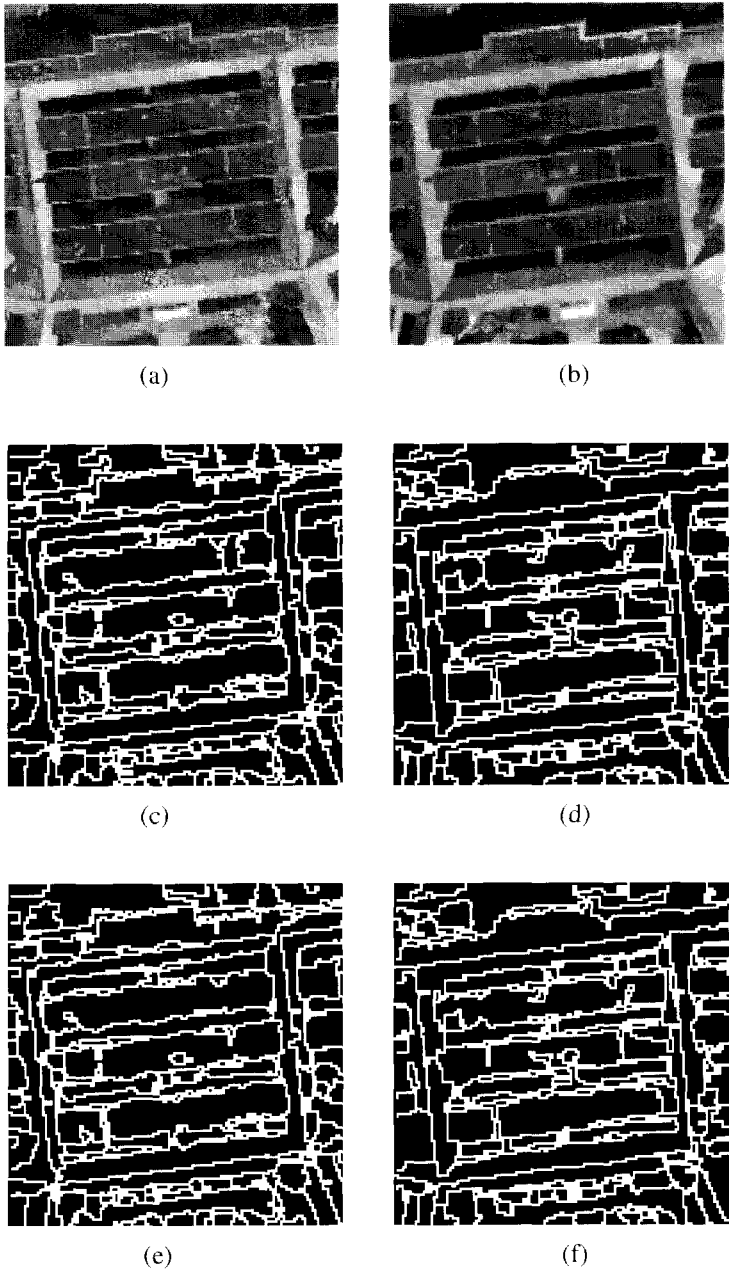


Figure 6.10: Result on another part of the Pentagon images, (a) and (b) are the original images; (c) and (d) are the initial segementation results; (e) and (f) are the improved segmentation results.

# Chapter 7

## Global Matching

### 7.1 Introduction

As described in Chapter 4, our matching algorithm consists of local matching and global matching. The local matching only compares the local attributes of stereo regions. It is possible there are multiple matches from the local matching, i.e., one region in the left image can have more than one regions assigned in the right image. The goal of the global matching is to establish unique matches, taking care of topological constraint and other matching constraints. The constraints we want to consider in the global matching consist of the following [6]:

- uniqueness: each region in the left image can only have at most one corresponding region in the right image, and similarly each region in the right image can only be assigned to at most one region in the left image.
- connectivity (continuity): object matter is cohesive, therefore disparity values change smoothly, except at a few depth discontinuities.
- ordering: object matter is not transparent, which means that two good matching pairs cannot change order.

We formulate the global matching as a relaxation problem. In the next section, we briefly introduce the relaxation labeling problem, and extend the conventional scheme to meet our requirements. In Sections 7.3 and 7.4, we give two formulations for the global matching. The gradient method to minimize the objective function is shown in Section 7.5. The algorithm for selecting unique matches is addressed in Section 7.6. The results of the experiments and conclusion are presented in Section 7.7 and Section 7.8, respectively.

## 7.2 Relaxation labeling

Many problems in image processing and computer vision can be formulated as consistent labeling problems. For example, in shape recognition, the set of measurements may comprise the Fourier descriptors of the object boundary. Such object-specific information is seldom sufficient to allow unambiguous interpretation of the object. Nevertheless, as collections of objects in the real world are invariably ordered in some sense, the constraints imposed by the mutual relationships between individual objects in ordered collections may be used to reduce or even eliminate the ambiguity.

The idea of using contextual information is far from new. Ullman [135] exploited constraints imposed by triplets of pattern primitives to reduce substantially the errors that occur with a pattern recognition system after a learning sequence of fixed length. Clowes [136] and Huffman [137] used constraints between straight line segments to eliminate nonsensical interpretations of an ideal line drawing representing a set of polyhedra. Many other methods of using contextual information have been suggested in the literature (e.g. by Zucker [138], Kittler and Foglein [139] and Haralick [140]). They broadly fall into the following categories: group classification methods, contextual decision rules, rewriting rules (grammatical or structural methods) and relaxation methods. We will concentrate on the last category, namely on relaxation labeling in general and on probabilistic relaxation labeling in particular.

The pioneering work in relaxation labeling is normally credited to Waltz [141] who considered the problem of line drawing interpretation studied earlier by Clowes [136] and Huffman [137]. His formulation of the consistent labeling problem allowed only the unambiguous interpretation of line segments. This was achieved by sequentially filtering out inconsistent label pairs on connected segments. This approach was made popular by Rosenfeld et al [142] who showed that Waltz's filtering can be carried out in parallel and could therefore be implemented as a network of processors, each associated with one object in the image. An extensive theoretic underpinning of the consistent labeling problem has been given by Montanari [143], Mackworth [144], Haralick et al [145, 146], Henderson [147], Ullmann et al [148], Shapiro [149], Nudel [150], Nishihara and Ikeda [151] and Kasif and Rosenfeld [152] and others [153–157]. An excellent survey can be found in Kittler and Illingworth's paper [158]. A lot of papers have been devoted to various aspects of relaxation labeling, such as edge and feature enhancement [159–165], object labeling [166–168], texture segmentation [169], shape and stereo matching [27, 170], computation method and architecture design [171–175], projection [176–178].

### 7.2.1 Definition of the labeling problem

In a labeling problem, one is given: 1) a set of objects; 2) a set of labels for each object; 3) a neighbor relation over the objects; and 4) a constraint relation over labels at pairs (or  $n$ -tuples) of neighboring objects.

Generally speaking, a solution to a labeling problem is an assignment of labels to each object in a manner which is consistent with respect to the constraint relation 4) above. Noting that 1) and 3) above define a graph, the problem can also be described as assigning labels to nodes in a graph.

To abstract and formalize the general situation, consider a graph with a set of labels attached to each node. We shall denote the nodes by the variable  $i$ , which can take on integer values between 1 and  $N$ , the set of labels attached to node  $i$  by  $\Lambda_i$ , and the individual label (elements of  $\Lambda_i$ ) by the variable  $\lambda$ . For the simplicity, we will assume that the number of labels at each node is  $M$ , independent of  $i$ , so that the variable  $\lambda$  takes on integer values from 1 to  $M$ . The constraint  $\Lambda_{ij}$  is the set of all pairs  $(\lambda, \lambda')$  such that label  $\lambda$  at object  $i$  is compatible with label  $\lambda'$  at object  $j$ . Label pairs in  $\Lambda_i \times \Lambda_j$  which are not in  $\Lambda_{ij}$  represent pairs of incompatible labels at the corresponding objects  $i$  and  $j$ . Constraints are only defined over neighboring nodes.

### Discrete relaxation

Discrete relaxation is accomplished by means of the label discarding rule: discard a label  $\lambda$  at a node  $i$  if there is a neighbor  $j$  of  $i$  such that every label  $\lambda'$  currently assigned to  $j$  is incompatible with  $\lambda$  at  $i$ , i.e.,  $(\lambda, \lambda') \notin \Lambda_{ij}$  for all  $\lambda'$  assigned to  $j$ . The discrete relaxation labeling process is defined by the iterative application of the label discarding rule, applied in parallel at each node, until limiting label sets are obtained. Note that the label discarding rule prescribes that a label is retained if at every neighboring node there is at least one compatible label that this property will hold for all labels in the limit sets.

### Continuous relaxation labeling processes

The constraints used in the labeling problem described in the previous subsection do not allow for labels to express a preference or relative dislike for other labels at neighboring nodes. Continuous relaxation labeling attempts to allow greater flexibility in the constraints by replacing these logical assertions about compatibilities with weighted values representing relative preferences. That is, the constraints are generalized to real-valued compatibility coefficients  $p_{ij}(\lambda/\lambda')$  signifying the relative support for label  $\lambda$  at object  $i$  that arises from label  $\lambda'$  at object  $j$ . The magnitude of  $p_{ij}(\lambda/\lambda')$  is proportional to the strength of the constraint.  $p_{ij}(\lambda/\lambda') = 1$  indicates the full support of object  $j$  under label  $\lambda'$

to assign label  $\lambda$  to object  $i$ . When there is no interaction between labels, two events of assigning  $\lambda$  to object  $i$  and assigning  $\lambda'$  to object  $j$  are said to be independent.

Having given the compatibility coefficients, continuous relaxation uses probabilities for label assignments. We denote the probability with which label  $\lambda$  is assigned to node  $i$  by  $p_i(\lambda)$ , and will require that

$$0 \leq p_i(\lambda), \quad \forall i, \lambda$$

and

$$\sum_{\lambda=1}^M p_i(\lambda) = 1 \quad i = 1, \dots, N. \quad (7.1)$$

### 7.2.2 Exactly consistent labeling

The constraints of interrelationships of objects may be interpreted as conditional probabilities  $p_{ij}(\lambda_k/\lambda_l)$ . According to Bayes rule,

$$p_{ji}(\lambda_l/\lambda_k) = \frac{p_{ij}(\lambda_k/\lambda_l)p_j(\lambda_l)}{p_i(\lambda_k)} \quad j \neq i, \quad \forall i, j, \forall k, l \text{ and } p_i(\lambda_k) \neq 0 \quad (7.2)$$

Since

$$\sum_{l=1}^M p_{ji}(\lambda_l/\lambda_k) = 1, \quad (7.3)$$

we get

$$\frac{\sum_{l=1}^M p_{ij}(\lambda_k/\lambda_l)p_j(\lambda_l)}{p_i(\lambda_k)} = 1$$

From such fact, we derive the definition for exact consistency.

**Definition 7.1:** The labeling  $p_i(\lambda_k)(i = 1, \dots, N)(k = 1, \dots, M)$  is exactly consistent proving

$$p_i(\lambda_k) = \sum_{l=1}^M p_{ij}(\lambda_k/\lambda_l)p_j(\lambda_l), \quad \forall i, \forall k \quad (7.4)$$

□

In a lot of cases, the exactly consistent labeling may not exist, the minimally inconsistent labeling, therefore, is required.

### 7.2.3 Minimally inconsistent labeling

If we can not find exactly consistent labeling, we then need to search for a solution which minimizes the inconsistency which is the difference between the probability at an object and the evidence gathered from relevant neighboring objects.

For each object  $i$ , the evidence for the assignment of label  $\lambda_k$  got from a neighboring object  $j$  is defined as

$$e_{ij}(\lambda_k) = \sum_{l=1}^M p_{ij}(\lambda_k / \lambda_l) p_j(\lambda_l) \quad (7.5)$$

Thus the combination of evidence from all the neighboring objects is

$$Q_i(\lambda_k) = \frac{1}{\sum_{j \in V_i(\lambda_k)} w_{ijk}} \sum_{j \in V_i(\lambda_k)} w_{ijk} e_{ij}(\lambda_k) \quad (7.6)$$

where  $V_i(\lambda_k)$  expresses the neighbor of object  $i$  (not including object  $i$  itself). The label assignment of object  $i$  is not independent of the assignment of objects in  $V_i(\lambda_k)$ . We stress that the objects which do have no relationship with object  $i$  should not be included in  $V_i(\lambda_k)$ .

The support function for object  $i$  for assignment  $\lambda_k$  is defined as the normalized result of  $Q_i(\lambda_k)$ , that is

$$q_i(\lambda_k) = \frac{Q_i(\lambda_k)}{\sum_{l=1}^M Q_i(\lambda_l)} \quad (7.7)$$

The measure of inconsistency at object  $i$  is defined as

$$\|\mathbf{p}_i - \mathbf{q}_i\|^2 = \sum_{k=1}^M (p_i(\lambda_k) - q_i(\lambda_k))^2, \quad i = 1, \dots, N \quad (7.8)$$

**Definition 7.2:** The labeling  $p_i(\lambda_k) (i = 1, \dots, N) (k = 1, \dots, M)$  is minimally inconsistent proving

$$\|\mathbf{p}_i - \mathbf{q}_i\|^2 = \min., \quad i = 1, \dots, N \quad (7.9)$$

□

### 7.2.4 Extension of conventional relaxation labeling

From the definitions, the solution for exactly consistent labeling and minimally inconsistent labeling only depends on the inter-relations between the objects and does not depend on the unary constraints of objects and the initial probability. In the conventional algorithms for relaxation labeling, the updating process starts from the initial probabilities, and stops when the algorithm reaches a local minimum point. In the global matching stage of stereo matching, we require that the following properties should preserve:

- the final matching result should take care of the output of local matching, topological constraint and other matching constraints.
- during the updating process, any bad match should be suppressed and possible good matches should be promoted.
- the matching of large regions is more reliable than small regions, so they need different treatment in global matching.

Based on our requirements, we extend the relaxation labeling by modifying the objective function as follows:

$$\Phi = \frac{1}{2N} \sum_{i=1}^N \bar{w}_i \gamma_i \|\mathbf{p}_i - \bar{\mathbf{q}}_i\|^2 + \frac{1}{2N} \sum_{i=1}^N \bar{w}_i (1 - \gamma_i) \|\mathbf{p}_i - \bar{\mathbf{p}}_i\|^2 \quad (7.10)$$

The global criterion now comprises two items, the first item is the consistency measure between a match and its neighboring matches, and the second is the distance between the current probability and its initial value.  $\bar{w}_i$  is a weighting factor related to each match.  $\gamma_i$  is a balance factor and is the function of initial probability.

In the above equation,

$$\|\mathbf{p}_i - \bar{\mathbf{q}}_i\|^2 = \sum_{k=1}^M (p_i(\lambda_k) - \bar{q}_i(\lambda_k))^2, \quad (7.11)$$

and  $\bar{\mathbf{q}}_i$  is a function of  $\mathbf{q}_i$ , i.e.

$$\bar{\mathbf{q}}_i = \bar{q}(\mathbf{q}_i), \quad (7.12)$$

The function  $\bar{q}(q)$  is defined as

$$\bar{q}(q) = \frac{1}{1 + e^{-c_q(q-0.5)}}. \quad (7.13)$$

It is readily seen that

$$\bar{q}(q) + \bar{q}(1 - q) = 1. \quad (7.14)$$

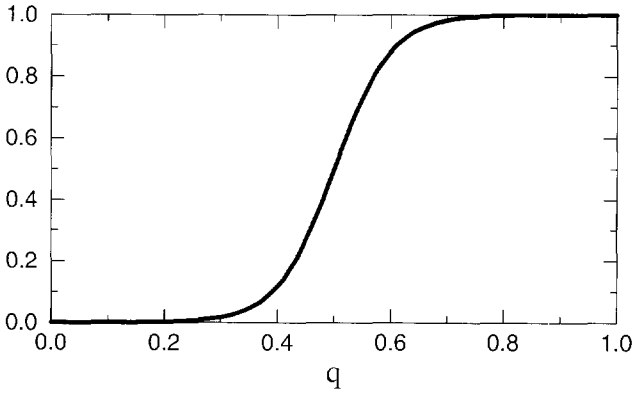


Figure 7.1: The function  $\bar{q}(q)$  with  $c_q = 20.0$ .

The function  $\bar{q}(q)$  with  $c_q = 20.0$  is illustrated in Fig.7.1.

In Eq.(7.10),  $\bar{\mathbf{p}}_i = \{\bar{p}_1, \bar{p}_2, \dots, \bar{p}_N\}$  contains the initial probabilities for matches, which in our application is the result of the local matching.

### 7.3 Formulation of global stereo matching as relaxation labeling

We propose here two formulations for global stereo matching. For simplicity, in the following we denote the regions in the left image as left regions and regions in the right image as right regions.

#### Formulation A

In this formulation, each region in the left image is denoted as an object, and regions in the right image are regarded as labels. We use the following notation, i.e., regions in the left image are indexed by  $i (i = 1, \dots, n)$ , and regions in the right image are expressed by labels  $\lambda$  which take integer value from 1 to  $m$ . We still use  $p_i(\lambda)$  to denote the probability with which label  $\lambda$  is assigned to node  $i$  (the probability of left region  $i$  matched with right region  $\lambda$ ) and require that

$$0 \leq p_i(\lambda), \quad \forall i, \lambda \quad (7.15)$$

and

$$\sum_{\lambda=1}^m p_i(\lambda) = 1, \quad \forall i = 1, \dots, n. \quad (7.16)$$

It can easily be seen that Eq.(7.16) expresses the fact that for each left region the sum of the probabilities with which the left region has been assigned to right regions is 1, i.e., Eq.(7.16) is the weighted expression of uniqueness for left regions. We may write a similar formula for the uniqueness of right regions, that is

$$\sum_{i=1}^n p_i(\lambda) = 1, \quad \forall \lambda = 1, \dots, m. \quad (7.17)$$

The above constraint is not considered in the theoretical study of the previous section. If we want to use the solution developed in the previous section under this formulation, only equation (7.16) is guaranteed, whereas (7.17) cannot be considered. One possible way to avoid this obstacle is to perform the relaxation processes twice: in the first relaxation, left regions are considered as objects and right region as labels; in the second relaxation, right regions are regarded as objects and left region as labels. We take the overlapping of the results from these two processes as the final matching.

## Formulation B

We do not separate left and right regions as objects and labels in this formulation. Instead, each candidate match is considered as an object. The labels have only two values 1 and 0, representing a correct match and no match, respectively. The consistent labeling problem in this case is to assign each candidate match  $i$  a label  $\lambda$  which indicates whether match  $i$  is a good match or not.

In order to connect matches of left regions with right regions, we use the match matrix:

$$\begin{bmatrix} (1, 1) & (1, 2) & \dots & (1, m) \\ (2, 1) & (2, 2) & \dots & (2, m) \\ \dots & & & \\ (n, 1) & (n, 2) & \dots & (n, m), \end{bmatrix}$$

so match  $i$  can be written as  $\{(u_i, v_i), u_i = 1, \dots, n; v_i = 1, \dots, m\}$ .

Under this formulation, we are quite comfortable with the condition (7.1). But we have introduced some difficulties in expressing the conditional probability  $p_{ij}(\lambda, \lambda')$  which must take into account several constraints including uniqueness, which is not needed to be put in the conditional probability under "formulation A". A solution is described below.

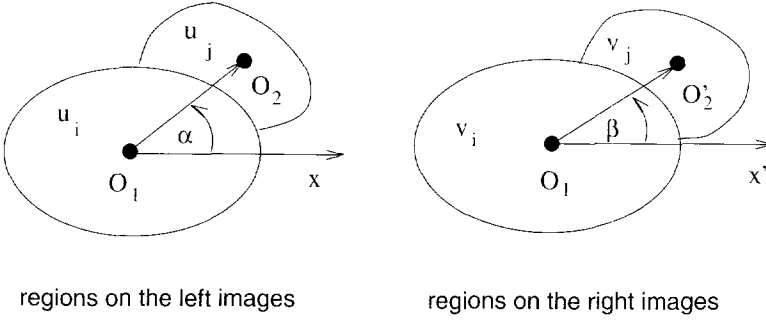


Figure 7.2: Neighboring matches.

We denote two matches  $i = (u_i, v_i)$  and  $j = (u_j, v_j)$ , then define

$$p_{ij}(1/1) = \begin{cases} 0 & \text{if match } j \text{ does violate the uniqueness constrained} \\ & \text{by match } i, \text{ i.e. } u_i = u_j, \text{ or } v_i = v_j. \\ 0 & \text{if } i \text{ and } j \text{ are not neighbors} \\ CO(\alpha, \beta) & \text{otherwise} \end{cases} \quad (7.18)$$

$$p_{ij}(0/1) = 1 - p_{ij}(1/1) \quad (7.19)$$

$$p_{ij}(1/0) = p_i(1) \quad (7.20)$$

$$p_{ij}(0/0) = 1 - p_{ij}(1/0) \quad (7.21)$$

In the above definition, it is assumed that two matches affect each other only if both matches are good matches (both are assigned to "1"). In Eq.(7.18), we give different priority to different constraints: uniqueness takes top priority, i.e., if two matches violate uniqueness, the conditional probability takes zero, and connectivity and ordering do not need to be considered. The second priority is given to neighboring (neighboring match means that their corresponding regions on the left and right images are neighboring). If two matches are neighbors and do not violate the uniqueness, connectivity and ordering are considered further. Connectivity and ordering are of equal priority because a match can only fill in one of these two conditions at one time.

$CO(\alpha, \beta)$  is called the *Connectivity-Ordering function* as defined in the following equation,

$$CO(\alpha, \beta) = \begin{cases} 1 & \text{if } |\beta - \alpha| \leq 90^\circ \\ |\cos(0.5(\alpha - \beta))| & \text{if } |\beta - \alpha| > 90^\circ \end{cases} \quad (7.22)$$

The angles  $\alpha$  and  $\beta$  are shown in Figure 7.2, where  $O_1$  is the geometric center of the left region  $u_i$ , and  $O'_1$  is the geometric center of the right region  $v_i$ , similarly with  $O_2, O'_2$ .  $x$  is the epipolar line at  $O_1$ , and  $x'$  is the epipolar line at  $O'_1$ .

This definition is based on the observation on different relations between neighboring matches  $i$  and  $j$  as shown in Fig.7.3:

- In case (a),  $\alpha = 0$  and  $\beta = 0$ , which indicates matches  $i$  and  $j$  fully support each other, so

$$CO(\alpha, \beta) = 1.0.$$

- In case (b), match  $j$  violates the ordering constrained by the match  $i$ , so

$$CO(\alpha, \beta) = 0.$$

- Case (c) and (d) still indicate that the two matches support each other, therefore

$$CO(\alpha, \beta) = 1.0.$$

- Case (e) somehow violates the ordering condition. It is not exactly according with case (b). We use the cos function to monitor the difference, that is,

$$CO(\alpha, \beta) = |\cos(0.5(\alpha - \beta))|.$$

Notice that  $p_{ij}(\lambda, \lambda')$  is symmetric, i.e.,  $p_{ij}(\lambda, \lambda') = p_{ji}(\lambda', \lambda)$ .

In the next development, we use “*formulation B*” because every constraint is satisfactorily fulfilled.

## 7.4 Objective function of global matching

Since in “*formulation B*” the global matching has only two labels (we use “1” to indicate the “good” matches, and “0” the “bad” ones), the objective function for minimally consistent labeling can be greatly simplified.

Eq.(7.5) becomes:

$$e_{ij}(1) = p_{ij}(1/1)p_j(1) + p_{ij}(1/0)p_j(0)$$

$$e_{ij}(0) = p_{ij}(0/1)p_j(1) + p_{ij}(0/0)p_j(0)$$

It is easily seen that

$$e_{ij}(1) + e_{ij}(0) = 1 \quad (7.23)$$

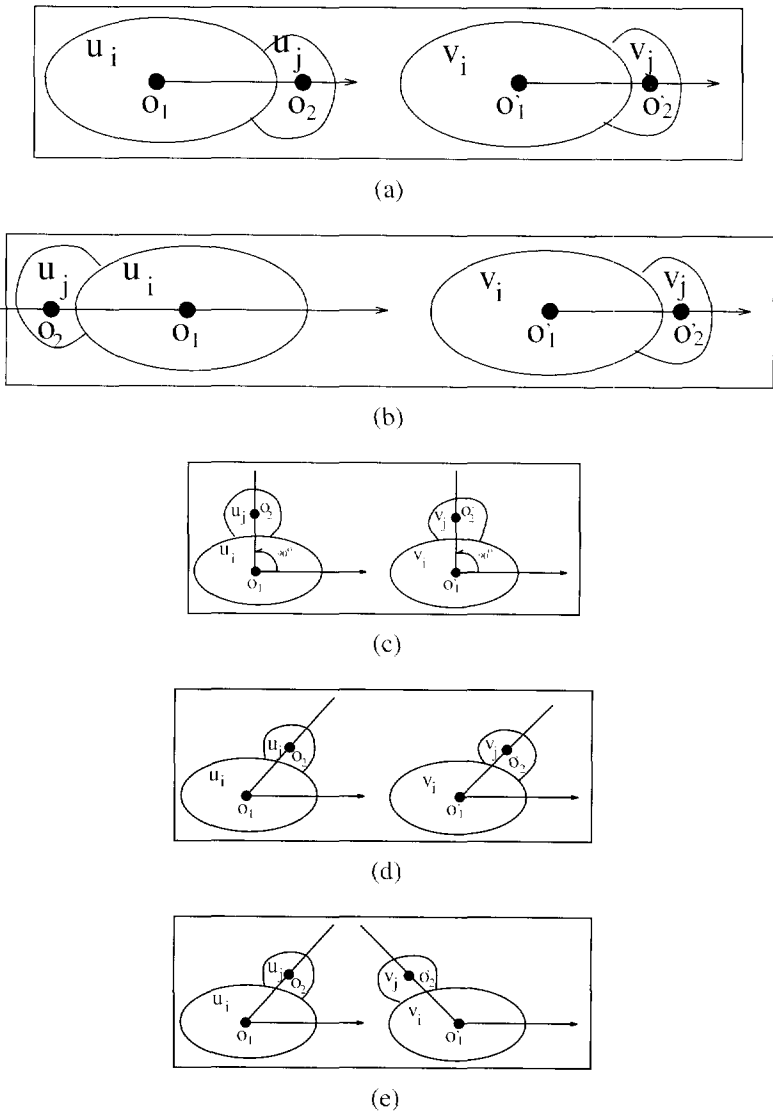


Figure 7.3: Different relations between matches  $i$  and  $j$ .

If we assume that in Eq.(7.6)  $w_{ijk}$  and the neighbor set  $V_i(\lambda_k)$  are independent of the label, i.e.  $w_{ijk} = w_{ij}$  and  $V_i(\lambda_k) = V_i$ , Eq.(7.6) becomes:

$$Q_i(\lambda_k) = \frac{1}{\sum_{j \in V_i} w_{ij}} \sum_{j \in V_i} w_{ij} e_{ij}(\lambda_k)$$

with the condition (7.23), we get:

$$Q_i(1) + Q_i(0) = 1.$$

Then we can further derive that

$$q_i(\lambda_k) = Q_i(\lambda_k)$$

and

$$q_i(1) + q_i(0) = 1.$$

Combining the above result and Eq.(7.14), we further get:

$$\bar{q}(q_i(1)) + \bar{q}(q_i(0)) = 1.$$

Because of this result, we have

$$p_i(0) - \bar{q}(q_i(0)) = \bar{q}(q_i(1)) - p_i(1),$$

and similarly

$$p_i(0) - \bar{p}_i(0) = \bar{p}_i(1) - p_i(1).$$

The objective function  $\Phi$  is now

$$\Phi = \frac{1}{N} \sum_{i=1}^N \bar{w}_i \gamma_i (p_i(1) - \bar{q}(q_i(1)))^2 + \frac{1}{N} \sum_{i=1}^N \bar{w}_i (1 - \gamma_i) (p_i(1) - \bar{p}_i(1))^2.$$

Since we are only interested in “good” matches, we can simplify our notations by letting  $p_i = p_i(1)$ ,  $p_{ij} = p_{ij}(1/1)$ ,  $q_i = q_i(1)$ ,  $\bar{q}_i = \bar{q}(q_i)$ . Since the event that match  $i$  is a good match is independent of the event that match  $j$  is a bad match,  $p_{ij}(1/0) = p_i(1) = p_i$ . Therefore,

$$\begin{aligned} e_{ij}(1) &= p_{ij} p_j + p_i (1 - p_j) \\ q_i &= \frac{1}{\sum_{j \in V_i} w_{ij}} \sum_{j \in V_i} w_{ij} p_{ij} p_j \\ &\quad + p_i - \frac{1}{\sum_{j \in V_i} w_{ij}} \sum_{j \in V_i} w_{ij} p_i p_j. \end{aligned}$$

The objective function becomes

$$\Phi = \frac{1}{N} \sum_{i=1}^N \bar{w}_i \gamma_i (p_i - \bar{q}_i)^2 + \frac{1}{N} \sum_{i=1}^N \bar{w}_i (1 - \gamma_i) (p_i - \bar{p}_i)^2. \quad (7.24)$$

It should be pointed out that the isolated matches are not included in the global objective function.  $N$ , therefore, is actually the number of the un-isolated matches.

## 7.5 Gradient method for minimization

One of the widely used techniques for minimizing a differentiable nonlinear function  $\Phi(\mathbf{p})$  is the steepest descent technique. The updating rule is like the following

$$\mathbf{p}^{n+1} = \mathbf{p}^n + \rho_n \mathbf{u}_n \quad n \geq 0 \quad (7.25)$$

where  $\mathbf{u}_n$  is the negative gradient  $d\Phi/d\mathbf{p}$  at point  $\mathbf{p}^n$  and  $\rho_n$  is a positive number.

The gradient of the objective function for global matching is in the following form:

$$\begin{aligned} \frac{\partial \Phi}{\partial p_j} = & \frac{1}{N} \left\{ 2\bar{w}_i \gamma_i (p_j - \bar{q}_j) \left( 1 - \frac{\partial \bar{q}_j}{\partial p_j} \right) \right. \\ & \left. - \sum_{i \in V_j} 2\bar{w}_i \gamma_i (p_i - \bar{q}_i) \frac{\partial \bar{q}_i}{\partial p_j} + 2\bar{w}_j (1 - \gamma_j) (p_j - \bar{p}_j) \right\} \end{aligned} \quad (7.26)$$

where

$$\frac{\partial \bar{q}_i}{\partial p_j} = \begin{cases} \frac{\partial \bar{q}_i}{\partial p_i} \left( 1 - \frac{\sum_{k \in V_i} w_{ik} p_k}{\sum_{k \in V_i} w_{ik}} \right) & \text{if } i = j \\ \frac{\partial \bar{q}_i}{\partial p_i} \left( \frac{w_{ij} (p_{ij} - p_i)}{\sum_{k \in V_i} w_{ik}} \right) & \text{if } i, j \text{ have relation} \\ 0 & \text{otherwise} \end{cases} \quad (7.27)$$

and

$$\bar{q}'_i = \frac{\partial \bar{q}_i}{\partial q_i} = c_q \frac{e^{-c_q(q_i - 0.5)}}{(1 + e^{-c_q(q_i - 0.5)})^2}. \quad (7.28)$$

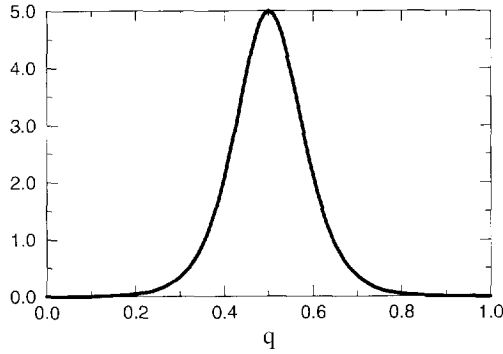


Figure 7.4: The function  $\bar{q}'_i$  with  $c_q = 20.0$ .

Note that two matches  $i$  and  $j$  have a relationship means that 1) they belong to multiple matches; 2) they are neighboring matches.

The function  $\bar{q}'_i$  with  $c_q = 20.0$  is illustrated in Fig.7.4.

In the global matching, we need to specify the  $\bar{w}_i, w_{ij}, \gamma_i$ . The function  $\bar{w}_i$  is defined to be the function of the areas of stereo regions of a match, let

$$AR^i = \frac{|AR^i_l + AR^i_r|}{2}, \quad (7.29)$$

then

$$\bar{w}_i = \frac{1}{1 + e^{-c_w(AR^i - TH_{at})}} \quad (7.30)$$

where  $AR^i_l, AR^i_r$  represent the area of the left region and right region for match  $i$ , respectively.

For  $w_{ij}$ , we have a similar equation as follows

$$AR^{ij} = \frac{|AR^i_l + AR^i_r + AR^j_l + AR^j_r|}{4}, \quad (7.31)$$

then

$$\bar{w}_{ij} = \frac{1}{1 + e^{-c_w(AR^{ij} - TH_{at})}}. \quad (7.32)$$

The balance function  $\gamma_i$  is defined by

$$\gamma_i = \begin{cases} 1.0 & \text{if match } i \text{ has uniqueness or ordering constraints imposed} \\ \gamma^0 & \text{if match } i \text{ does not have uniqueness or ordering constraints} \\ & \text{imposed, but has neighboring matches} \\ 0 & \text{if match } i \text{ is isolated} \end{cases} \quad (7.33)$$

In Eq.(7.25),  $\rho_n$  is a factor used to provide a trade-off between speed and the stability of the iterative process. Since our updating process is subject to a bound constraint, i.e.

$$0 \leq \mathbf{p}^{n+1} \leq 1,$$

we require

$$\rho_n \leq \rho_n^{\max}. \quad (7.34)$$

The algorithm for determining  $\rho_n^{\max}$  is described in the following:

---

**Algorithm 7.1 :**

1. Accept as the current feasible point

$$\mathbf{p} = \{p_1, \dots, p_N\}$$

and the updating direction

$$\mathbf{u} = \{u_1, \dots, u_N\}$$

2. LOOP ( $k = 1, \dots, N$ )

if ( $u_k \geq 0$ )  
      $d = 1 - p_k$   
 else  
      $d = p_k$   
      $s_k = d|u_k|$

END LOOP

3. choose the smallest  $s_k$  as output  $\rho_n^{\max}$
- 

## 7.6 Select unique matches

After a number of iterations during the global matching, we achieve updated matching measures. In principle, the updated measures have resolved the uniqueness constraints. But sometimes the numerical optimization procedure cannot guarantee that the local

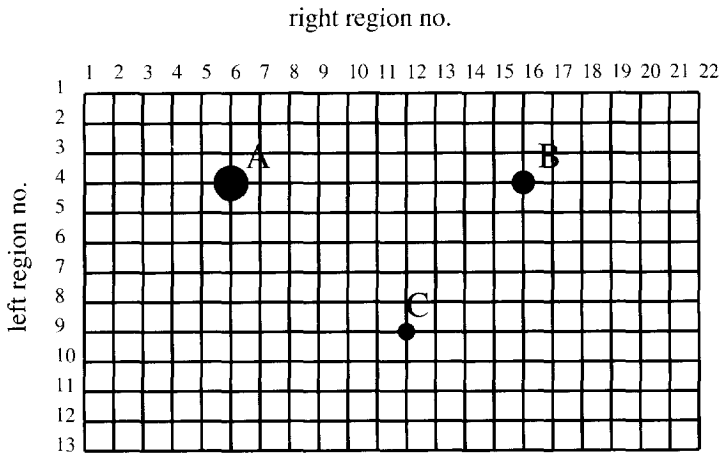


Figure 7.5: Select unique matches.

minimum is reached. We therefore need an algorithm to select unique matches. The algorithm we use is described below:

---

**Algorithm 7.2 :**

- sort all the matching measures;
  - select a match from the matching measure list in descendant order, check whether it is in conflict with other matches (i.e. uniqueness). In Fig.7.5, the grid displays the matching matrix, with the horizontal lines representing the left region number and the vertical lines representing the right regions number. The grid nodes are the positions for the possible matches. The uniqueness constraint can be interpreted that on each horizontal line, there should not be more than one match, similarly with the vertical lines. Suppose in this figure, point "A" is the match which has the biggest matching measure, "B" is the second, and "C" the third. We first select "A" as a good match, and mark the corresponding horizontal and vertical line as "occupied" (in this figure, "occupied" lines are thicker). When we go on checking other matches, they are selected as good matches only when the corresponding horizontal and vertical lines are not "occupied". In this example, match "B" will not be selected, and match "C" will be selected.
-

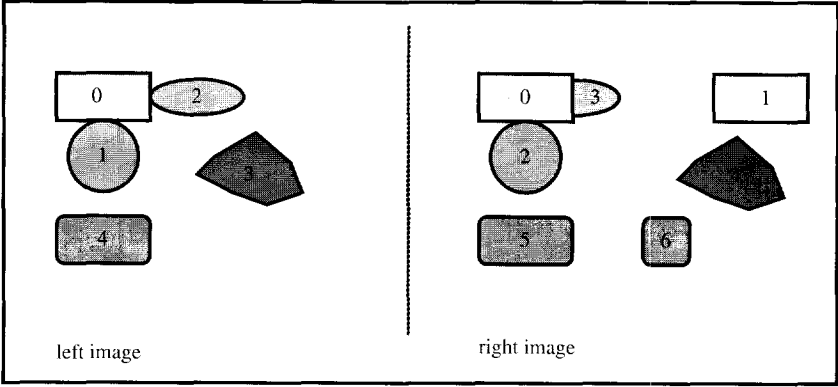


Figure 7.6: A simulated stereo case.

match no.	left region no.	right region no.	initial match measure
0	0	0	1.0
1	0	1	1.0
2	1	2	1.0
3	2	3	0.3
4	3	4	1.0
5	4	5	1.0
6	4	6	0.5

Table 7.1: Matching relations of the simulated stereo case.

## 7.7 Results

In this section, we first show the experimental result on a simulated case. Several results from practical examples are presented later.

The simulated case is illustrated in Fig.7.6. The match numbers, their corresponding left region and right region, and initial match measures are listed in Table 7.1. The description of the matches is given below:

- match “0” and “1” are multiple matches, only one of them should be selected in the final matching. Since match “0” has the support from neighboring matches “2” and “3”, it should stay in the final matching.
- match “2” is already a good match, it should remain during the global matching.

match no.	initial measure	final measure ( $\gamma^0 = 0.5$ )	final measure ( $\gamma^0 = 1.0$ )
0	1.0	1.00	1.00
1	1.0	0.00	0.00
2	1.0	1.00	1.00
3	0.3	0.65	1.00
4	1.0	1.00	1.00
5	1.0	1.00	1.00
6	0.5	0.00	0.00

Table 7.2: Results of the simulated stereo case.

- the match measure for match “3” should gain in the global matching, because it can get full support from match “0”.
- match “4” is an isolated match, its match measure should not change in the procedure of global matching.
- match “5” and “6” are multiple matches, either of them can have connections with their neighbors. It is reasonable to select match “5” in the global matching, and decrease the match measure of match “6”.

The results of global matching for this case are shown in Table 7.2. The results meet our expectation as described above, except a difference on match “3” caused by the different  $\gamma^0$  value. It is easily seen that  $\gamma^0$  provides a trade-off between the initial measure and the gain from the neighboring support. We would like to choose a value near “0.5” so that neither information source will be ignored.

In Fig.7.7, the updating procedure of the global matching is illustrated by the change of the  $\Phi$  value during the iterations. With  $\gamma^0 = 1.0$ , the  $\Phi$  value can finally reach 0. When we assign  $\gamma^0 = 0.5$ , there is a small residual in the final result, which is the effect due to the compromise between the fulfillment of all kinds of matching constraints, and the initial measures from local matching.

The experiment based on the Pentagon images (the stereo image is shown in Figure 7.8) is given next. The segmentation results in 216 and 327 regions for left and right images, respectively. A Kuwahara filter with size 5x5 is used to suppress the noise in the original images before segmentation. After region matching and re-merging using matching information, we get 47 candidate matches (most of matched regions are on the building area, unmatched regions mostly correspond to the terrain). Fig.7.9 shows the

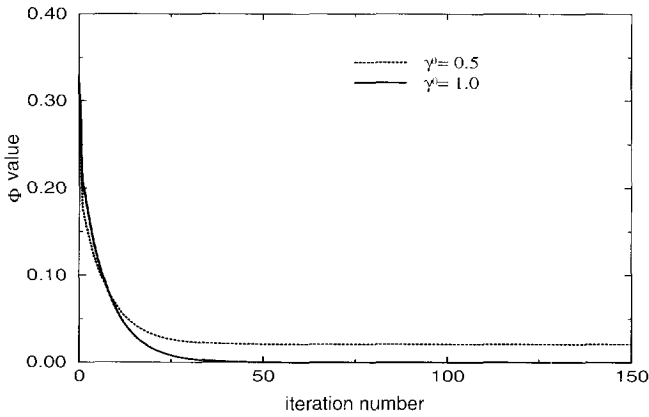


Figure 7.7: The iteration process for simulated stereo case.

iteration process, and Fig.7.10 is the constructed disparity map in a projective view. We also give the results on three other stereo images in Figs.7.11, 7.12, 7.13, 7.14, 7.15, 7.16, 7.17, 7.18, and 7.19.

## 7.8 Conclusions

Relaxation labeling is a very useful tool in computer vision. In this chapter, we extend the conventional relaxation labeling to solve our global matching problem. Within our formulation, all the constraints contained in the global matching are well fulfilled. It can be concluded that we have solved the global matching problem and the results from the experiments are acceptable. Our contributions lie in: 1) introducing a soft thresholding function when combining the evidence; 2) introducing a balance factor to control the updating process.

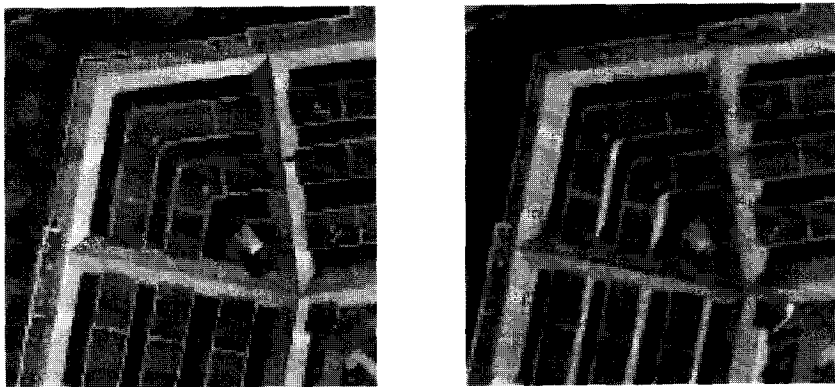


Figure 7.8: A part of the Pentagon images (I).

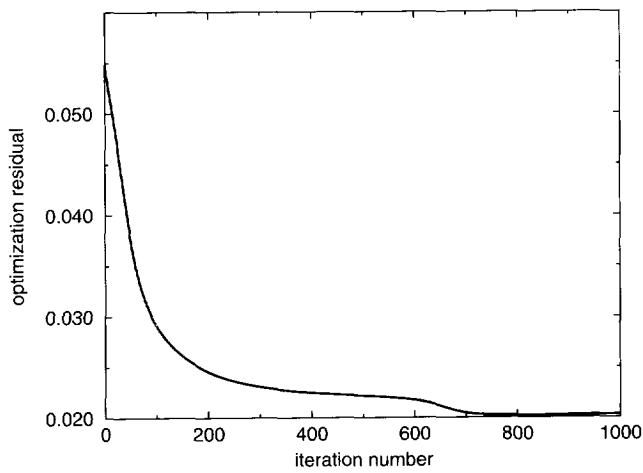


Figure 7.9: Iteration process for Pentagon images (I).

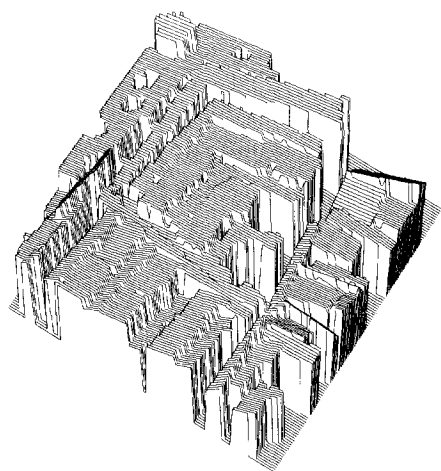


Figure 7.10: Reconstructed disparity map for the Pentagon images (I): a perspective view.

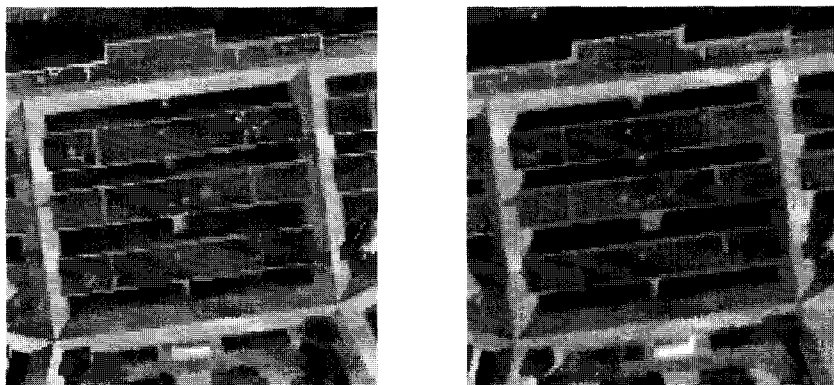


Figure 7.11: A part of Pentagon image (II).

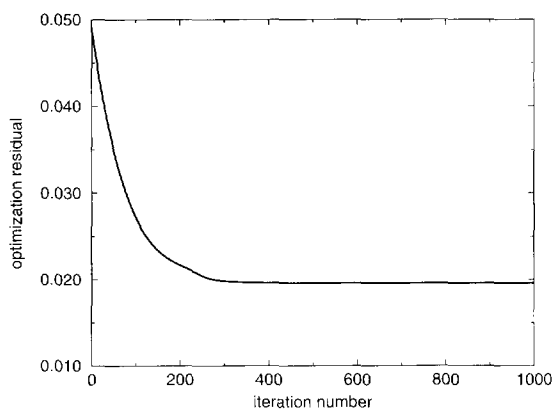


Figure 7.12: Iteration process for the Pentagon images (II).

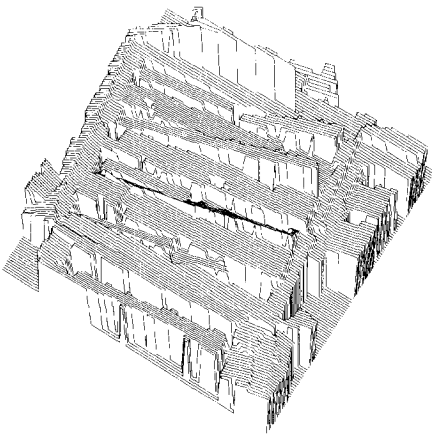


Figure 7.13: Reconstructed disparity map for the Pentagon images (II): a perspective view.

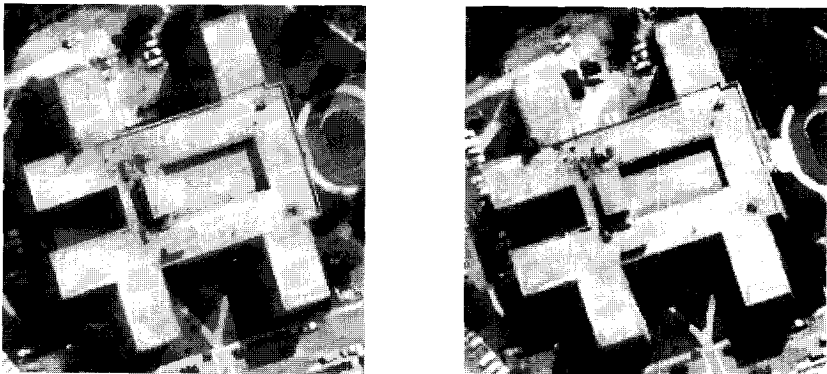


Figure 7.14: Stereo building images (I).

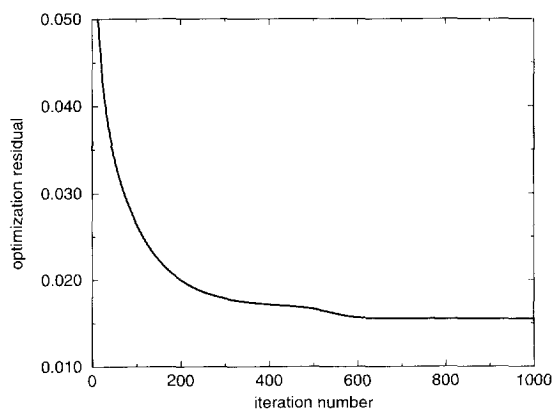


Figure 7.15: Iteration process for building images (I).

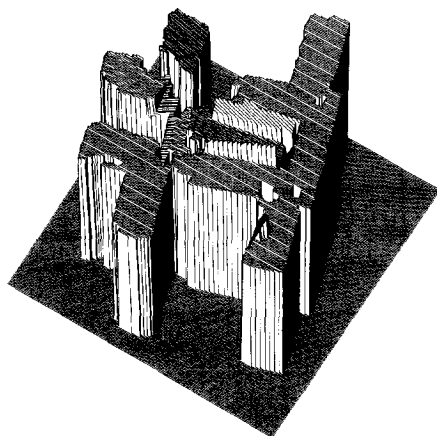


Figure 7.16: Reconstructed disparity map for building images (I): a perspective view.

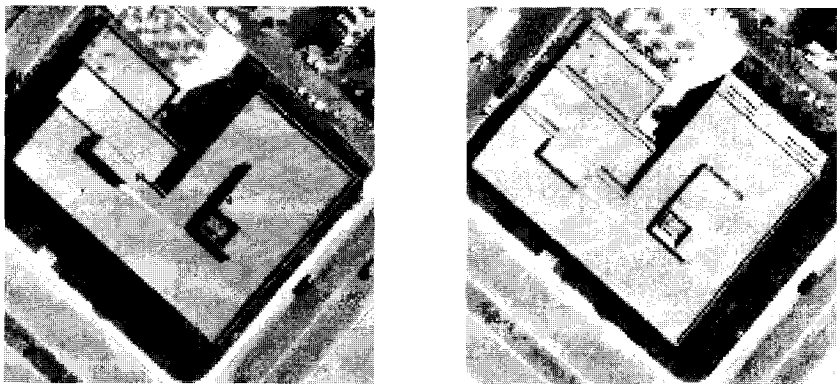


Figure 7.17: Stereo building images (II).

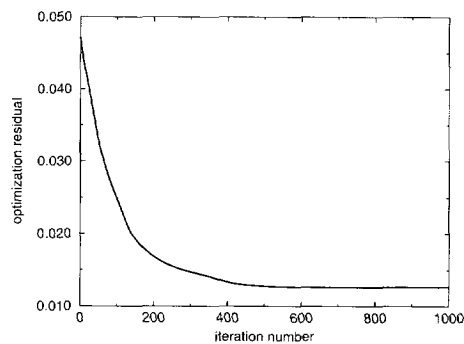


Figure 7.18: Iteration process for building images (II).

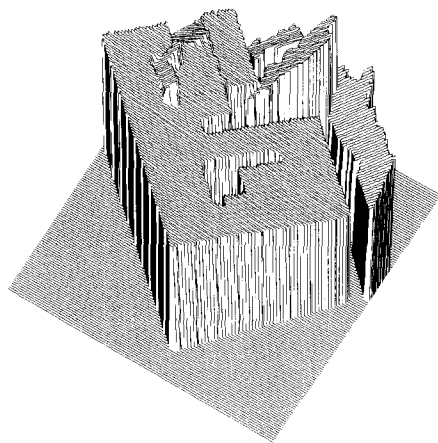


Figure 7.19: Reconstructed disparity map for building images (II).

## Chapter 8

# General Conclusions and Discussions

### 8.1 The problem

Stereo vision is one of the central issues in computer vision and photogrammetry. It is an important method for machine perception because it leads to relatively direct measurements and, unlike monocular techniques, does not infer depth from weak and unverifiable photometric and statistical assumptions, nor does it require specific detailed object models. The key problem in stereo vision is the matching – establishing the relationships between the features over stereo images. There have been tremendous efforts in investigating the methods using signal (image intensity), edges and lines as matching features. In this thesis, our effort is devoted to region-based method, because we consider that the image regions are stronger features for the matching in the sense that they contain both the geometric properties (the boundaries of regions) and radiometric properties (intensities, textures, etc.). The method, on the other hand, requires good segmentation, which is also a difficult task. In this thesis, we have spent quite a lot of effort in tackling the segmentation problem.

### 8.2 Our solutions

#### 8.2.1 Segmentation

In **Chapter 2**, the Minimal Description Length (MDL) principle is applied to the region growing problem. There are some references available for the application of MDL in computer vision, but it is our contribution to use it in the region growing algorithm. In

a region growing algorithm, there are usually three main issues to be considered, i.e. 1) the modeling; 2) the decision making mechanism for merging or splitting; 3) the implementation. When we consider whether two small regions should be merged or not, there are several issues: 1) what is the best model to be used for the studied data (assuming we have a number of preselected models); 2) the determination of the good points, noise, and outliers; 3) if we want to consider several different sources, how can we combine them to make a unique decision. The MDL principle has the power to permit estimates of the entire model, its parameters, their number, and even the way the parameters appear in the model, i.e., the model structure. In **Chapter 2**, the MDL principle has been used to describe the models, possible noise and outliers of image intensities. The decision making for region growing is rather simple when using the MDL principle, i.e., what we need to do is to see whether the encoded result is decreased or not. In the experiments, we have compared our results with the split-and-merge method using the variance criterion. The results show that the MDL-based method can achieve the similar, sometimes even better results than the split-and-merge method.

In **Chapter 3**, we pursued the problem further, by including shape constraints in the segmentation. In many applications, the scene consists of objects whose shapes are usually quite regular. It is reasonable to expect that usage of a shape constraint can improve the segmentation result when applicable. The shape model used in this chapter is a closed polygon depicted by a number of smooth segments. In order to get shape constraints, we introduced a method for optimal curve fitting which combines the techniques of detecting extreme points and a splitting process. The shape constraints are also encoded using the MDL principle. The experiments show that using shape constraints can improve the segmentation result when the noise is relatively severe and the images contain objects with regular shapes.

### 8.2.2 Local matching

Our matching paradigm consists of local matching and global matching. During the local matching, each region in one image is compared *locally* with the regions in the other image, which means that only local attributes are used in the matching. The topological information and other matching constraints are considered in the *global matching*.

Attributes related to regions consists of: 1) geometric attributes, such as region area, shape of region boundary, etc; 2) physical and optical attributes, such as color or grey value information. In this thesis, a thresholding method is used to compare the intensity difference between stereo regions. **Chapter 4** is devoted to curve matching (matching boundaries of stereo regions). There have been many efforts in solving the matching problem of stereo curves. We found that the existing methods are not satisfactory because

of: 1) some methods can only deal with two dimensional rotation, translation, and scaling, but cannot deal with perspective transformation; 2) some methods are sensitive to noise and cannot tackle errors due to bad segmentation and occlusion; 3) some methods have requirements on the shapes of the curves concerned (e.g. mathematically describable, smooth curvatures, etc.). In our region based matching, we require that the curve matching algorithm to meet the requirements: 1) the method should be able to get a correct matching measure even when occlusions occur and there are differences between stereo curves due to the inconsistency of segmentation results over stereo images; 2) no assumptions are made on the mathematical property of the shape of region boundaries. The assumptions we have made are that the curves are planar, closed, and simple. The latter assumptions (closed and simple) are easy to meet if the curves concerned stem from the region based segmentation. The first assumption is met in our current experiments where the images contain mostly block-like buildings. We, of course, would like to extend our idea to deal with other curves than planar curves, which remains one of our future research topics.

The merit of the overall matching between the stereo curves can be judged by the difference between the two normalized curves. By normalization in the stereo case, we mean that one curve is transferred from one image to the other image. The first step in our algorithm is to establish the point correspondences. Based on the disparity information, we can estimate the parameters for curve normalization for each three points along the stereo curves. The most likely values for the parameters are calculated by a histogram technique. After curve normalization, the differences between the two curves are measured using a distance transformation. From many tests, it follows that the method is quite promising and is robust against imperfect segmentation and occlusion.

In our matching algorithm, it is required to consider several features associated with each region, i.e., grey intensity (can be characterized by mean value, for example), area, shape, etc. We need a mechanism to combine them in order to make a unified decision. Several methods for combining evidences produced by multiple information sources have been studied by different researchers; these include Bayesian reasoning, Mycin style measures of belief and disbelief, the Dempster-Shafer structure, fuzzy logic, etc. In our research we use the ideas of fuzzy measure and fuzzy integral introduced by Sugeno for the representation and development of multiple criteria aggregation functions. The reasons are: 1) there is a solid theoretic background for fuzzy measure and fuzzy integral, which provides a general framework for representing and aggregating multiple criteria (it is worth mentioning that the probability measure is a special case of the fuzzy measure); 2) no assumptions are required for independent events. In **Chapter 5**, we applied the fuzzy measure and fuzzy integral to the fusion of stereo cues. We designed a learning method to calculate the fuzzy measures for each stereo cue, from which we found that shape of

image regions play the most important role in the matching.

In **Chapter 6**, the method for integration of segmentation and stereo matching is proposed. The performance of region-based method is strongly dependent on the results of segmentation algorithms. Because the stereo images are the results of capturing 3-D scenes from different positions and angles, the segmentation results over the stereo images are usually inconsistent. It is very difficult and may be impossible to improve the segmentation methods for perfect segmentation results (here "perfect" means consistent segmentation results over stereo images when stereo matching is concerned). In order to get good matching results, it is necessary to re-segment these regions after matching, that is, after an initial region-based segmentation, a candidate stereo matching is carried out, which assigns the corresponding regions from one image to the other image by shape-based matching. During the next step of segmentation, stereo information is included, that is, in considering the merging of one region with its neighboring region, the corresponding regions in the candidate pools are extracted and the matching is carried out using intensity and shape information from both images. The experiments have shown that the segmentation results are improved by such integration and consequently the matching results are improved. It should be pointed out that the mechanism we have used in this thesis is not very complex. In order to handle complex situations, rule-based systems can be used. Nevertheless, the result shows the success of the method and demonstrates that integrating different vision modules in a unified framework is a good direction in solving difficult vision problems and has potentially many practical applications.

### 8.2.3 Global matching

Relaxation labeling is a very useful tool in computer vision. In **Chapter 7**, the conventional relaxation labeling is extended to solve our global matching problem. Within the formulation, all the constraints contained in the global matching are well fulfilled. It can be concluded that we have solved the global matching and the results from the experiments are satisfactory. The contributions lie in: 1) introducing a soft thresholding function when combining the evidences; 2) introducing a balance factor to control the updating process.

## 8.3 Future work

Because the restriction on the period of the research, there are still a lot of interesting issues which can be and sometimes need to be improved. Note the following:

- Use graph theory to speed up the region growing algorithm. There are already some articles available for using graph theory in segmentation, such as [179], but the combination of using MDL and graph theory still needs to be investigated further.

- Study the possibility of using topological information in segmentation.
- Extend the curve matching algorithm to 3-D curves on other kinds of surfaces.
- Currently the rule and mechanism for integrating segmentation are rather simple. More detailed investigation is therefore necessary to the more complex situations for various applications.
- Integration of different matching methods like signal based, edge based, region based, and others. We have noticed in our experiments that even on the building areas of test images, it is possible that the internal area of an object is hardly homogeneous (see Figs.7.14 and 7.16), thus it creates difficulties for region-based segmentation. One of the possible solutions to this problem can be the signal-based matching method.



# Bibliography

- [1] J. Aloimonos and D. Shulman, *Integration of Visual Modules — an Extension of the Marr Paradigm*, Academic Press, Boston, 1989.
- [2] D. Marr, *Vision*, Freeman company, San Francisco, 1979.
- [3] S.T. Barnard and M.A. Fischer, "Computational and biological models of stereo vision", in *Proc. of Image Understanding Workshop*, 1990.
- [4] H.P. Moravec, "Towards automatic visual obstacle avoidance", in *Proc. of 5th Joint Conference of Artificial Intelligence*, p. 584, Cambridge, 1977.
- [5] L.B. Wolff, "A general formalization of stereo vision", in *Proc. of SPIE's Intelligent Robots and Computer Vision: Sixth in a Series*, vol. 848, 1987.
- [6] M.J.P.M. Lemmens, "A survey on stereo matching techniques", in *Proc. of 16th Inter. Congress of International Society for Photogrammetry and Remote Sensing*, Japan, 1988.
- [7] B.P. Wrobel, "Facets stereo vision (FAST vision) - a new approach to computer stereo vision and to digital photogrammetry", in *Proc. of ISPRS Intercommission Conference on Fast Processing of Photogrammetric Data*, Interlaken, Switzerland, 1987.
- [8] B.P. Wrobel and M. Weisensee, "Implementation aspect of facets stereo vision and some application", in *Proc. of ISPRS Intercommission Conference on Fast Processing of Photogrammetric Data*, Interlaken, Switzerland, 1987.
- [9] U.V. Helava, "Object-space least-square correlation", *Photogrammetric Engineering and Remote Sensing*, vol. 54, pp. 711–714, June 1988.
- [10] M.J. Hannah, "A system for digital stereo image matching", *Photogrammetric Engineering and Remote Sensing*, vol. 55, pp. 1765–1770, Dec. 1989.
- [11] S.T. Barnard, "Recent progress in CYCLOPS: A system for stereo cartography", in *Proc. of Image Understanding Workshop*, pp. 449–455. SRI, 1990.

- [12] F. Ackermann, "Digital image correlation: performance and potential application in photogrammetry", *Photogrammetric Record*, vol. 11, pp. 429–439, October 1984.
- [13] A. Pertl, "Digital image correlation with the analytical plotter planicomp C100", in *Inter. Arch. of Photogrammetry*, vol. III, pp. 874–881, 1984.
- [14] A. Pertl, "Digital image correlation with an analytical plotter", *Photogrammetria*, vol. 40, pp. 9–19, 1985.
- [15] W. Förstner, "Quality assessment of object location and point transfer using digital image correlation techniques", in *Proc. of 15th Congress of International Society for Photogrammetry and Remote Sensing, Commission III*, pp. 169–191, Rio de Janeiro, 1984.
- [16] A.W. Grün and E.P. Baltsavias, "Geometrically constrained multiphoto matching", *Photogrammetric Engineering and Remote Sensing*, vol. 54, pp. 633–641, May 1988.
- [17] D. Rosenholm, "Accuracy improvement of digital matching for evaluation of digital terrain models", in *Int. Arch. of Photogrammetry*, pp. 573–587, 1986.
- [18] G. Medioni and R. Nevatia, "Segment-based stereo", *Computer Vision, Graphics, and Image Processing*, vol. 31, pp. 2–18, 1985.
- [19] W. Förstner, "A feature based correspondence algorithm for image matching", in *Int. Arch. of Photogrammetry*, pp. 1–17, Rovaniemi, 1986.
- [20] L. Dreschler, *Ermittlung markanter punkte auf der bildern bewegter objekte und berechnung einer 3D beschreibung auf dieser grundlage*, PhD thesis, University of Hamburg, 1981.
- [21] L. Dreschler and H.-H. Nagel, "Volumetric model and 3D trajectory of a moving car derived from monocular tv frame sequences of a street scene", *Computer Graphics and Image Processing*, vol. 20, pp. 199–228, 1982.
- [22] Y.-N. Zhang, "Matching in image-object dual spaces", in *Proc. of SPIE's Industrial Vision Metrology*, vol. 1526, 1991.
- [23] Y.-N. Zhang, "A new stereo matching approach in image/object dual spaces", in *Proc. of XVII Congress of International Society for Photogrammetry and Remote Sensing, Commission III*, Washington, D.C., U.S.A., 1992.
- [24] K.C. Lo and N.J. Mulder, "Automatic DEM data generation and change detection by region matching", in *Proceeding of 17th International Congress of Photogrammetry and Remote Sensing*, Washington, D.C., 1992.

- [25] H. Maitre and W. Luo, "Using models to improve stereo reconstruction", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 269–277, Feb. 1992.
- [26] H.-J. Lee and W.-L. Lei, "Region matching and depth finding for 3D objects in stereo aerial photographs", *Pattern Recognition*, vol. 23, pp. 81–94, 1990.
- [27] A. Goshtasby and C.V. Page, "Image matching by a probabilistic labeling process", in *Proc. of International Conference on Pattern Recognition*, pp. 307–309, Montreal, Canada, July 1984.
- [28] L.G. Shapiro and R.M. Haralick, "Structural descriptions and inexact matching", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 3, pp. 504–519, September 1981.
- [29] K.L. Boyer and A.C. Kok, "Structural stereopsis for 3-d vision", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, pp. 144–166, March 1988.
- [30] M. You and A.K.C. Wong, "An algorithm for graph optimal isomorphism", in *Proc. of International Conference on Pattern Recognition*, pp. 316–319, 1984.
- [31] W.H. Tsai and K.S. Fu, "Error-correcting isomorphism of attributed relational graphs for pattern analysis", *IEEE Transactions on System, Man, and Cybernetics*, vol. SMC-9, pp. 757–768, Dec. 1979.
- [32] L. Kichen, "Relaxation applied to matching quantitative relational structures", *IEEE Transactions on System, Man, and Cybernetics*, vol. SMC-10, pp. 96–101, Feb. 1979.
- [33] S. Umeyama, "An eigen decomposition approach to weighted graph matching problems", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, pp. 695–703, Sep. 1988.
- [34] A.K.C. Wong and M. You, "Entropy and distance of random graphs with application to structure pattern recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-7, pp. 599–609, Sep. 1985.
- [35] F. Esposito, D. Malerba, and G. Semeraro, "Classification in noisy environments using a distance measurement between structural symbolic descriptions", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 390–402, March 1992.
- [36] A. Sanfeliu and K.-S. Fu, "A distance measure between attributed relational graphs for pattern recognition", *IEEE Transactions on System, Man, and Cybernetics*, vol. SMC-13, May/June 1983.
- [37] L. Quan and R. Mohr, "Matching perspective images using geometric constraints and perceptual grouping", in *Proc. of IEEE Second International Conference on Computer Vision*, pp. 679–684, Florida, 1988.

- [38] N.M. Nasrabadi and Y. Liu, "Stereo vision correspondence using a multichannel graph matching technique", *Image and Vision Computing*, vol. 7, pp. 237–245, 1989.
- [39] E. Mjolsness, G. Gindi, and P. Anandan, "Optimization in model matching and perceptual organization", *Neural Computation*, vol. 1, pp. 218–229, 1989.
- [40] R. Mohan, G. Medioni, and R. Nevatia, "Stereo error detection, correction, and evaluation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, Feb. 1989.
- [41] W.E.L. Grimson and D.P. Huttenlocher, "On the verification of hypothesized matches in model-based recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, Dec. 1991.
- [42] B.K.P. Horn, "Obtaining shape from shading information", in P.H. Winston, editor, *The Psychology of Computer Vision*. McGraw-Hill, New York, 1975.
- [43] T.M. Strat, "A numerical method for shape from shading from a single image", Master's thesis, A.I. lab., M.I.T., Cambridge, MA, 1979.
- [44] K. Ikeuchi and B.K.P. Horn, "Numerical shape from shading and occluding boundaries", *Artificial Intelligence*, vol. 17, pp. 141–184, 1981.
- [45] B.K.P. Horn, "Calculating the reflectance map", *Appl. Opt.*, vol. 18, pp. 1770–1779, 1979.
- [46] A.P. Pentland, "Local shading analysis", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, pp. 170–187, March 1984.
- [47] M.J. Brooks and B.K.P. Horn, "Shape and source from shading", in *Proc. of International Joint Conference on Artificial Intelligence*, pp. 932–936, Los Angeles, CA, August 1985.
- [48] C.M. Brown, D.H. Ballard, and E. Rainero, "Constraint propagation in shape from shading", in *Proc. of Image Understanding Workshop*, 1983.
- [49] K.A. Stevens, *Surface perception from local analysis of texture and contour*, Technical report TR 512, M.I.T., Cambridge, MA, 1979.
- [50] A. Witkin, *Shape from contour*, PhD thesis, Department of Psychology, M.I.T., 1980.
- [51] L.S. Davis, S.M. Dunn, and L. Janos, "Efficient recovery of shape from texture", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 5, pp. 485–492, 1983.
- [52] K.I. Kanatani, "Detection of surface orientation and motion from texture by a stereological technique", *Artificial Intelligence*, vol. 23, pp. 213–237, 1984.

- [53] R. Bajcsy and L. Lieberman, "Texture gradient as a depth cue", *Computer Vision, Graphics, and Image Processing*, vol. 5, pp. 52–67, 1976.
- [54] T. Kanade and J. Kender, "Skewed symmetry: mapping image regularities into shape", Technical report, Dept. of Computer Science, Carnegie-Mellon University, 1981.
- [55] A. Witkin, "Recovering surface shape and orientation from texture", *Artificial Intelligence*, vol. 17, pp. 17–45, 1981.
- [56] M. Brady and A. Yuille, "An extremum principle for shape from contour", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, 1984.
- [57] M. Sugeno, *Theory of fuzzy integrals and its application*, PhD thesis, Tokyo Institute of Technology, 1974.
- [58] M. Sugeno, "Fuzzy measures and fuzzy integrals: A survey", in M.M. Gupta, G.N. Saridis, and B.R. Gaines, editors, *Fuzzy Automata and Decision Processes*, pp. 89–102. North Holland, Amsterdam, 1977.
- [59] R.R. Yager, "Element selection from a fuzzy subset using the fuzzy integral", *IEEE Transactions on System, Man, and Cybernetics*, vol. 23, pp. 467–477, 1993.
- [60] R.R. Yager, "A general approach to criteria aggregation using fuzzy measures", *International Journal of Man-Machine Studies*, vol. 38, pp. 187–213, 1993.
- [61] R.M. Haralick and L.G. Shapiro, "Image segmentation techniques", *Computer Vision Graphics and Image Processing*, vol. 29, pp. 100–132, 1985.
- [62] J.J. Gerbrands, E. Backer, and X.S. Cheng, "Multiresolution cluster segmentation using spatial context", in *Proc. of International Conference on Pattern Recognition*, pp. 1333–1335, 1986.
- [63] J.J. Gebrands and E. Backer, "Split-and-merge segmentation of SLAR-imagery: segmentation consistency", in *7th International Conference on Pattern Recognition*, pp. 284–286, Montreal, 1984.
- [64] G.B. Benie, K.P.B. Thomson, and M. Goldberg, "A comparison of four segmentation algorithms in the context of agricultural remote sensing", *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 44, pp. 1–13, 1989.
- [65] P.J. Besl and R. Jain, "Segmentation through variable-order surface fitting", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, pp. 167–192, Mar. 1988.
- [66] B. Bhanu, S. Lee, and J. Ming, "Self-optimizing control system for adaptive image segmentation", in *Proceedings of Image Understanding Workshop*, pp. 583–596, Pittsburg, U.S.A., September 1990.

- [67] J.F. Haddon and J.F. Boyce, "Image segmentation by unifying region and boundary information", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, pp. 929–948, 1990.
- [68] G. Hu and G. Stockman, "Representation and segmentation of a cluttered scene using fused edge and surface data", in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 313–318, San Diego, CA, U.S.A., June 1989.
- [69] S.-P. Liou, A.H. Chiu, and R.C. Jain, "A parallel technique for signal-level perceptual", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, April 1991.
- [70] T.N. Pappas and N.S. Jayant, "An adaptive clustering algorithm for image segmentation", in *Proc. of Second International Conference on Computer Vision*, pp. 310–315, Tampa, FL, U.S.A., Dec. 1988.
- [71] A.A. Rodriguez and O.R. Mitchell, "Image segmentation by background extraction refinements", in *Proc. of SPIE Vol. 1192 Intelligent Robots and Computer Vision VIII: algorithms and techniques*, pp. 122–130, 1989.
- [72] W. Snyder, G. Bilbro, A. Logenthiran, and S. Rajala, "Optimal thresholding – a new approach", *Pattern Recognition Letters*, vol. 11, pp. 803–810, 1990.
- [73] T.S. Sumanaweera, G. Healey, B.U. Lee, T.O. Binford, and J. Ponce, "Image segmentation using geometric and physical constraints", in *Proceedings of Image Understanding Workshop*, pp. 1091–1099, Cambridge, MA, U.S.A., April 1988.
- [74] C.J. Tsikos and R.K. Bajcsy, "Segmentation via manipulation", *IEEE Transactions on Robotics and Automation*, vol. 7, June 1991.
- [75] D.Y. Kim, P. Meer, D. Mints, and A. Rosenfeld, "Robust image recovery by a least median squares technique", in *Proc. of Image Understanding and Machine Vision, Optical Society*, June 1989.
- [76] P. Fua and A. J. Hanson, "Objective functions for feature discrimination: Theory", in *Proc. of Image Understanding Workshop*, Palo Alto, May 23–26 1989. Morgan Kaufmann.
- [77] P. Fua and A. J. Hanson, "Objective functions for feature discrimination: Applications to semiautomated and automated feature extraction", in *Proc. of Image Understanding Workshop*, Palo Alto, May 23–26 1989. Morgan Kaufmann.
- [78] P. Fua and A.J. Hanson, "An optimization framework for feature extraction", *Machine Vision and Applications*, vol. 4, pp. 59–87, 1991.
- [79] Y.G. Leclerc, "Image and boundary segmentation via minimal-length encoding on the connection machine", in *Proc. of Image Understanding Workshop*, pp. 1056–1069, Palo Alto, CA, U.S.A., May 1989.

- [80] Y.G. Leclerc, "Region grouping using the minimum-description-length principle", in *Proc. of Image Understanding Workshop*, pp. 473–479, 1990.
- [81] A. Leonardis, A. Gupta, and R. Bajcsy, "Segmentation as the search for the best description of the image in terms of primitives", in *Proc. of Third International Conference on Computer Vision*, pp. 121–125, Osaka, Japan, Dec. 1990.
- [82] R. Meer, D. Mintz, and A. Rosenfeld, "Least median of squares based robust analysis of image structure", in *Proc. of Image Understanding Workshop*, pp. 231–254, 1990.
- [83] J. Rissanen, "Modeling by shortest data description", *Automatica*, vol. 14, pp. 465–471, 1978.
- [84] J. Rissanen, "A universal priori for integers and estimation by minimum description length", *The Annals of Statistics*, vol. 11, pp. 416–431, 1983.
- [85] J. Rissanen, "Universal coding information prediction and estimation", *IEEE Transactions on Information Theory*, vol. IT-30, July 1984.
- [86] T. Darell, S. Sclaroff, and A. Pentland, "Segmentation by minimal description", in *Proc. of International Conference on Computer Vision*, pp. 112–116, 1990.
- [87] K. Keeler, "Map representations and coding-based priors for segmentation", in *Proc. of Inter. Conference on Computer Vision and Pattern Recognition*, pp. 420–425, Hawaii, 1991.
- [88] W. Förstner, "Image analysis techniques for digital photogrammetry", Technical report, Institute for photogrammetry, Stuttgart University, 1989.
- [89] H. Bunke and O. Bunke, *Nonlinear regression, functional relations and robust methods*, John Wiley & Sons, Chichester, New York, Brisbane, Toronto, Singapore, 1989.
- [90] G.E.P. Box and S.L. Anderson, "Permutation theory in the derivation of robust criteria and the study of departures from assumption", *J. Royal Statist. Soc.*, vol. 17, pp. 1–34, 1955.
- [91] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery, *Numerical recipes in C*, Cambridge University Press, 1992.
- [92] S.-Y. Chen, W.-C. Lin, and C.-T. Chen, "Split-and-merge image segmentation based on localized feature analysis and statistical tests", *CVGIP: Graphical Models and Image Processing*, vol. 53, pp. 457–475, 1991.
- [93] K.C. Strasters and J.J. Gerbrands, "Three-dimensional image segmentation using a split, merge and group approach", *Pattern Recognition Letters*, vol. 12, pp. 307–325, May 1991.

- [94] S.H.Y. Hung, "On the straightness of digital arcs", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-7, pp. 203–215, 1985.
- [95] K.-M. Eom and J. Park, "Contour models for curvature estimation and shape decomposition", in *Proc. of 11th International conference on Pattern Recognition*, vol. II, pp. 393–396, The Hague, The Netherlands, 1992.
- [96] T. Pavlidis and S.L. Horowitz, "Segmentation of plane curves", *IEEE Transactions on Computer*, vol. C-23, pp. 860–870, 1974.
- [97] Y.-N. Zhang, "Utilizing high level knowledge in middle level image analysis", in *Proc. of the Congress of International Society of Photogrammetry and Remote Sensing Commission III*, pp. 599–607, Washington, D.C. U.S.A., 1992.
- [98] S. Dudani, K.J. Breeding, and R.D. McGhee, "Aircraft identification by moment invariants", *IEEE Transactions on Computers*, vol. 26, pp. 39–45, 1977.
- [99] S.K. Parui, S. Eswara Sarma, and D. Dutta Majumder, "How to discriminate shapes using the shape vector", *Pattern Recognition Letters*, vol. 4, pp. 201–204, 1986.
- [100] Y.J. Tejjwani and R.A. Jones, "Machine recognition of partial shapes using feature vectors", *IEEE Transactions on System, Man, and Cybernetics*, vol. 15, pp. 504–516, 1985.
- [101] A. Rosenfeld and A.C. Kak, *Digital picture processing*, Academic Press, New York, 1982.
- [102] L.S. Davis, "Shape matching using relaxation techniques", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 1, pp. 60–72, 1977.
- [103] B. Bhanu and O.D. Faugeras, "Shape matching of two-dimensional objects", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, pp. 137–156, 1984.
- [104] R.L. Kashyap and B.J. Oommen, "A geometric approach to polygonal dissimilarity and shape matching", in *Proc. of International Conference on Pattern Recognition*, pp. 472–479, 1982.
- [105] W.H. Tsai and S.S. Yu, "Attributive string matching with merging for shape recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 7, pp. 453–462, 1985.
- [106] E. Salari and S. Balaji, "Recognition of partially occluded objects using B-spline representation", *Pattern Recognition*, vol. 24, pp. 653–660, 1991.
- [107] B. Kamgar-Parsi, A. Margalit, and A. Rosenfeld, "Matching general polygonal arcs", *CVGIP: Image Understanding*, vol. 53, pp. 227–234, 1991.

- [108] D. Sherman and S. Peleg, "Stereo by incremental matching of contours", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, pp. 1102–1106, Nov. 1990.
- [109] M.-H. Han and D. Jang, "The use of maximum curvature points for the recognition of partially occluded objects", *Pattern Recognition*, vol. 23, pp. 21–33, 1990.
- [110] A.M. Bruckstein, R.J. Holt, A.N. Netravali, and T.J. Richardson, "Invariant signatures for planar shape recognition under partial occlusion", *CVGIP: Image Understanding*, vol. 58, pp. 49–65, July 1993.
- [111] D.J. Brauneegg, "Stereo feature matching in disparity space", A.I. memo, AI laboratory, Massachusetts Institute of Technology, September 1989.
- [112] Z. Pizlo, "Recognition of planar shapes from perspective images using contour-based invariants", *CVGIP: Image Understanding*, vol. 56, pp. 330–350, November, 1992.
- [113] F. Mokhtarian and A. Mackworth, "Scale-based description and recognition of planar curves and two-dimensional shapes", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, pp. 34–44, 1986.
- [114] G. Borgefors, "Hierarchical chamfer matching: a parametric edge matching algorithm", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, pp. 849–865, 1988.
- [115] H.-C. Liu and M.D. Srinath, "Partial shape classification using contour matching in distance transformation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, pp. 1072–1079, Nov. 1990.
- [116] G. Borgefors, "Distance transformations in digital images", *Computer Vision, Graphics, and Image Processing*, vol. 34, pp. 344–371, 1986.
- [117] H. Tahani and J.M. Keller, "Information fusion in computer vision using the fuzzy integral", *IEEE Transactions on System, Man, and Cybernetics*, vol. 20, pp. 733–741, may 1990.
- [118] R. Kruse, "Fuzzy integrals and conditional fuzzy measure", *Fuzzy Sets Systems*, vol. 10, pp. 309–313, 1983.
- [119] L.A. Zadeh, "Fuzzy sets", *Information and Control*, vol. 8, pp. 338–353, 1965.
- [120] P.R. Halmos, *Measure theory*, Van Nostrand, New York, 1950.
- [121] W.F. Pfeffer, *Integrals and measures*, Marcel Dekker, New York, 1977.
- [122] G.A. Shafer, *A mathematical theory of evidence*, Princeton University Press, Princeton, NJ, 1976.

- [123] L.A. Zadeh, "Fuzzy sets as a basis for a theory of possibility", *International Journal of Fuzzy Sets System*, vol. 1, pp. 3–28, 1978.
- [124] T. Terano and M. Sugeno, "Macroscopic optimization by using conditional fuzzy measures", in M. Gupta and E. Sanches, editors, *Fuzzy Information and Decision Processes*, pp. 79–86. North-Holland, New York, 1982.
- [125] D. Dubois and H. Prade, "A class of fuzzy measures based on triangular norms", *Int. J. Gen. Syst.*, vol. 8, pp. 43–61, 1982.
- [126] K. Leszczynski, P. Penczek, and W. Grochulski, "Sugeno's fuzzy measures and fuzzy clustering", *Fuzzy Sets Systems*, vol. 15, pp. 147–158, 1985.
- [127] S.T. Wierzbach, "On fuzzy measure and fuzzy integral", in M. Gupta and E. Sanches, editors, *Fuzzy Information and Decision Processes*, pp. 79–86. North-Holland, New York, 1982.
- [128] L.A. Zadeh, "A computational approach to fuzzy quantifiers in natural languages", *Comput. Math. Applicat.*, vol. 9, pp. 149–184, 1983.
- [129] R.R. Yager, "On ordered weighted averaging aggregation operators in multicriteria decision making", *IEEE Transactions on System, Man, and Cybernetics*, vol. 18, pp. 183–190, 1988.
- [130] Y. Sekita and Y. Tabata, "A consideration on identifying fuzzy measures", in *XXIII Intn. Mtg. of the Institute of Management Sciences*, Athens, July 1977.
- [131] S.T. Wierzbach, "An algorithm for identification of fuzzy measure", *Fuzzy Sets and Systems*, vol. 9, pp. 69–78, 1983.
- [132] J.A. Feldman, "Four frames suffice: a provisional model of vision and space", *Behavioral Brain Sciences*, vol. 8, pp. 265–313, 1985.
- [133] D.H. Ballard and C.M. Brown, *Computer Vision*, Prentice-Hall, Inc., 1982.
- [134] J. Y. Aloimonos, L.S. Davis, and A. Rosenfeld, "Maryland progress in image understanding", in *Proc. of Image Understanding workshop*, 1990.
- [135] J.R. Ullman, "A consistency technique for pattern association", *T-IRE (IT)*, vol. 8, pp. 74–81, 1962.
- [136] H.B. Clowes, "On seeing things", *Artificial Intelligence*, vol. 2, pp. 79–116, 1971.
- [137] D.A. Huffman, "Impossible objects as nonsense sentences", in Metzger B. and Michie D., editors, *Machine Intelligence*, vol. 6, pp. 295–323. Edinburgh University Press, Edinburgh, 1971.
- [138] S.W. Zucker, "Region growing: childhood and adolescence", *Computer Graphics and Image Processing*, vol. 5, pp. 382–399, 1976.

- [139] J. Kittler and J. Foglein, "Contextual classification of multispectral pixel data", *Image Vision Computing*, vol. 2, pp. 13–29, 1984.
- [140] R.M. Haralick, "Decision making in context", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 5, pp. 417–428, 1983.
- [141] D.L. Waltz, "Understanding line drawings of scenes with shadows", in Winston P.H., editor, *The Psychology of Computer Vision*. McGraw-Hill, New York, USA, 1957.
- [142] A. Rosenfeld, R.A. Hummel, and S.W. Zucker, "Scene labeling by relaxation operations", *IEEE Transactions on Syst. Man Cybern.*, vol. SMC-6, pp. 420–453, June 1976.
- [143] U. Montanari, "Networks of constraints: fundamental properties and applications to picture processing", *Inf. Sci.*, vol. 7, pp. 95–132, 1974.
- [144] A.K. Mackworth, "Consistency in networks of relations", *Artificial Intelligence*, vol. 8, pp. 99–118, 1977.
- [145] R.M. Haralick and L.G. Shapiro, "The consistent labeling problem: Part I", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-1, pp. 173–184, 1979.
- [146] R.M. Haralick and L.C. Shapiro, "The consistent labeling problem: Part II", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-2, pp. 193–204, 1980.
- [147] G. Henderson, "A note on discrete relaxation", *Computer Vision, Graphics and Image Processing*, vol. 28, pp. 384–388, 1984.
- [148] J.R. Ullman, R.M. Haralick, and L.G. Shapiro, "Computer architectures for solving consistent labeling problems", *Computer Journal*, vol. 28, 1985.
- [149] L.G. Shapiro, "Solving consistent labeling problems having the separation property", in *Proc. of Seventh International Conference on Pattern Recognition*, pp. 313–315, Montreal Canada, July 1984.
- [150] B. Nudel, "Consistent-labeling problem and their algorithms: expected complexities and theory based heuristics", *Artificial Intelligence*, vol. 21, pp. 135–178, 1983.
- [151] S. Nishihara and K. Ikeda, "A constraint synthesizing algorithm for the consistent labeling problem", in *Proc. of Seventh International Conference on Pattern Recognition*, pp. 310–312, Montreal, Canada, July 1984.
- [152] S. Kasif and A. Rosenfeld, "The fixed points of images and scenes", in *Proc. of CVPR Conference*, pp. 454–456, Washington, USA, 1983.

- [153] S.A. Kuschel and C.V. Page, "Augmented relaxation labeling and dynamic relaxation labeling", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-4, pp. 676–683, 1982.
- [154] H.M. Kalayeh and D.A. Landgrebe, "Adaptive relaxation labeling", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-6, pp. 369–372, 1984.
- [155] J.Y. Koo, K.H. Park, and M. Kim, "Improving the labeling accuracy by a new probabilistic relaxation labeling", *Pattern Recognition Letters*, vol. 3, pp. 399–402, 1985.
- [156] J.F. Boyce, J. Feng, and E.R. Haddow, "Relaxation labelling and the entropy of neighbourhood information", *Pattern Recognition Letters*, vol. 6, pp. 225–234, 1987.
- [157] H. Shvaytser and S. Peleg, "A new approach to the consistent labeling problem", in *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 320–327, San Francisco, CA, June 10-13.
- [158] J. Kittler and J. Illingworth, "Relaxation labelling algorithms-a review", *Image and Vision Computing*, vol. 3, pp. 206–216, 1985.
- [159] E.R. Hancock, M. Haindl, and J. Kittler, "Multiresolution edge labelling using hierarchical relaxation", in *Proc. of 11th IAPR International Conference on Pattern Recognition*, vol. II, Conference B: Pattern Recognition Methodology and Systems, pp. 140–144, The Hague, The Netherlands, August 1992. IEEE Computer Society Press.
- [160] N. Krishnakumar, S. Sitharama Iyengar, R. Holyer, and M. Lybanon, "Feature labelling in infrared oceanographic images", *Image and Vision Computing*, vol. 8, pp. 142–147, 1990.
- [161] J.H. Duncan and T. Birkholzer, "Reinforcement of linear structure using parametrized relaxation labeling", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 502–515, 1992.
- [162] Chakravarty, "A generalized line and junction labeling scheme with applications to scene analysis", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-1, pp. 202–206, 1979.
- [163] J.A. Richards, D.A. Landgrebe, and P.H. Swain, "Pixel labeling by supervised probabilistic relaxation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-3, pp. 188–191, 1981.
- [164] E.R. Hancock and J. Kittler, "Edge-labeling using dictionary-based relaxation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-12, pp. 165–181, 1990.

- [165] H.M. Alnuweiri and V.K. Prasanna Kumar, "Fast image labeling using local operators on mesh-connected computers", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-13, pp. 202–207, 1991.
- [166] J.K. Udupa and V.G. Ajjanagadde, "Boundary and object labelling in 3D images", *Computer Vision Graphics and Image Processing*, vol. 51, pp. 355–369, 1990.
- [167] O.D. Faugeras and K.E. Price, "Semantic description of aerial images using stochastic labeling", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-3, pp. 633–643, 1981.
- [168] U. Eckhardt and G. Maderlechner, "Thinning binary pictures by a labeling procedure", in *Proc. of 11th IAPR International Conference on Pattern Recognition*, vol. III, Conference C: Image Speech and Signal Analysis, pp. 582 – 585, The Hague, The Netherlands, August 30. IEEE Computer Society Press.
- [169] J.Y. Hsiao and A.A. Sawchuk, "Supervised textured image segmentation using feature smoothing and probabilistic relaxation techniques", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-11, pp. 1279–1292, 1989.
- [170] M.C. Ibson and L. Zapalowski, "On the use of relaxation labelling in the correspondence problem", *Pattern Recognition Letters*, vol. 4, pp. 103–110, 1986.
- [171] T.A. Jamison and R.J. Schalkoff, "Image labelling: A neural network approach", *Image and Vision Computing*, vol. 6, Nov. 1988.
- [172] H.M. Alnuweiri and V.K. Prasanna, "Parallel architectures and algorithms for image component labeling", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 1014–1034, 1992.
- [173] H.-H. Liu, T.Y. Young, and A. Das, "A multilevel parallel processing approach to scene labeling problems", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-10, pp. 586–590, 1988.
- [174] Shiaw-Shian Yu and Wen-Hsiang Tsai, "Relaxation by the hopfield neural network", *Pattern Recognition*, vol. 25, pp. 197–210, 1992.
- [175] D. Reisis and V.K. Prasanna Kumar, "Parallel processing of the labeling problem", in *Proc. of IEEE Computer Society Workshop on Computer Architecture for Pattern Analysis and Image Database Management*, pp. 381–385, Miami Beach FL., November 1985.
- [176] P. Parent and S. W. Zucker, "Radial projection: An efficient update rule for relaxation labeling", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-11, pp. 886–889, 1989.
- [177] X. Zhuang, R.M. Haralick, and H. Joo, "A simplex-like algorithm for the relaxation labeling process", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-11, pp. 1316–1321, 1989.

- 
- [178] L. Kirousis, "Effectively labeling planar projections of polyhedra", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-12, pp. 123–130, 1990.
- [179] Z. Wu and R. Leahy, "An optimal graph theoretic approach to data clustering: theory and its application to image segmentation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 1101–1113, Nov. 1993.

# Integratie van Segmentatie en Stereo-Matching

## Samenvatting

Twee ogen of camera's die kijken naar dezelfde objecten vanuit verschillend perspectief verschaffen de middelen om driedimensionale vorm en positie te bepalen. Wetenschappelijk onderzoek naar dit effect wordt betiteld met termen als *stereo vision* of *stereopsis*. Stereo vormt een belangrijke methode voor machineperceptie omdat het leidt tot relatief rechtstreekse metingen doordat, in tegenstelling tot bij enkelbeeld technieken, hier diepte niet wordt afgeleid onder zwakke en niet verifieerbare fotometrische of statistische aannamen, en omdat er voorts geen specifieke gedetailleerde modelkennis over de objecten bij benodigd is.

Segmentatie en *stereo matching* vormen twee essentiële stappen in *stereo vision* en zijn moeilijk op te lossen problemen. Binnen de traditionele benadering worden segmentatie en *stereo vision* gewoonlijk afzonderlijk behandeld. Het segmentatieproces –in dit proefschrift behandelen we alleen *region based* segmentatie– beoogt te komen tot een opdeling van een beeld in gebieden die uniform en homogeen zijn ten aanzien van een of meerdere eigenschappen als grijswaarde en textuur. Segmentatietechnieken zijn in wezen ad hoc van karakter. Hun onderlinge verschil ligt in feite in de wijze waarop ze een of meer van de gewenste gebiedseigenschappen benadrukken en de wijze waarop ze de gewenste eigenschappen vervolgens tegen elkaar afwegen. De gebieden in een beeld ontstaan niet alleen ten gevolge van de geometrische eigenschappen van de oppervlakken van voorwerpen, maar ook als gevolg van hun optische eigenschappen, de richting van het zonlicht, schaduwen etcetera. Verschillen in benadering van beeldmodellen en verschillende implementaties leiden tot een variëteit aan segmentatietechnieken. Veel onderzoekers hebben geconstateerd dat er niet één enkel methode bestaat die op zich een volledige segmentatie van een beeld kan leveren. Iedere methode echter kan mogelijk wel een deel van de informatie opleveren die nodig is voor een meer betekenisvolle interpretatie van de scene. Het is redelijk om te verwachten dat er problemen op zullen treden bij het “versmelten” (Eng: *fusing*) van resultaten afkomstig van verschillende methoden en uit verschillende bronnen, zoals in dit proefschrift de stereobeelden.

*Stereo Matching* is nodig om de resultaten van stereobeelden te kunnen combineren, maar daar staat tegenover dat hiervoor op zijn beurt weer een goede segmentatie vereist is.

Onvolkomen segmentatie en mogelijke occlusie maken het moeilijk om gebieden in twee beelden als projecties van een en hetzelfde object te vergelijken. Om een goed matching resultaat te verkrijgen is het nodig deze gebieden na het matchen, nogmaals te segmenteren. Uitgaande van bovenstaande constatering stellen we een nieuw schema voor om segmentatie en *stereo matching* te integreren. Na een eerste *region-based* segmentatie wordt een kandidaat stereo matching uitgevoerd die gebieden uit het ene beeld toekent aan corresponderende gebieden uit het andere door middel van vorm overeenkomst (Eng: *shape matching*). Tijdens de volgende segmentatiestap wordt stereoinformatie meegenomen. Dit geschiedt door tijdens de afweging over het al dan niet samenvoegen van een gebied met aangrenzende gebieden in het beeld, ook de (stereo-) corresponderende gebieden van de kandidaten hierbij te betrekken. Deze matching wordt dan uitgevoerd gebruikmakend van intensiteits- en vorminformatie uit beide beelden. Tenslotte wordt een globale matching uitgevoerd waarbij de andere matchingscriteria zoals uniciteit, ordening en topologische relaties worden meegenomen.

## Deel 1: Segmentatie

Hoofdstuk 2 en 3 houden zich bezig met de segmentatieproblematiek. We gebruiken het minimale beschrijvingslengte (Eng: *Minimal Description Length (MDL)*) principe om beeldintensiteiten te coderen en introduceren een robuuste schattingsmethode om het intensiteitsmodel van het beeld uit de beeldgegevens te bepalen. Tevens ontwikkelen we een aanpak voor een op vormcriteria gebaseerde segmentatie die *region-growing* en *region boundary fitting* integreert. Het vormmodel dat gebruikt wordt bestaat uit een gesloten polygoon gerepresenteerd door een aantal gladde segmenten. Om vorm voorwaarden te verkrijgen introduceren we een methode voor optimale curve passing (Eng: *curve fitting*) die de techniek van extremadetectie met een splitsingsmethode combineert.

## Deel 2: Lokale matching

Hoofdstuk 4,5 en 6 behandelen de lokale matching waarbij slechts de lokale attributen van beeldgebieden in ogenschouw worden genomen. Om een nauwkeurige matching van de grenzen van stereogebieden te verkrijgen wordt een methode beschreven voor het oplossen van het *stereo matching* probleem voor het algemene geval van gesloten vlakke curven onder de conditie van mogelijk onvolkomen segmentatie en aanwezige occlusies, uitgaande van gegeven cameraparameters.

De methode berekent parameters gerelateerd aan een plat objectvlak, dat wil zeggen: de hellingshoeken (Eng: *slant* en *tilt*) en schaalfactor, en benut een histogramtechniek

om deze parameters te schatten. De parameterschatting is gebaseerd op de dispariteitsinformatie van de curven. Punt-correspondentie speelt een belangrijke rol in de methode. We lossen het probleem op met behulp van dynamisch programmeren. De uiteindelijke matching wordt vastgesteld door het toepassen van een afstands-transformatie (Eng: *distance transformation*)

Omdat aan elk beeld gebiedje een aantal attributen toegekend zijn, dient het probleem van het samenvoegen van stereoaanwijzingen aangepakt te worden binnen de lokale matching stap. In dit proefschrift gebruiken we de theorie van vage maten (*fuzzy measure*) en vage integralen (*fuzzy integral*). De vage maat- en integraaltheorie leveren een algemeen raamwerk voor het representeren en samenvoegen van multi-criteria informatie. Uitgaande van de ideeën van Yager generaliseren we de vage integraaltheorie om aldooende een verscheidenheid aan aggregatiefuncties te verkrijgen. Een groot probleem verbonden aan het gebruik van vage maat en integraal vormt het juist identificeren en bepalen van de vage maat. We ontwikkelen een leerprocedure om nauwkeurige formules te verkrijgen bedoeld voor XS informatiefusie.

Hoofdstuk 6 bevat een gedetailleerde discussie omtrent de integratie van segmentatie en *stereo matching*. De introductie van terugkoppeling stelt het voorgestelde systeem in staat de matching resultaten terug te sturen naar de her-segmentatietrap om zodoende het segmentatieresultaat te verbeteren en bijgevolg het matchingsresultaat.

## Deel 3: Globale matching

De globale matching, waarin de topologische informatie en andere matching criteria in rekening worden genomen, zijn onderwerp van bespreking in hoofdstuk 7. De lokale matching vergelijkt slechts de lokaal gedefinieerde attributen van stereogebeden. Het kan voorkomen dat de lokale matching meerdere match mogelijkheden oplevert, d.w.z. een enkel gebied uit bijvoorbeeld het linker beeld kan meerdere gebieden uit het rechterbeeld toegewezen krijgen. Het doel van de globale matching is om unieke matches te verkrijgen met inachtneming van topologische en andere beperkingen. We formuleren globale matching als een relaxatieprobleem. Daartoe modificeren we de traditionele relaxatiemethode op enkele punten om aan onze wensen tegemoet te komen: 1) in de *support* functie wordt een zachte drempel toegepast om slechte matches te onderdrukken en mogelijke goede matches te bevorderen; 2) een balansfactor wordt opgevoerd om de invloed te sturen tussen de topologische beperkingen en de initiële waarschijnlijkheden afkomstig uit de lokale matching. In dit hoofdstuk worden de uiteindelijke resultaten van dit proefschrift gepresenteerd.



# Acknowledgements

This thesis would not be successfully finished without the help from many people. First of all, I would thank my supervising committee (Prof. E. backer, Prof. G.H. Ligterink, Dr. J.J. Gerbrands, and ir. M.J.M.P. Lemmens) for helping to focus and sharpen the topic, contributing insights to the finished work, and providing criticism on my manuscripts during many meetings.

The Information Theory Group in the Department of Electrical Engineering has provided me a good facility to carry out the research. I am grateful to everybody at the Information Theory Group for all the help, especially to the people working in the image processing field: Jan van Lieshout (he also translated the summary into Dutch), Freek Odijk, Peter van Beek, Marcel Reinders, and Herjan Barnard.

To Prof. M.J.M. Bogaerts I owe many debts of gratitude. Without his continuous support, it would be impossible for me to get a chance and go on to pursue the PhD in the Netherlands. I also would like to express my gratitude to everybody at the Section of Geo-Information in the Department of Geodesy for an enjoyable time, especially to Dr. M.J. Kraak, ir. H.J.G.L. Aalders, Prof. J. Gazdzicki, drs. T.P.M. Tijssen, mw. E.M. Seinstra-Fendel and mw. H. Verwest-Sinnema for all the help.

I am obliged to Geeris Holding Netherlands BV for supporting some of my work. I would also thank the Pattern Recognition Group in the Department of Applied Physics for some useful help, Mrs. J.B. Zaat-Jones for correcting the English text, and Prof. M. Molenaar and Prof. G. Vosselman for useful discussions.

Without my education in China, my Ph.D would be groundless. I sincerely thank many of my teachers and colleagues in China for giving me the necessary support and help. I am particularly engaged to thank my teacher Prof. CHEN Ruie for the high expectation and generous suggestion.

Doing PhD research is a long journey, which needs high concentration, and sometime means the neglect of family affairs. I am highly indebted to my wife XIE Chen and my parents for their encourage, care and support.



# Curriculum Vitae

Yaonan ZHANG was born in Jiangsu Province, China, on July 21, 1962. He received the degrees of BSc and MSc in Photogrammetry and Remote Sensing from Zhengzhou Institute of Surveying and Mapping, China, in 1982 and 1985, respectively. From 1988 to 1989, he studied at the International Institute of Aerospace Survey and Earth Science (ITC), Enschede, the Netherlands, in Geographic Information System (GIS), and graduated with a diploma (with distinction). Since November 1989, he was employed by the Centre for Computer Graphics and Mapping, TU Delft and also by the FRANK project (sponsored by Geeris Holding Netherlands BV). In July, 1992, he became a full AIO (research assistant), jointly with the Departments of Electrical Engineering and Geodesy, TU Delft.

His MSc thesis was on a study of interpolation methods and accuracy estimation for digital elevation model. At ITC, he developed a prototype for a cadastral information system and studied the integrity problem of GIS. During employment with the FRANK project, he was responsible for image rectification, camera calibration, feature extraction, etc. The main interests during his PhD research include: segmentation, matching, vision module integration, fusion of visual information, etc. His future interests are related to the analysis of image sequences, mathematical analysis of vision problems and model based image processing in various applications.

During the PhD research period, he supervised several Master students at the Information Theory Group, Department of Electrical Engineering, TU Delft, in the areas of artificial neural networks and medical image processing.

He is a member of the International Association for Pattern Recognition and the International Society for Photogrammetry and Remote Sensing.



# List of Publications

H.C. Kok, Y.-N. Zhang, and J.J. Gerbrands, "Detection of the left ventricle from MRI time sequences using the Hough transform and B-splines", submitted to *Computing Science in the Netherlands 1994 (CSN94)*, Utrecht, The Netherlands, Nov. 21-22, 1994.

Y.-N. Zhang, M. J. P. M. Lemmens, and G.H. Ligterink, "Initial probabilities, minimal inconsistent labeling and application to stereo matching", in *Proc. of International Conference on Volume Image Processing*, Utrecht, The Netherlands, 1993. also submitted to *Pattern Recognition Letters*, in revision.

Y.-N. Zhang, "Integration of segmentation and stereo matching", in *Proc. of SPIE's Visual Communications and Image processing*, Cambridge, MA, Nov. 1993.

Y.-N. Zhang, J.J. Gerbrands, "A method for matching general planar curves", submitted to *Image and Vision Computing*.

Y.-N. Zhang, J.J. Gerbrands, and E. Backer, "Shape constrained region growing", to be submitted to *Pattern Recognition*.

Y.-N. Zhang, "A new stereo matching approach in image/object dual spaces", in *Proc. of XVII Congress of International Society for Photogrammetry and Remote Sensing, Commission III*, Washington, D.C., U.S.A., 1992.

Y.-N. Zhang, "Deformable model for image segmentation", in *Proc. of XVII Congress of International Society for Photogrammetry and Remote Sensing, Commission IV*, Washington, D.C., U.S.A., 1992.

Y.-N. Zhang, "Utilizing high level knowledge in image analysis", in *Proc. of XVII*

*Congress of International Society for Photogrammetry and Remote Sensing, Commission III*, Washington, D.C., U.S.A., 1992.

Y.-N. Zhang, "Geometrically constrained segmentation for remote sensing images", in *Proc. of first workshop of the Association for Chinese Professionals in GIS*, Baffulo, NY, Aug. 1992.

Y.-N. Zhang, "Matching in image-object dual spaces", in *Proc. of SPIE's Industrial Vision Metrology*, vol. 1526, 1991.

Y.-N. Zhang, "A pool-based searching approach for self calibration of multi-cameras and motion analysis under possible incorrect correspondences", technical report, Delft University of Technology, 1991.

Y.-N. Zhang, "Rectification of images by fish-eye lens", technical report, Delft University of Technology, 1990.

Y.-N. Zhang, "Accuracy estimation of DTM using high sampling density profiles", in *Proc. of XVI Congress of International Society for Photogrammetry and Remote Sensing*, Kyoto, Japan, 1988.