

DIRECT SURFACE RECONSTRUCTION FROM MULTIPLE VIEWS BY  
SURFACE GROWING

A THESIS SUBMITTED TO  
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES  
OF

THE MIDDLE EAST TECHNICAL UNIVERSITY

BY

UĞUR MURAT LELOĞLU

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

IN

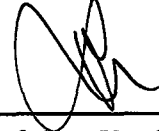
THE DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

JUNE 2001

116507

116507

Approval of the Graduate School of Natural and Applied Sciences.



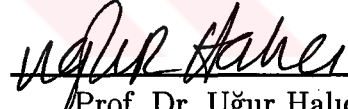
Prof. Dr. Tayfur Öztürk  
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Doctor of Philosophy.



Prof. Dr. Fatih Canatan  
Head of Department

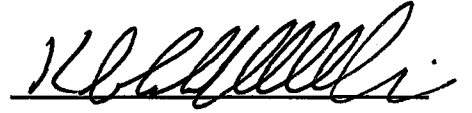
This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Doctor of Philosophy.



Prof. Dr. Uğur Halıcı  
Supervisor

Examining Committee Members

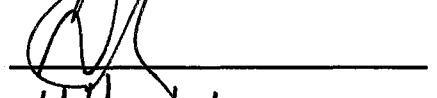
Prof. Dr. Kemal Leblebicioğlu



Prof. Dr. Uğur Halıcı



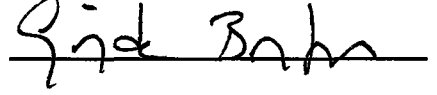
Prof. Dr. Turhan Çiftçibaşı



Assoc. Prof. Dr. Volkan Atalay



Assoc. Prof. Dr. Güzde Bozdağı



# ABSTRACT

## DIRECT SURFACE RECONSTRUCTION FROM MULTIPLE VIEWS BY SURFACE GROWING

Lelođlu, Uđur Murat

Ph. D., Department of Electrical and Electronics Engineering

Supervisor: Prof. Dr. Uđur Halıcı

June 2001, 80 pages

To obtain 3D surface descriptions from multiple optical images taken at arbitrary viewpoints is a challenging task of computer vision. The usual approach is first matching the images point by point to find 3D points, then trying to fit surfaces to resulting points. In this work, a radically different approach is presented. The surfaces that generate the images are estimated directly in the object space without explicit image matching phase and without an initial guess for the surfaces.

To be able to solve this complex problem, an attention mechanism is employed: Instead of reconstructing all surfaces at once, we begin from a point of attention, i.e., a small surface patch serving as a seed. The surfaces are reconstructed one by one through a surface growing approach where the surface parameters are continuously updated as new surfels (surface elements) are added. The algorithm is hybrid: Both re-projection error on multiple images and a novel feature match function guide the surface growing. For each surface, radiometric correction parameters are estimated and it is shown that surface patches are the right scale for radiometric correction parameter estimation. Resulting surface patches are accepted only if converging evidence is collected from both re-projection error and from matching features, over a relatively large area on multiple images. The final result can directly be used as 3D surface models. The performance of the algorithm is demonstrated on real images.

Keywords: 3D reconstruction, image matching, surface growing

# ÖZ

## YÜZEY BÜYÜTME İLE ÇOK SAYIDA GÖRÜNTÜDEN DOĞRUDAN YÜZEY GERİÇATMA

Leloğlu, Uğur Murat

Doktora, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Yöneticisi: Prof. Dr. Uğur Halıcı

Temmuz 2001, 80 sayfa

Rastgele noktalardan alınmış birkaç optik görüntüden üçboyutlu yüzey tanımları elde etme yapay görünün zorlu problemlerindedir. Alışılmış yaklaşım, önce görüntüleri nokta nokta eşleyerek üçboyutlu noktaları bulmak, daha sonra da bulunan noktalara yüzeyler oturtmaya çalışmaktır. Bu çalışmada, tamamen farklı bir yaklaşım sunulmuştur. Görüntüleri meydana getiren yüzeyler, açık bir eşleme aşaması veya bir başlangıç tahmini olmadan, doğrudan nesne uzayında bulunmaktadır.

Bu karmaşık problemi çözebilmek için bir dikkat mekanizması kullanıldı: Bütün yüzeyleri aynı anda geriçatmak yerine, bir dikkat noktasından, yani tohum görevi görece küçük bir yüzey parçasından başlanmaktadır. Yüzeyler, yeni yüzey öğeleri eklendikçe yüzey parametrelerinin sürekli olarak güncellendiği bir yüzey büyütme yaklaşımıyla birer birer geriçatılmaktadır. Algoritma melezdir: Hem çok sayıda görüntüye yeniden izdüşüm hatası, hem de yeni bir öznitelik eşleme işlevi yüzey büyütme yol gösterir. Ortaya çıkan yüzeyler ancak, hem yeniden izdüşüm hatasından, hem de eşlenen özniteliklerden, çok sayıda görüntü üzerinde görece geniş bir alanda, birbirlerini güçlendiren kanıt toplandığında kabul edilmektedir. Sonuçlar doğrudan üçboyutlu modeller olarak kullanılabilir. Algoritmanın başarımı gerçek görüntüler üzerinde gösterilmiştir.

Anahtar Kelimeler: Üç-boyutlu geriçatma, görüntü eşleme, yüzey büyütme

All colors made me happy: even gray.  
My eyes were such that literally they  
Took photographs. Whenever I'd permit,  
Or, with a silent shiver, order it,  
Whatever in my field of vision dwelt -  
An indoor scene, hickory leaves, the swelte  
Stilettoes of a frozen stillicite -  
Was printed on my eyelids' nether side  
Where I would tarry for an hour or two,  
And while this lasted all I had to do  
Was close my eyes to reproduce the leaves,  
or indoor scene, or trophies of the eaves.

Pale Fire  
V. Nabokov

To my Father

## ACKNOWLEDGMENTS

Firstly, I would like to express my gratitude to Prof. Dr. Uğur Halıcı, Prof. Dr. Kemal Leblebicioğlu and Assoc. Dr. Volkan Atalay for their guidance and suggestions. I thank to Prof. Dr. Henri Maître, Dr. Michel Roux and the people in the image department of ENST for their help during my stay in Paris. I also thank to friends in Signal Processing and Remote Sensing Laboratory. Special thanks to Erol Tunalı who carried great amount of load instead of me during thesis writing. Most importantly, I would like to thank to my wife Zeynep and to my family, especially to my father.

Besides, I would like to thank Eurosense Inc. (Wemmel, Belgium) and the IMPACT project (see [90] for more information on the project) for supplying us the fully calibrated aerial images of Brussels and the VRML model and to IGP at Swiss Federal Institute of Technology for the AMOBE dataset [37].

Finally, I would like to acknowledge TÜBİTAK-BİLTEN and Prof. Dr. Murat Aşkar for continued support and TÜBİTAK-BAYG for financing half of my research period in ENST.



# TABLE OF CONTENTS

ABSTRACT . . . . .	iii
ÖZ . . . . .	v
DEDICATION . . . . .	vii
ACKNOWLEDGMENTS . . . . .	vii
TABLE OF CONTENTS . . . . .	ix
LIST OF TABLES . . . . .	xii
LIST OF FIGURES . . . . .	xiii
CHAPTER	
1 INTRODUCTION . . . . .	1
1.1 Problem Definition . . . . .	1
1.2 Introduction . . . . .	2
1.3 Organization of the Thesis . . . . .	6
2 BACKGROUND ON 3D SURFACE RECONSTRUCTION FROM MULTIPLE IMAGES . . . . .	7
2.1 Image Formation . . . . .	8
2.2 Camera Calibration . . . . .	11
2.3 Triangulation . . . . .	12
2.4 Image Matching . . . . .	13
2.4.1 Geometry of Matching . . . . .	14
2.4.2 Problems . . . . .	17
2.4.3 Constraints . . . . .	17

2.4.4	Primitives . . . . .	18
2.4.4.1	Features . . . . .	18
2.4.4.2	Area-Based Primitives . . . . .	19
2.4.4.2.1	Surface Reflectance Model . . . . .	19
2.4.4.2.2	Photometric Distortion . . . . .	20
2.4.4.2.3	Two-Image Correlation-Like Measures . . . . .	21
2.4.4.2.4	Multi-Image Correlation-Like Measures . . . . .	22
2.4.4.2.5	Matching Color Images . . . . .	24
2.4.5	Strategies . . . . .	25
2.4.5.1	Region Growing on Disparity Maps . . . . .	25
2.5	3D Surface Reconstruction from Multiple Images . . . . .	26
2.5.1	Surface Fitting to Unorganized Data Points . . . . .	26
2.5.2	Minimizing Re-Projection Error by Optimization . . . . .	27
2.5.3	Volumetric Methods . . . . .	28
2.5.4	Generating Surfaces in 3D . . . . .	28
2.6	Error Assessment . . . . .	30
3	3D RECONSTRUCTION BY SURFACE GROWING . . . . .	33
3.1	Surface Representation . . . . .	34
3.2	Hypothesis Generation . . . . .	35
3.3	Re-Projection Error and Radiometric Correction . . . . .	39
3.4	Feature Match Function . . . . .	40
3.5	Surface Growing . . . . .	42
3.6	Feature Matching . . . . .	44
3.7	Surface Verification . . . . .	45
3.8	Surface Merging . . . . .	46
3.9	Boundary Detection and Correction . . . . .	47
3.10	Surface Growing and Frequently Used Constraints . . . . .	47
3.11	Algorithm Complexity . . . . .	48

4	EXPERIMENTAL RESULTS . . . . .	51
4.1	Quality Assessment . . . . .	61
5	CONCLUSION . . . . .	69
5.1	Future Work . . . . .	70
	REFERENCES . . . . .	72
	VITA . . . . .	80



## LIST OF TABLES

### TABLE

4.1	Error means and variances for the surface shown in Figure 4.11. . . . .	60
4.2	Error means and variances for the surface shown in Figure 4.2.	61



## LIST OF FIGURES

### FIGURE

2.1	Image projection from 3D to the image plane. . . . .	9
2.2	Finding the 3D coordinates of a point by triangulation. . . . .	13
2.3	Epipolar geometry. . . . .	15
2.4	A plane viewed by two cameras. . . . .	16
3.1	Flow diagram of the main algorithm. . . . .	34
3.2	An Example Planar Surface. . . . .	36
3.3	Interest points detected on an aerial image. . . . .	38
3.4	The distance from a line segment to another one. . . . .	42
3.5	The Gaussian functions used in the line segment similarity function. . . . .	43
3.6	Flow diagram of the surface growing algorithm. . . . .	49
3.7	Projections of three line segments on the 2D infinite line. . . . .	50
3.8	Two strips at both sides of the line is used to detect surface boundaries. . . . .	50
4.1	Three Brussels images used from the IMPACT dataset. . . . .	53
4.2	Example of a growing surface. . . . .	54
4.3	Reconstructed surfaces re-projected on one of the Brussels images. Each surface is shown in a different color. . . . .	55
4.4	3D rendering of the surfaces obtained from the Brussels images (With texture). . . . .	56
4.5	3D rendering of the surfaces obtained from the Brussels images (Without texture). . . . .	56
4.6	3D rendering of a roof with the VRML model obtained manually. . . . .	57
4.7	Total re-projection error as a function of depth and angle around the unit vector $\mathbf{u}$ . . . . .	59
4.8	Four stages of surface growing coded in different colors. The evolution of the error function is inspected during the surface growing process. . . . .	60

4.9	From <i>a)</i> to <i>d)</i> : The evolution of the average re-projection error as the surface grows. The error is drawn as a function of rotation angles around the unit vectors $\mathbf{u}$ and $\mathbf{v}$ . <i>e)</i> and <i>f)</i> : The error functions drawn as a function of rotation angles around the unit vectors $\mathbf{u}$ and $\mathbf{v}$ , respectively. . . . .	62
4.10	From <i>a)</i> to <i>d)</i> : The evolution of the feature match function as the surface grows. Feature match is normalized to unity and is drawn as a function of rotation angles around the unit vectors $\mathbf{u}$ and $\mathbf{v}$ . <i>e)</i> and <i>f)</i> : The feature match function drawn as a function of rotation angles around the unit vectors $\mathbf{u}$ and $\mathbf{v}$ , respectively. . . . .	63
4.11	The largest surface patch projected on one of the Brussels images. This surface is used for analyzing the error distributions and photometric distortion. . . . .	64
4.12	The error distributions in each channel of each camera for the surface in Figure 4.2. . . . .	65
4.13	The error distributions in each channel of each camera for the surface in Figure 4.11. . . . .	65
4.14	Three images used from the AMOBE dataset. . . . .	66
4.15	Reconstructed surfaces re-projected on Zürich image. Each surface is shown in a different color. . . . .	67
4.16	3D rendering with texture (AMOBE Image). . . . .	67
4.17	3D rendering without texture (AMOBE Image). . . . .	68

# CHAPTER 1

## INTRODUCTION

### 1.1 Problem Definition

The reconstruction of 3D surfaces from multiple images taken by calibrated cameras at arbitrary points is a challenging research topic in computer vision.

It has many application areas like cartography, city planning, environmental research, military systems, line-of-sight analysis for telecommunication network planning, micro-climate simulations, risk assessment, architecture, film industry, virtual reality and reverse engineering. When smooth and textured surfaces are involved, the solution is relatively easy. But, in the domain of complex man-made environments, like urban areas, the problem becomes difficult due to factors like frequent depth discontinuities, large occlusions, textureless areas and glossy surfaces. Lighting differences among images and changing scenes (e.g. moving cars) introduces extra difficulties. Although, many labo-

ratories are conducting active research on this topic, the problem is not solved yet.

This thesis is focused on the 3D reconstruction of complex man-made scenes. We use color images and try to exploit most of the information available in multiple color channels. We assume that internal and external calibration information is readily available, because the multi-camera calibration problem is completely solved today: Even when the cameras are not calibrated and pose information is not collected by techniques like GPS and INS, cameras can be calibrated by classical bundle adjustment techniques [77], [83] or relatively new projective geometry methods [32], [27].

The number of images used for reconstruction by current algorithms range from two to hundreds. Two images do not carry enough information while large number of images are difficult to collect, store, calibrate and process. In this thesis, we will use 3-4 images for reconstruction. Although, the system developed is open to interactive usage, it is completely automatic.

## 1.2 Introduction

The stereo matching algorithms, which are commonly used for 3D reconstruction of smooth surfaces, fail when applied to high-resolution aerial images of urban areas. Because, such scenes contain frequent depth discontinuities and large occluded or textureless areas. To be able to build 3D models of such



scenes, multiple images are used. As a result, more information about the underlying structure is available: *i)* Some surfaces that cannot be seen in an image may be visible in another one. Using sufficient number of views with various viewing angles, even a complete description of all visible surfaces can be obtained. *ii)* Parametric surface estimation is statistically more robust because of the redundancy in multiple images. *iii)* An image point that cannot be matched to any point in a second image due to weak texture or matching ambiguities may be matched, when multiple images are used.

However, it should be kept in mind that not all algorithms using more than two images are “true multi-image algorithms”. A true multi-image algorithm, as defined by Collins [15], should be able to handle any number of cameras greater than two while keeping its complexity as a linear function of the number of images without creating a dominant eye, i.e., a reference camera.

Most algorithms for 3D surface reconstruction from two or more images work in two successive stages: *i)* Points and/or features are matched (either pairwise or in multiple images simultaneously) and they are projected into the object space. *ii)* Surfaces, which fit to these points and features in 3D, are estimated.

In this thesis, we suggest a radically different “true multi-image” approach: The surfaces are directly estimated from the images. The establishment of image correspondences and surface reconstruction are performed simultaneously.

Similar algorithms either deform an initial surface in order to minimize an error function or they reconstruct small surface patches and link them to form a surface afterwards. Our algorithm creates the surfaces directly without requiring an initial guess and handles the constraints, like surface continuity and orderedness, naturally.

It is assumed that *i)* several multi-band images of a scene are available, *ii)* the cameras are fully calibrated, *iii)* the scene is confined to a known volume and, *iv)* the scene contains mostly planar, opaque surfaces.

To be able to solve this complex problem, an attention mechanism is employed: We do not attempt to reconstruct all surfaces at once. Instead, we begin from a point of attention, i.e., a small surface patch serving as a seed. A surface growing approach is taken where the surface parameters are continuously updated as new surfels (surface elements, after Szeliski and Tonnesen [81]) are added until the surface cannot grow any more. This approach handles piecewise continuous surfaces naturally without imposing an explicit smoothness constraint.

Another important aspect of the algorithm is that it is both area and feature based. While surfels are added based on their re-projection error, a novel feature match function guide the growing of the surface and the updating of the surface parameters. Features on these surfaces are also matched after surface growing and are used for surface verification, surface boundary detection and

hypothesis generation.

During surface growing, the radiometric correction parameters are also estimated for the surface. It is shown in the thesis that a planar surface is the right scale of radiometric correction estimation. Global parameters estimated for a whole image are not satisfactory because the parameters vary from surface to surface. On the other hand, small rectangular windows are too small for accurate estimation of the parameters.

The matches obtained by only multi-image correlation or feature matching are not reliable. Even local mechanisms, like support collection from the neighborhood or comparing flanking regions of linear features, cannot resolve the ambiguity. However, our algorithm accepts a surface patch only if converging evidence is collected from both small re-projection error and matching features, over a relatively large area on multiple images.

Use of parametric surface models constrain the solution space, so that a surface can be fit to areas with very weak texture. On the other hand, specific object models are not used, so the system is flexible enough to model complex polyhedral objects. In this work, a planar surface model is used to demonstrate the validity of the idea, but it can be generalized to higher order surfaces.

## 1.3 Organization of the Thesis

This thesis is organized as follows. In Chapter 2, the related work is briefly reviewed. Chapter 3 describes the proposed surface growing algorithm in detail. In Chapter 4, experimental results on real images are presented and finally, Chapter 5 concludes the thesis.



## CHAPTER 2

### BACKGROUND ON 3D SURFACE

### RECONSTRUCTION FROM MULTIPLE IMAGES

3D Surface reconstruction by image matching has been a very active research area during the last two decades. Surveys and overviews of image matching techniques can be found in [3], [52], [20], [9], [34], [36] and [79]. Once the cameras are calibrated, that is, the internal parameters of cameras like focal length and principal point, and the camera pose (the position and orientation of the camera in a given common coordinate system) are known, the 3D coordinate of a point can be found, if its projection on at least two images are available. However, finding the corresponding points in images taken from different positions is a difficult task. Even when coordinates of 3D points are estimated, the problem is not completely solved. What we want to obtain as final result

is a description of surfaces in 3D. In the next two sections, we will review the background on image matching and surface reconstruction, respectively.

## 2.1 Image Formation

A two-dimensional (2D) digital image is composed of one or more 2D array of numbers each array corresponding to a color channel. Although there are many different ways in which digital images are formed, we will constrain our scope to images by an optical perspective camera. In such an imaging system, light rays reflected (we assume that surfaces do not emit light) by surfaces pass through an optical system and reach a sensor. The amount of the light observed depends on the incident light (spectrum, incidence angle etc.), the properties of the surface (color, mattness, roughness etc.) and the viewing angle. An accurate model of the optical system may be very complicated [77]. Throughout this thesis, we will assume that the cameras can be modeled by a pin-hole camera (See Figure 2.1). When a point  $P$  with homogeneous coordinates  $(x_p, y_p, z_p, 1)$  in the camera coordinate system is projected on the image plane ( $z = f$  plane where  $f$  is the focal length of the lens), its coordinates

on the image plane are given by

$$s \begin{bmatrix} u_p \\ v_p \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ z_p \\ 1 \end{bmatrix} \quad (2.1)$$

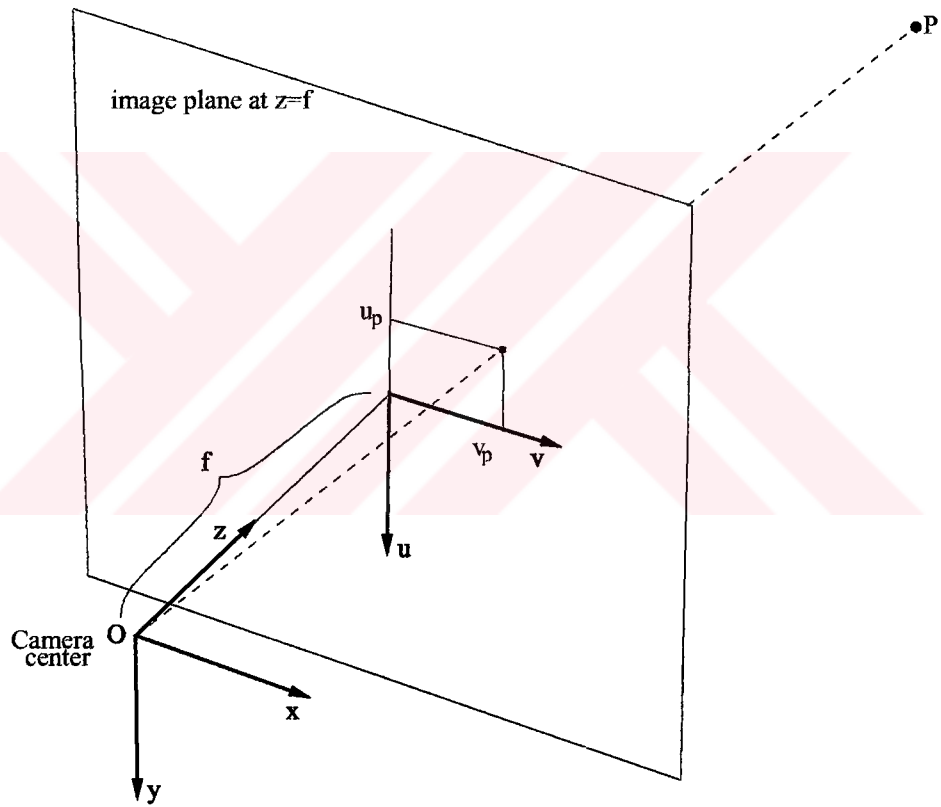


Figure 2.1: Image projection from 3D to the image plane.

The sensors are generally configured on a regular grid aligned with axes  $u$  and  $v$ , so the image coordinates (in terms of pixel row and column numbers)

are given by

$$s \begin{bmatrix} u_p \\ v_p \\ 1 \end{bmatrix} = \begin{bmatrix} f/d_u & 0 & -c_u/d_u & 0 \\ 0 & -f/d_v & -c_v/d_v & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ z_p \\ 1 \end{bmatrix} \quad (2.2)$$

where  $c_u$  and  $c_v$  are the coordinates of the upper-left corner pixel and,  $d_u$  and  $d_v$  are the spacing between the cells of the image sensor in the horizontal and vertical directions, respectively. If the coordinate of the point is expressed in a coordinate system other than the camera coordinate system, then the point should be transformed to the camera coordinate system before projection as

$$\begin{bmatrix} x_p \\ y_p \\ z_p \\ 1 \end{bmatrix} = \begin{bmatrix} R & -RO \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \quad (2.3)$$

where column vector  $O$  is the origin of the camera in the “world” coordinate system and  $R$  is the  $3 \times 3$  rotation matrix defining the rotation between the two coordinate systems.

The sensor, a CCD (charge coupled device) cell, is sensitive to certain frequencies. The amount of the charge integrated in the cell is converted to a digital number by a analog-to-digital converter. During the process, certain amount of electronic noise is added to the signal as well as a quantization noise. Sometimes, the images are taken on photographs and scanned afterwards.



That may add noise, non-linearities and camera extra parameters like an affine transformation from film coordinates to image coordinates.

## 2.2 Camera Calibration

For 3D reconstruction in Euclidean space, the camera positions and orientations (camera pose) in a common coordinate system should be known as well as the internal parameters of the cameras. In some cases, the internal parameters are known in advance. The camera pose can also be known up to a certain extent: The cameras may be fixed in a predetermined configuration (mostly in close-range photogrammetry applications) or the camera pose can be measured during image formation by instruments like tilt sensors, GPS's (Global Positioning Systems) or INS's (Inertial Navigation Systems). In any case, the internal and external parameters can be estimated or fine-tuned after the images are taken. For that purpose, image coordinates of a number of 3D points on two or more images should be known. These coordinates may be marked manually or detected automatically. In some calibration set-ups, the 3D coordinates of test points are known. Let  $\{p_1, \dots, p_m\}$  be the test points and  $S_i = \{s_1, \dots, s_{k_i}\}$  be the set of indices of images that can see the  $P_i$ . If  $f_{ji}$  is the coordinates of  $i$ 'th point marked on the  $j$ 'th image and  $\hat{f}_{ji}$  is the projection of the  $i$ 'th point on the  $j$ 'th image, then  $(f_{ji} - \hat{f}_{ji})$  is the error. In this formulation, the unknowns are a combination of camera internal parameters,

camera pose and coordinates of 3D points. If the number of equations,

$$2 \times \sum_{i=1}^m k_i, \quad (2.4)$$

is greater than the number of unknowns, the system can be solved in the least-square sense. There are many possible formulations of the problem. Some focus on finding the internal parameters of the camera using special targets, while others estimate all the parameters together. For classical bundle adjustment techniques refer to [77] and [83]. Good reviews of relatively new methods based on projective geometry can be found in [32] and [27].

## 2.3 Triangulation

Assume that the internal and external parameters of a camera are known. Then, a point  $P$  is known to be on the line passing through the camera center and the projection of  $P$  in the image. Without additional constraints, it is not possible to determine the 3D position of  $P$ . If a second view of  $P$  is available, then  $P$  should be where the lines converge (See Figure 2.2). This method is called triangulation and can be used to determine the 3D position of a point, with the rare exception of collinear lines.

Of course, because of the noise in measurements, approximations in the camera model and errors in camera calibration, the lines do not intersect but pass very close to each other. The error in the 3D position of the point can be reduced by changing the viewing geometry or by adding new cameras.

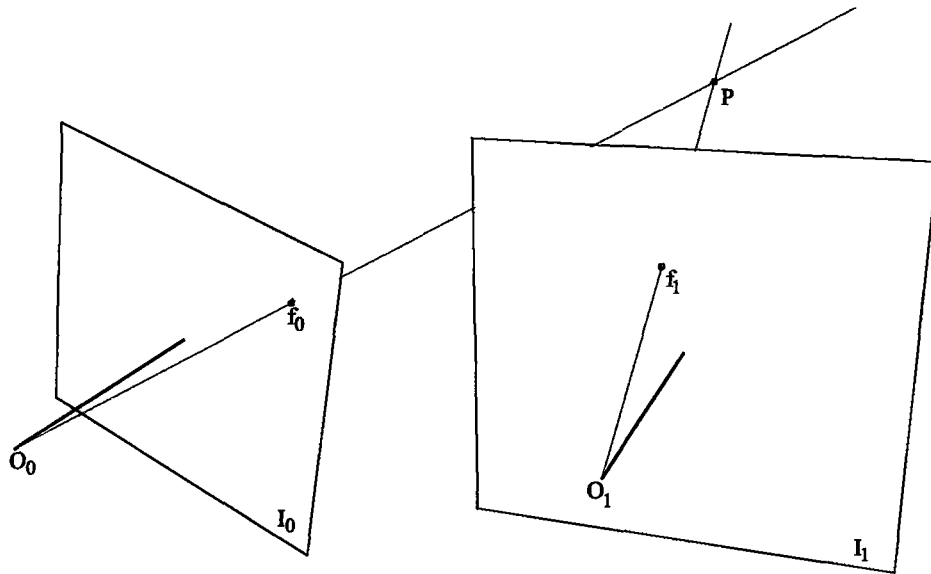


Figure 2.2: Finding the 3D coordinates of a point by triangulation: The point  $P$  is at the intersection of the two rays from two cameras.

For triangulation, the projections of the same point on a 3D object should be identified in each image. Matching corresponding points is a difficult task and the next section is devoted to a brief presentation of background information on the solution of this problem.

## 2.4 Image Matching

Identifying points in multiple images that correspond to the same 3D point is an important phase in many 3D surface reconstruction algorithms. Even when the 3D surfaces are generated without an explicit image matching phase, the surfaces are verified by using their re-projections on the images and comparing the consistency of the corresponding points or features; so the images are

matched implicitly. In the following subsections, some important subject related to image matching will be reviewed.

### 2.4.1 Geometry of Matching

Let  $f_0 = (u_0, v_0)^T$  and  $f_1 = (u_1, v_1)^T$  be the projections of a point  $P$  on image planes  $I_0$  and  $I_1$ , respectively, in a calibrated stereo rig. Then,

$$\begin{bmatrix} u_0 & v_0 & 1 \end{bmatrix} \mathbf{Q} \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} = 0 \quad (2.5)$$

where  $Q$  is called *the essential matrix* [55]. The essential matrix can be derived from the available internal and external parameters or it can be estimated using at least 8 corresponding point pairs [55], [31]. If the internal parameters of the cameras are not known, the matrix  $Q$  can still be estimated, in terms of pixel coordinates instead of image plane coordinates. In that case,  $Q$  is known as the fundamental matrix [56]. If  $f_0$  is known, solving for  $f_1$  gives the equation of a line, which is known as the *epipolar line*. The focal centers of the cameras, the object point and the epipolar lines on both images are all on the same plane (See Figure 2.3).

As a result, the search for a match in the corresponding image reduces to a one-dimensional search. In some cases, the objects are known to be confined in a certain space. So, the search is further restricted to an interval along the epipolar line. When matching two images, the images can be re-projected

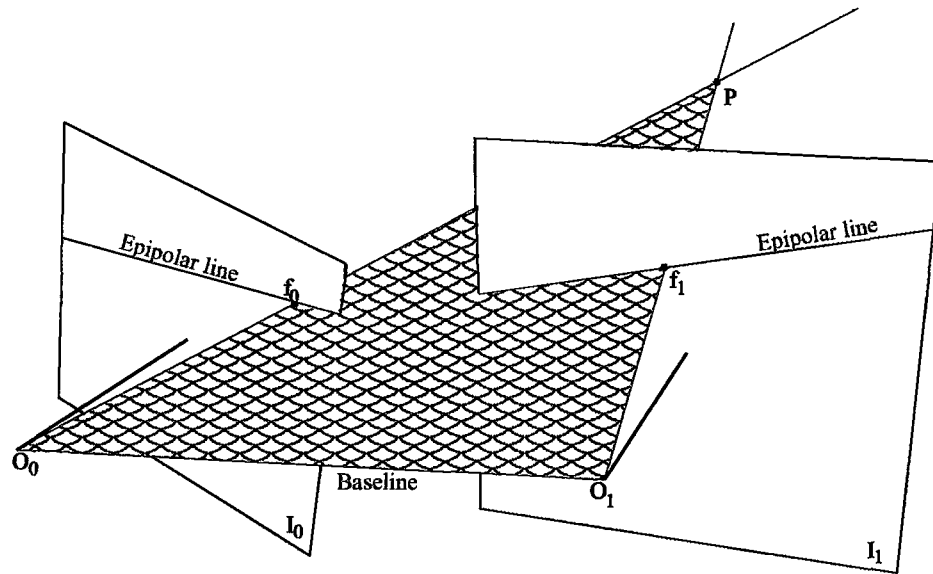


Figure 2.3: Epipolar lines are the intersection of the image planes with the plane formed by the focal centers and the object point.

onto new image planes such that corresponding epipolar lines are collinear with image rows with the same index. Efficient epipolar alignment algorithms can be found in [28] and [31]. Computing those epipolarly aligned images greatly simplifies some matching algorithms. In rectified pairs, the difference between the horizontal coordinates of corresponding points is known as disparity, which is inversely proportional to the distance of the object point to the reference camera. For such a pair, the problem can be solved in terms disparity, resulting in a disparity map, without worrying about the actual 3D coordinates.

When matching three images, geometric constraints can be expressed in terms of a trifocal tensor. But, it is not possible to align more than two images epipolarly except some special camera geometries.

Since this thesis deals with planar approximations, the geometry of a plane viewed by multiple cameras deserves special attention. Let  $f_0 = (u_0, v_0)$  and  $f_1 = (u_1, v_1)$  be the projections of a point  $p$  on a plane  $P_0$  on the image planes  $I_0$  and  $I_1$ , respectively, as shown in the Figure 2.4.

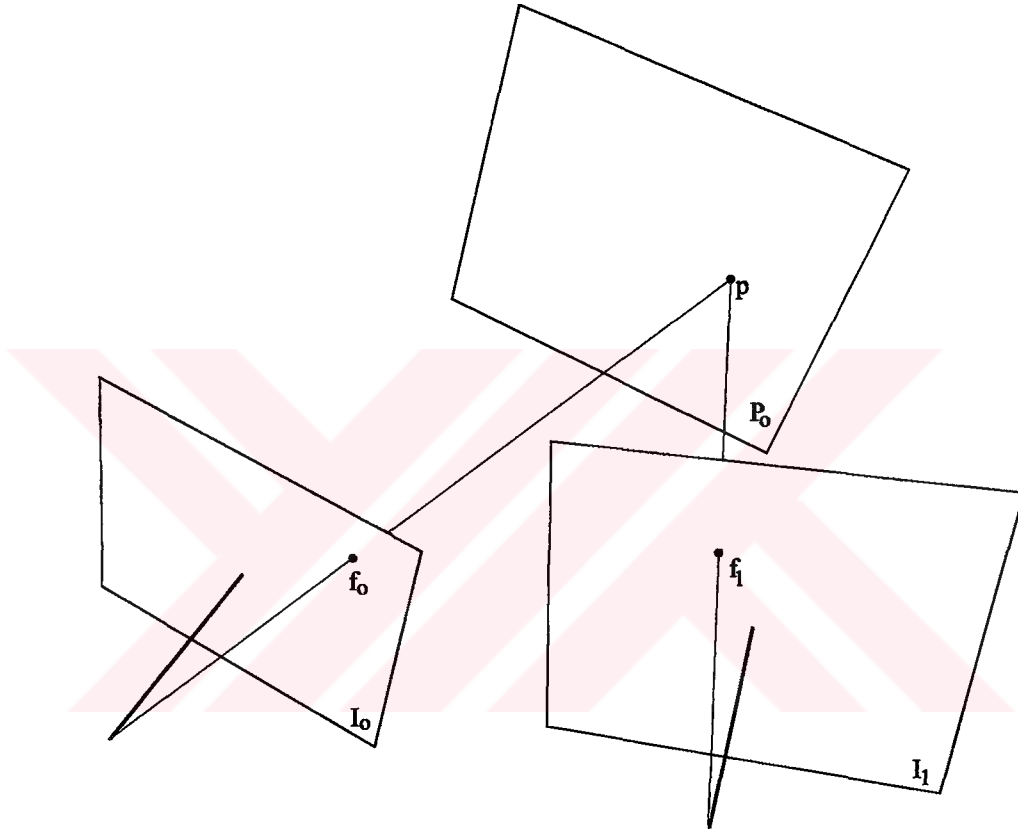


Figure 2.4: A plane viewed by two cameras. The coordinates of  $f_0$  and  $f_1$  are related by a homography.

Then the relation between  $f_0$  and  $f_1$  is given by

$$\begin{bmatrix} u_0 \\ v_0 \\ 1 \end{bmatrix} = \mathbf{H} \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} \quad (2.6)$$

where the  $3 \times 3$  matrix  $\mathbf{H}$  is called a planar homography and can be derived from the camera geometry and the parameters of the plane (See, for example, [22] for calculation of the homography).

## 2.4.2 Problems

Matching images point-by-point is an ill-posed problem and can be solved by imposing some sort of regularization. Some important sources of ambiguity in image matching problems are noise, lack of texture or repetitive texture. Physical changes in the scene as well as illumination differences between the times images are taken, make the problem much more difficult. Occluded areas, that is, areas that cannot be seen in some images but can be seen in others, are another important source of error.

## 2.4.3 Constraints

Although the matching geometry constrains the search space, there are still many possible solutions. Researchers introduce other constraints to be able to solve the correspondence problem like: Smoothness, orderedness, uniqueness and compatibility.

Smoothness constraint is based on the fact that scenes are, in general, composed of smoothly varying surfaces. So, the solution, e.g. the disparity map, is forced to be a smooth function. Since this regularization causes problems

around depth discontinuities, it may be relaxed allowing discontinuities. In that case, solution is piecewise continuous.

Orderedness constraint makes sense in matching of epipolarly corrected image pairs. The horizontal order of the points in one image should be preserved in the corresponding points in the other image. There are cases where this constraint is violated in real images.

Compatibility constraint requires that if a point  $p_0$  matches point  $p_1$ , then point  $p_1$  also matches  $p_0$ . This constraint makes sense if the disparity maps for left and right images are found independently.

#### **2.4.4 Primitives**

Algorithms for matching points across two or more images are roughly divided into two categories as feature-based and area-based. Feature-based algorithms try to match detected features like corners, line segments, arcs or regions. Area-based algorithms, on the other hand, try to match images point-by-point. Each approach has its own pros and cons. Our algorithm tries to benefit from the advantages of both worlds. So, the next two subsections are devoted to feature-based and area-based matching primitives, respectively.

##### **2.4.4.1 Features**

Since, the feature matching step of our algorithm is based on linear line segments, we will focus on the similarity measures for features. The attributes



used for line segment matching in literature are overlap length [89], [58]; orientation differences [58], overlap ratio, length comparison [50]; integration of perpendicular distance along the line segments [4]; contrast difference [58]; cross-correlation of the segment neighborhood [2], [73], [71]; gradient direction [73] and chromatic statistics in the neighborhood [70], [66], [7]. We have used orientation difference and distance between segments in addition to evidence from 3D surface model.

#### 2.4.4.2 Area-Based Primitives

Ideally, a point observed by several cameras should lead to same measurement in all the images. But, in reality, this is not the case. Noise in the image formation process, surface reflectance properties, illumination differences are the main reasons for the differences in the measurements. Since area-based methods try to match each point in the image, they are sensitive to such differences. Next two subsections elaborate on the solutions to these problems. Then, similarity measures for area matching and color image matching are discussed. Finally, one of the methods for area-based matching, which is region growing on disparity maps, is reviewed because of its close relation to our algorithm.

##### 2.4.4.2.1 *Surface Reflectance Model*

The simplest and the most frequently used assumption about the surface properties is Lambertian and opaqueness assumptions. A Lambertian surface

is such that it reflects the same light in all directions. These assumptions are true for most of the surfaces, but algorithms based on these assumptions cannot cope with semi-transparent or specular surfaces. There are some attempts to model non-Lambertian surfaces: Pedersini et al. [67] assume partially matte surfaces illuminated by a dominant light source at infinity and a small amount of diffuse light. Bhat and Nayar [6] deal with specular reflection from rough surfaces. Schultz [72] uses a specific model for water to reconstruct the ocean surface from multiple images.

#### 2.4.4.2.2 Photometric Distortion

Photometric differences among images of the same point may be the result of either differences in the imaging system or of illumination differences. A model frequently used is

$$I_0(\mathbf{x}_0) = a I_1(\mathbf{x}_1) + b \quad (2.7)$$

where  $\mathbf{x}_0$  and  $\mathbf{x}_1$  are the projections of the same point to the images  $I_0$  and  $I_1$ , respectively. Here,  $a$  is camera offset and  $b$  is the camera gain. Bignon [7] estimates the offset and the gain using means of several corresponding areas selected manually. The images to be matched are photometrically equalized to a chosen reference image before the image matching phase. Rosenholm [69] estimates the two parameters and disparities simultaneously using the least-squares technique. Kim and Muller [44] model only a photometric shift term

(offset). Pedersini et al. [67] model both offset and gain. Silva and Santos-Victor [74] remove the effect of  $b$  by preprocessing the images using a zero mean filter. An alternative of explicitly modeling the photometric distortion is using similarity metrics that are not sensitive to certain kinds of distortions. Next section reviews the similarity measures used for area matching.

#### 2.4.4.2.3 Two-Image Correlation-Like Measures

The usual way to obtain dense matches in stereo pairs is calculating correlation (normalized zero-mean cross-correlation in general) over a neighborhood of the points to be matched. If  $I_0$  and  $I_1$  are small (usually rectangular) 2D areas to be matched, then cross-correlation and normalized zero-mean cross-correlation are defined as

$$C_1 = (I_0 - \bar{I}_0) \cdot (I_1 - \bar{I}_1) \quad (2.8)$$

and

$$C_2 = \frac{((I_0 - \bar{I}_0) \cdot (I_1 - \bar{I}_1))}{\sqrt{(I_0 - \bar{I}_0) \cdot (I_0 - \bar{I}_0)} \sqrt{(I_1 - \bar{I}_1) \cdot (I_1 - \bar{I}_1)}}, \quad (2.9)$$

respectively, where  $\bar{I}_0$  and  $\bar{I}_1$  are the mean values of the signals and  $(\cdot)$  denotes the inner product. Subtracting the mean removes the effect of any offset differences and normalization compensate for any gain differences.

Another measure commonly used for area-based matching is the sum of squared differences (SSD) [63], [43] given by

$$Er_1 = var(I_0 - I_i). \quad (2.10)$$

When normalized properly, the maximization of the correlation score is equivalent to the minimization of the SSD [57].

Kuglin and Hines [47] introduced a correlation measure based on the phases of band-pass filters. Zyka and Sara [91] use rank correlation while Bhat and Nayar [5] use ordinal measures. Both of these measures are insensitive to monotonic transformations among the images. A few other measures defined for two signals can be found in [57].

#### 2.4.4.2.4 Multi-Image Correlation-Like Measures

When matching multiple images, correlation cannot be generalized directly. One solution is to choose a reference camera, calculating the correlations pairwise, and summing the results up [2], [10]. That is, if the correlation of a patch  $I_0$  from the reference camera and its corresponding patch  $I_i$  in the  $i$ th camera assuming a surface at the distance  $d$  from the reference camera is  $C_i(d)$ , then the multi-image correlation is given by

$$C_3(d) = \frac{1}{n-1} \sum_{i=1}^{n-1} C_i(d). \quad (2.11)$$

Paparoditis and Maillet [65] defined a multi-image correlation for directly measuring the similarity of  $n$  signals,  $I_0, \dots, I_{n-1}$  as

$$C_4 = \frac{\text{var} \left( \sum_{i=0}^{n-1} I_i \right)}{\sum_{i=0}^{n-1} \text{var} (I_i)}. \quad (2.12)$$

Note that the correlation given by Equation 2.12 can be calculated without defining a reference camera. So, a search for the maximum of the correlation

can be done along any line in the object space and the performance of the search does not depend on arbitrary choice of a reference camera.

An interesting multi-image similarity score is proposed by Collins [16]. He calculates the image gradient directions and back-projects them on a plane in the object space. If the gradient directions from  $n$  cameras are  $\theta_0, \dots, \theta_{n-1}$ , then the similarity is

$$C_5 = \sqrt{\left(\sum_{i=0}^{n-1} \cos \theta_i + \sum_{i=0}^{n-1} \sin \theta_i\right)^2} \quad (2.13)$$

Okutomi and Kanade [63] uses SSD for matching multiple images taken on the same baseline by choosing a reference camera and combining SSDs obtained pairwise as

$$Er_2 = \sum_{i=1}^{n-1} w_i \text{var}(I_0 - I_i) \quad (2.14)$$

where the weights  $w_i$  are inversely proportional to the length of the baseline between pair of images and all  $I_i$ 's are found assuming the same distance from the reference camera. Jia et al. [41] apply the same algorithm for two orthogonal baselines. Narayanan et al. [61] and Mellor [59] use the same metric in a more general camera configuration benefiting from epipolar geometry. Wang and Ohnishi [86] also employ the same measure for finding the position and orientation of small rectangular patches in 3D.

Tsai [84] has defined a similar metric that does not need a reference camera, window variance, as

$$Er_3 = \sum_{i=0}^{n-1} w_i \text{var}(I_i - \bar{I}) \quad (2.15)$$

where

$$\bar{I}_i = \frac{1}{n} \sum_{i=0}^{n-1} I_i. \quad (2.16)$$

This metric is also used by Heleva [35] and Fua [24]. Tsai has proposed another metric, joint moment, which is defined as follows:

$$Er_4 = \frac{E \left\{ \prod_{i=0}^{n-1} I_i - \bar{I} \right\}}{\prod_{i=0}^{n-1} E \left\{ \left\{ E[I_i - \bar{I}]^n \right\}^{1/n} \right\}} \quad (2.17)$$

#### 2.4.4.2.5 Matching Color Images

The correlation-like methods are defined on gray level images. When using color images, the information from each channel should be combined. The most straightforward way is taking the average of correlations obtained from each color channel. Mellor [59] uses the dissimilarity measure

$$Er_5 = \frac{E_R}{\sigma_r^2} + \frac{E_G}{\sigma_g^2} + \frac{E_B}{\sigma_b^2} \quad (2.18)$$

where  $E_R, E_G, E_B$  are dissimilarity values obtained by SSD and  $\sigma_r^2, \sigma_g^2, \sigma_b^2$  are estimated noise variances in red, green and blue channels, respectively. This is, in fact, the Mahalanobis distance under the assumption of independent noise across channels. Girard et al. [29] use the median of correlation values instead of averages. Jordan and Bovik [42] employ a constraint called chromatic gradient matching constraint defined on each channel. If the constraint is satisfied in all channels, they accept the match. That is, they use a *logical and* function for combining color channels. Similarly, Park and Zimmermann

[66] reject point matches if any of the correlation coefficients calculated for different channels is below a threshold.

## 2.4.5 Strategies

In the computer vision literature, there are many different formulations of the matching problem as well as strategies for solving it. The problem can be expressed as a cost function and energy-minimization methods are employed. Hierarchical implementations that work on several resolution levels reach better solutions in less time. Relaxation schemes use relations among matches to reduce the ambiguity progressively.

The main strategy used in this thesis is to begin from a point where a partial solution is already obtained, and solving for the problem in the neighborhood using the partial solution as a constraint. An approach in the same vein is region growing on disparity maps which is briefly reviewed in the next subsection.

### 2.4.5.1 Region Growing on Disparity Maps

The region growing approaches of Otto and Chau [64], Lhuillier and Quan [53], D'apuzzo [17], and Skrinjar et al. [75], which use match propagation on a disparity map from seed matches, are also similar to our surface growing algorithm. However, in region growing on disparity maps, it is not possible to use more than two images simultaneously. Besides, one cannot benefit from

an explicit surface model.

## 2.5 3D Surface Reconstruction from Multiple Images

3D point clouds or disparity maps obtained by image matching are intermediate results and should be integrated into a 3D surface model. In this section, methods that manipulate or create surfaces in the object space are reviewed.

### 2.5.1 Surface Fitting to Unorganized Data Points

The algorithms for fitting surfaces to 3D points can be divided into two categories. In the first approach, the points are organized, that is, besides the coordinates of the points, connectivity information is also available for each view. There is a rich literature towards the integration of organized points from multiple sources, not necessarily from stereo, but also from laser range cameras. See Hilton et al. [38], [39] for a comparative analysis of several such methods.

In the second approach, surfaces are fitted to an unorganized point cloud in the object space. Liao and Medioni [54] deform a given initial 3D mesh by energy minimization. Chen and Medioni [13] use an iterative inflating balloon algorithm to model closed surfaces. Fradkin et al. [23] and Canu et al. [11] fit planar surface patches to 3D points defined by segmentation of color



images and by segmentation of digital elevation models, respectively. Koch [46] clusters points into regions of similar surface orientation. Fua [24] uses iterative re-weighted least squares, while Šára and Bajcsy [85] and Hoppe et al. [40] use eigenvalue analysis to estimate surface primitives that represent a group of 3D points in a voxel. In Fua’s work, the surface primitives are small disks in 3D called “oriented particles”, introduced by Szeliski and Tonnesen [81]. In [85], the local geometric primitives called “fish scales” are expressed as fuzzy sets. Leloğlu and Halıcı [51] fit growing planar surfaces to 3D points. In [40], a function whose zero-set represents the surface, is used. In all three cases, these primitives are grouped to form surfaces, using various techniques. But all these algorithms use 3D points to form surfaces.

### **2.5.2 Minimizing Re-Projection Error by Optimization**

In our approach, the surfaces in 3D are obtained with no need to a prior image matching phase. A method in this vein is maximization of multi-image correlation by changing parameters of a surface. Havaladar and Medioni [33] use generalized cylinders for this purpose that are hypothesized using groups of edges. But, their algorithm is restricted to three images taken from close points. Fua and Leclerc [25] and Zhang and Seitz [88] also minimize an objective function defined over multiple images by moving the vertexes of a triangular mesh, but the algorithm begins from an initial surface obtained by some other method.

Similarly, Faugeras and Keriven [22] use a surface evolution scheme that can deal with multiple objects. The surface evolution is driven by partial differential equations implemented using level set methods. It is not shown if the evolving surfaces converge to the solution on complicated environments with frequent discontinuities.

### **2.5.3 Volumetric Methods**

Some volumetric methods are also proposed to obtain surfaces without explicit matching. In the space carving method [48], part of the space that contains the objects is divided into voxels. The outermost voxels of the volume are removed if they are not justified by the images. When the volume does not shrink any more, a surface covering the volume is generated. Surveys of volumetric methods can be found in [76] and [21]. In those methods, there is no explicit surface model until the volume is generated. So, it is difficult to handle non-Lambertian surfaces and constraints on surface shape cannot be imposed.

### **2.5.4 Generating Surfaces in 3D**

An algorithm that generates surface patches in 3D from a stereo pair without a prior matching phase is proposed by Super and Klarquist [78]. A reference camera is chosen and the maximum of a correlation value obtained from a rectangular 3D patch is found as the patch is moved away from the reference

image and as its orientation is changed. The algorithm is developed for stereo pairs and is tested on only highly textured image pairs. The result is not a surface but a collection of surface patches. Similarly, Paparoditis and Maillet [65] calculate a multi-image correlation score as they scan the height range over a regular grid to obtain digital surface models from multiple aerial images. Wang and Ohnishi [87] segment a reference image and fit a planar surface patch in 3D for each 2D segment in the reference image, using a multi-image correlation. The performance of this surface reconstruction algorithm is limited with the performance of the segmentation algorithm. Besides, they create a reference camera.

An algorithm that generate models directly from images is developed by Kiriwara and Saito [45]. Triangular and rectangular prisms inferred in 3D are evaluated by using multiple image correlation and by comparing projected models to image gradients. Best combination of models and best parameters for these models are searched by employing genetic algorithms. Although the results are promising, models are restricted to simple building blocks and the method is very costly in terms of computation.

A surface growing approach is proposed by Mellor [59] for reconstructing buildings from multiple images. He divides the space into voxels and hypothesizes rectangular surface patches in each one. Each surface patch is tested using image correlation over multiple images while the surface is rotated to

various angles. After grouping surface patches, the surface grows by hypothesizing surface patches in neighboring voxels. Since large patches are used, fine detail can not be handled. Mellor's algorithm does not benefit from a parametric surface model that poses additional useful constraints in a polyhedral world. Schultz [72] also proposed a surface growing algorithm for almost planar ocean surface reconstruction. Using a model of water surface reflection, the surface orientation is determined. Knowing the 3D position of a seed and its surface orientation, positions of nearby points are calculated by a locally planar surface assumption. This method is tested on only smooth surfaces.

Another related algorithm is proposed by Baillard et al. [2]. Surface reconstruction begins from 3D lines. Half planes are inferred at one side of the 3D lines. The best angle of the plane that can rotate around the line is determined by maximization of a kind of multiple-image correlation. Then, half planes are merged and their boundaries are detected as intersections of planes. The planes are inferred directly in the object space, but the algorithm requires 3D lines initially and it neglects the details.

## 2.6 Error Assessment

Various types of 3D elements are generated by image matching algorithms including 3D points, 3D line segments, implicit surfaces, parametric surfaces, triangular meshes and geometric models. A quality assessment of the results

should consider two types of quality metrics: Reliability and accuracy.

Reliability [69] measures the ratio of correctly matched tokens in all matches. In that sense, a match is either true or false. The reliability of matched features can be evaluated by counting the false matches identified by inspection (See [62] and [14] for examples of point features and [71] for an example of linear features). On the other hand, accuracy is the measure of how close are the 3D elements to the “true” 3D features or surfaces. If false matches are not treated in a different category, they appear as outliers in the error statistics.

The error assessment can be done in two ways. The first approach is comparing the results to a ground truth generated by other means. The ground truth may be measurements of 3D points [69], [1]; 3D polygons [82]; digital elevation models as well as disparity maps [8], [68], [80], [19].

The second approach is measuring the quality of the results using their self-consistency. An earlier example is the work of Mohan et al. [60] where the reliability of an algorithm that match individual edgel points is measured. The edgels violating figural continuity constraint, i.e., neighboring edgels with large disparity differences are detected as false matches. Leclerc et al. [49] compare the coordinates of the same 3D points obtained from different image pairs. Szeliski and Zabih [80] use the predictive power of the 3D reconstruction as a quality metric. They compare a novel view to the image generated synthetically by a virtual camera with the same pose and the same parameters.

There are also attempts towards estimating the reliability of each match during matching phase. For example, Joinville et al. [18] use the shape of the correlation as a function of disparity to derive a reliability map.



## CHAPTER 3

### 3D RECONSTRUCTION BY SURFACE

#### GROWING

The essence of the proposed surface reconstruction paradigm is the inference of surfaces in 3D without explicit image matching and verification of the inferred surfaces using the images. The basic idea is to grow a surface in 3D, beginning from a small surface patch that serves as a seed. Since man-made environments are composed of mostly planar opaque surfaces, the surface model used is planar.

The flow diagram of the algorithm is depicted in Figure 3.1. The algorithm begins with hypothesis (seed) generation. Then the surface grows in the object space while offsets for each channel of each camera are estimated. When the region cannot grow any more, features are matched across views using the surface. If the surface is verified by matching features too, it is merged with

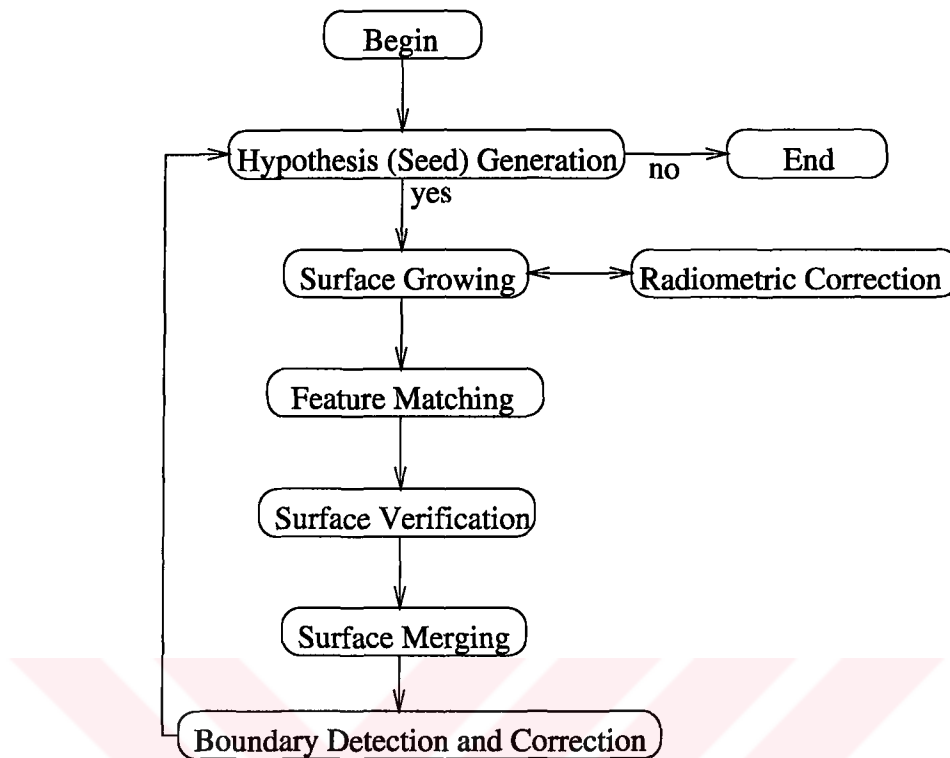


Figure 3.1: Flow diagram of the main algorithm.

coplanar neighbors. Following the boundary detection for the region, the steps of the algorithm is repeated with another seed as long as new seeds can be generated. Each of the steps of the algorithm is explained in the following subsections.

### 3.1 Surface Representation

For the surface growing algorithm, a surface representation scheme, that can represent any differentiable surface patch with arbitrary boundary and with any number of holes, is necessary. For that purpose, a surface in explicit parametric form and a discrete surface coordinate system can be used. Each



point in the discrete coordinate system represents a surfel. We have chosen a planar form (See Figure 3.2) for the sake of simplicity.

We define *planar surface*  $S$  as a 4-tuple  $S = \{\mathbf{O}, \mathbf{u}, \mathbf{v}, R\}$ .  $\mathbf{O} \in \mathbf{R}^3$  is the center and  $\mathbf{u}, \mathbf{v} \in \mathbf{R}^3$  are unit vectors of the surface coordinate system. The unit vectors  $\mathbf{u}, \mathbf{v}$  and the surface normal  $\mathbf{n} = \mathbf{u} \times \mathbf{v}$  form an orthonormal basis for  $\mathbf{R}^3$ . The surface is composed of the set of surfels,  $R = \{r_1, \dots, r_N\}$  such that  $r_i \in \mathbf{Z} \times \mathbf{Z} \times \mathbf{Cs}$ ,  $1 \leq i \leq N$ , where  $\mathbf{Cs}$  is any color space and the corresponding component represents the surfel color.

Center coordinates of the surfel  $r_i = \{a_i, b_i, \mathbf{c}_i\}$  in the object space is

$$C(r_i) = \mathbf{O} + s a_i \mathbf{u} + s b_i \mathbf{v} \quad (3.1)$$

where  $s \in \mathbf{R}$  is the surfel size and it is chosen such that the projection a surfel in the images is less than a pixel on the average.

Although the surface model is planar, it can be extended to higher order surfaces with small modifications. Note that the model also represent the texture of the surface. After the reconstruction, the 3D model is available without any further processing.

## 3.2 Hypothesis Generation

Surface growing begins with a seed which is a small planar patch made of small number of surfels. We have used a rectangular patch with a size of  $7 \times 7$  surfels.

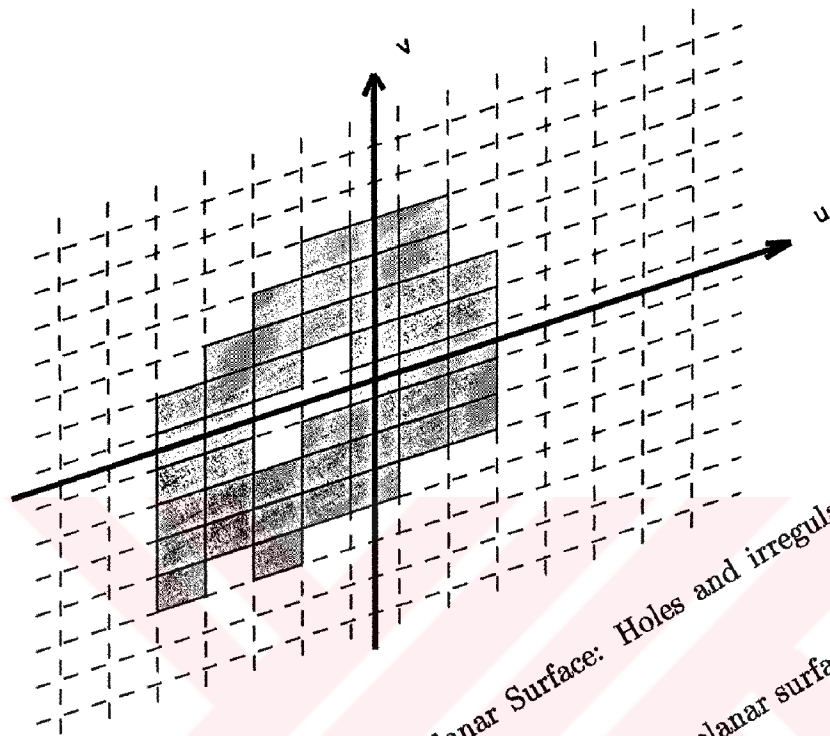


Figure 3.2: An Example Planar Surface: Holes and irregular shape can be represented.

This seed is a hypothesis that there exists a planar surface in the object space containing the seed.

If the hypothesis is false, that is, the seed does not correspond to a real object surface, there are two possibilities: *i*) It may not grow at all due to high re-projection error. *ii*) The region may grow because of lack of texture, but it is not verified by matched features. In those cases the hypothesis is falsified and the planar surface is removed.

Spurious surfaces, i.e., surfaces that happen to grow from false seeds, slow down the convergence of the algorithm considerably. So, an effective seed generation mechanism is necessary. Two different approaches can be used in

finding seed points: *i*) Bottom-up processes (using low-level raw information) like using the result of a search for the maximum of some multi-image correlation function, or *ii*) Top-down processes (using information from a more abstract level) like using already reconstructed surfaces in the neighborhood. We use both mechanisms simultaneously.

As bottom-up process, a small rectangular surface is generated and an initial point and orientation is searched that will minimize the re-projection error and maximize the feature match function that are described in the following sections. Since this is a quite involved search in three dimensions, a heuristic is used: First, an image is chosen at random. Since areas with strong texture are good candidates, interest points, i.e., extrema of the pixel values in each channel of the smoothed image, are detected as candidate seed points. The points near or on a line segment are removed, because such points are likely to correspond edges where two surfaces intersect (See Figure 3.3 for an example of interest point detection). A small frontoparallel (parallel to the image plane) planar surface patch is hypothesized on the line that connects the camera center and a random candidate. The re-projection error associated to that planar surface (See Subsection 3.5) is minimized as the seed surface is moved along the ray. Once the center point is fixed in the space, best orientation is searched using the feature match function, which is a measure of how well the features are matched (See Subsection 3.4), with a larger surface.

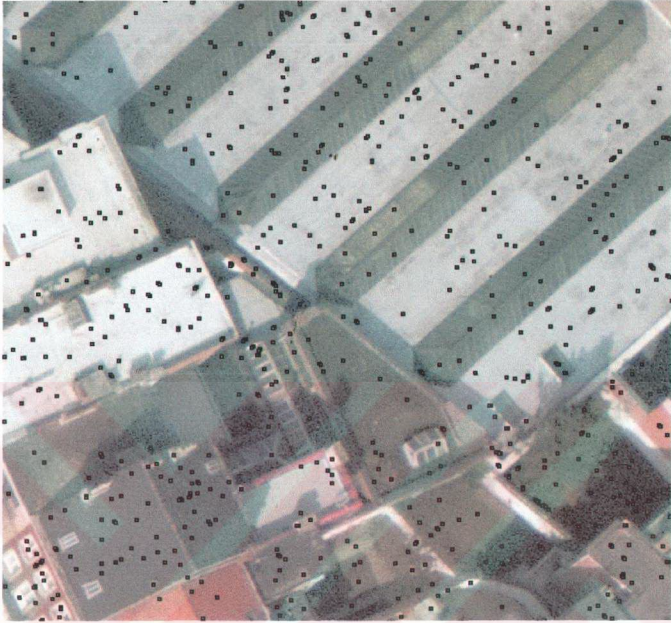


Figure 3.3: Interest points detected on an aerial image.

As top-down process, the 3D lines detected as the border of planar surfaces are used. In that case, a rectangular surface patch along the unoccupied side of the 3D line is inferred. As a result, the search for the seed surface patch reduces to a one-dimensional search as in the work of Baillard et al. [2].

### 3.3 Re-Projection Error and Radiometric Correction

Let  $P_k : \mathbf{R}^3 \rightarrow \mathbf{R}^2$  be the projection function that maps 3D points in absolute coordinates to the image coordinates of  $k$ 'th image,  $\mathbf{I}_k : \mathbf{R}^2 \rightarrow \mathbf{Cs}$ . Then, the color of the surfel  $r_{ji}$ , which is the  $i$ 'th surfel of the planar surface  $S_j$ , observed by the  $k$ 'th camera is given by

$$\mathbf{c}_{ji}^k = \mathbf{I}_k(P_k(C(r_{ji}))). \quad (3.2)$$

Let the color of  $r_{ji}$  be  $\mathbf{c}_{ji}$ .  $\mathbf{c}_{ji}^k$ 's will not be equal to  $\mathbf{c}_{ji}$  because of the differences of camera sensitivities and offsets, varying lighting conditions, non-Lambertian surfaces and noise. The tests that we have conducted on real images revealed that (see Chapter 4) the differences across images can be successfully modeled by Gaussian noise and an additive constant  $\mathbf{d}_j^k$  for  $k$ 'th camera observing  $j$ 'th surface. So,

$$\mathbf{c}_{ji}^k = \mathbf{c}_{ji} + \mathbf{d}_j^k + \tilde{\mathbf{g}} \quad (3.3)$$

where  $\tilde{\mathbf{g}}$  is a vector composed of independent zero mean Gaussian random variables with variance  $\sigma^2$ . A more general model including camera sensitivities can also be used, but it does not lead to better results with the images we have used for testing the algorithm. Assuming that a surfel  $r_{ji}$  is visible by all  $n$  cameras, the color of the surfel is estimated as

$$\hat{\mathbf{c}}_{ji} = \frac{1}{n} \sum_{k=1}^n \mathbf{c}_{ji}^k. \quad (3.4)$$

Then, the additive constants are found by the equation:

$$\hat{\mathbf{d}}_j^k = \frac{1}{N_j} \sum_{i=1}^{N_j} \mathbf{c}_{ji}^k - \hat{\mathbf{c}}_{ji}, \quad (3.5)$$

The mean square error  $e_{ji}$  associated to the surfel  $r_{ji}$  is,

$$e_{ji} = \frac{1}{n-1} \sum_{k=1}^n \|\mathbf{c}_{ji}^k - (\hat{\mathbf{c}}_{ji} + \hat{\mathbf{d}}_j^k)\|^2 \quad (3.6)$$

and the noise variance related to surface  $S_j$  is estimated as

$$\hat{\sigma}_j^2 = \frac{1}{N_j} \sum_{i=1}^{N_j} e_{ji}. \quad (3.7)$$

The idea of augmenting the RGB channels by the derivative magnitude images to increase the robustness of the system is used successfully by Gouet at al. [30]. Similarly, a second error term  $e'_{ji}$  is also calculated using gradient magnitude images obtained by Sobel operator, instead of color images. In that case, the additive constants are not estimated.

### 3.4 Feature Match Function

In this section, a feature match function is defined, which is a measure of how well features from all cameras match when projected onto the planar surface.

A 2D line segment  $l$  is represented by its terminal points  $(x_0, y_0; x_1, y_1)$  whose angle and length are given by

$$\theta = \tan \left( \frac{y_1 - y_0}{x_1 - x_0} \right) \quad (3.8)$$

and

$$\text{len}(l) = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}, \quad (3.9)$$

respectively. We define the similarity of two line segments  $S(l_i, l_j)$  as

$$S(l_i, l_j) = e^{-\frac{\Delta\theta^2}{t_\theta^2}} e^{-\frac{d_{ij}^2}{t_d^2}} \sqrt{\text{len}(l_i)\text{len}(l_j)} \quad (3.10)$$

where

$$\Delta\theta = \begin{cases} |\theta_i - \theta_j|_{\text{mod}=\pi}, & |\theta_i - \theta_j|_{\text{mod}=\pi} < \pi/2 \\ \pi - |\theta_i - \theta_j|_{\text{mod}=\pi}, & \text{else} \end{cases}. \quad (3.11)$$

The similarity measure is based on three quantities: Orientation difference, distance between the segments and segment lengths. The distance  $d_{ij}$  is the minimum of the distance of the closest point of  $l_j$  to the midpoint of  $l_i$  and the distance of the closest point of  $l_i$  to the midpoint of  $l_j$  (See Figure 3.4). In Equation 3.10, we have used the Gaussian function instead of hard thresholds to have a smooth and differentiable similarity function. Since the maximum error of the line segment detector is set to the default value of 1, the distance sensitivity parameter  $t_d$  is set to 3.1 so that Gaussian function decays to %90 at 1 pixel distance. The angle sensitivity parameter  $t_\theta$  is set to 15.4 which corresponds to %90 at 5 degrees.

To calculate the feature match function of a planar surface, it is re-projected on the images and the 2D line segments that touch to the image of the planar surface are selected. If the projections of all selected line segments in the  $p$ 'th image onto the plane defined by the planar surface are  $L_p = \{l_p^1, \dots, l_p^{n_p}\}$ , then

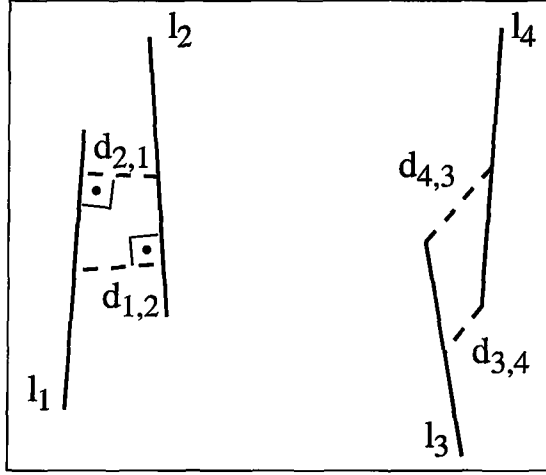


Figure 3.4: The distance from a line segment to another one is defined as the shortest distance from the midpoint of the first to the second.

the feature match function is given by

$$F = \sum_{p=1}^{n-1} \sum_{q=p+1}^n \sum_{i=1}^{n_p} \sum_{j=1}^{n_q} S(l_p^i, l_q^j). \quad (3.12)$$

### 3.5 Surface Growing

Let  $S = \{\mathbf{O}, \mathbf{u}, \mathbf{v}, R\}$  be the planar surface on which surface growing is applied.

At each iteration, set of neighboring surfels  $R_N$  of the set  $R = \{r_1, \dots, r_N\}$ , is found, such that

$$R_N = \{r_{Nl} | r_{Nl} \notin R \text{ and } \exists r_i \in R, d(r_{Nl}, r_i) = s\} \quad (3.13)$$

where  $d(\cdot)$  is the Euclidean distance of two surfel centers. The elements of this set are first checked to be sure that their projections on the images do not coincide with the projection of any other accepted surfel. The coinciding surfels, i.e., the surfels violating the uniqueness constraint, are removed from



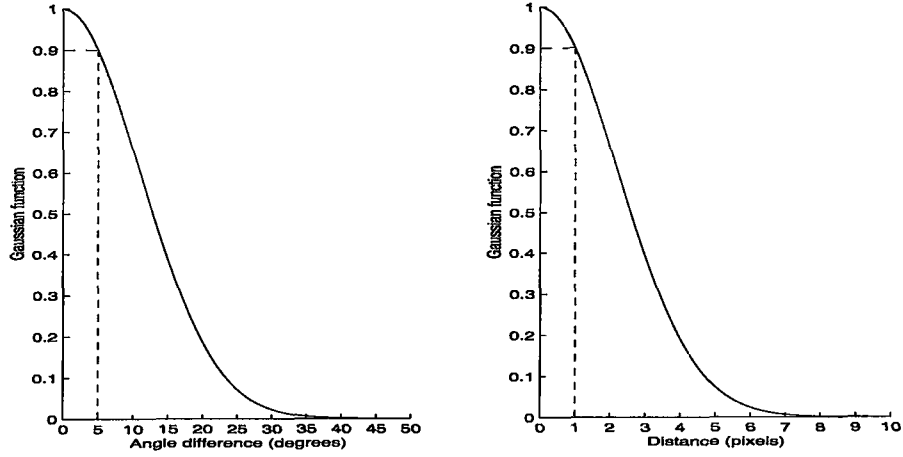


Figure 3.5: The Gaussian functions used in the line segment similarity function. *Left:* Gaussian function for angle difference between line segments. *Right:* Gaussian function for the distance between line segments.

the set. For surface growing, the errors  $e_{ji}$  and  $e'_{ji}$  are evaluated for each element of  $R_N$ . Then, neighboring surfels, satisfying the conditions  $e_{ji} < 1.65 \min(\hat{\sigma}_j, e_{th})$  (1.65 standard deviations cover 90% of the samples of a Gaussian distribution) and  $e'_{ji} < e'_{th}$ , are transferred to the set  $R$  where  $e_{th}$  is an upper bound for the threshold and  $\hat{\sigma}_j$  is the standard deviation of noise, i.e., the square root of the noise variance given by Equation 3.7. The threshold  $e'_{th}$  is determined experimentally, because, the distribution of the noise in the gradient magnitude images is very complicated and deriving a threshold from the noise variance is difficult.  $\hat{c}_{ji}$  is accepted as the color of the surfel. Surfels whose 4-neighbours are in the set are also added to the surface without checking the corresponding image variance.

Since the initial surface parameters of a seed may not be accurate due to small support area, the parameters need to be updated to minimize the total

re-projection error

$$e_j = \sum_{i=1}^{N_j} e_{ji} \quad (3.14)$$

or the feature match function, or a linear combination of the two, during surface growing. Among these alternatives, we have chosen the feature match function, because the computation time is much shorter when compared to the total re-projection error for typical images. For fine tuning the planar surface, each time the area of the surface doubles, the error function is minimized using steepest descent with respect to three parameters: two for orientation and one for movement along the surface normal.

## 3.6 Feature Matching

Note that, although feature match function is a measure of how well features match, it does not match the line segments explicitly. At this phase, the 2-D features detected on the images that touch to the projection of the planar surface on images are matched. This explicit feature matching is necessary for surface verification, surface boundary detection and top-down seed generation. We have used linear line segments as 2-D features. The line segments on the images are back-projected on the plane defined by the planar surface. If the 2D line that are projected on the planar surface from each camera have approximately the same orientation ( $\pm 5^\circ$ ) and the distance between them is less than maximum error allowed by the segment detector, they are matched.

When 2D line segments  $l_1, \dots, l_n$  from  $n$  cameras are matched, they are merged into a single 2D line segment that passes through the gravity center of the segment

$$\frac{1}{\sum_{i=1}^n \text{len}(l_i)} \sum_{i=1}^n \text{len}(l_i) \left( (x_0^i + x_1^i)/2, (y_0^i + y_1^i)/2 \right) \quad (3.15)$$

and that is parallel to the sum the 2D vectors

$$\sum_{i=1}^n (x_1 - x_0, y_1 - y_0). \quad (3.16)$$

To determine the extent of the merged line segment along the 2D infinite line, the projections of all line segments on the baseline are found as illustrated in Figure 3.7. The part of the line where at least two line segments correspond is taken as the merged line. After the line segments are merged, small gaps between collinear lines are completed by adding line segments between the terminal points and the line segments that intersect at a close distance are elongated until the intersection point.

### 3.7 Surface Verification

Although an error correction mechanism replaces spurious surfaces with real ones, most of the spurious surfaces can be detected and removed at this stage. The results of feature matching is used to verify the generated surface by imposing the constraint that each surface should lead to at least four non-collinear line matches.

Note that, a planar surface and related matches are verified based on converging evidence. For example, if line segments from multiple images are matched and the planar surface is verified, it is known that

- The lines are consistent geometrically,
- The neighborhood of the lines are consistent photometrically,
- There are other consistent line matches generated by the same planar homography, and
- The planar surface containing the 3D lines leads to small re-projection error.

Current line matching algorithms use only a subset of above criteria.

### 3.8 Surface Merging

Any two planar surfaces, whose projections in any image are connected, are considered for merging, if they are almost coplanar, that is, *i*) The angle between the surface normals is small ( $< 5^\circ$ ) and, *ii*) The distance of the centers of planar surfaces to the plane defined by the other planar surface are small ( $< 2s$ , two times the surfel size). The two candidate surfaces are merged into a single planar surface if the feature match function of the merged plane is larger than the sum of feature match functions of each candidate.

### 3.9 Boundary Detection and Correction

Since the matched line segments are not necessarily borders of the surface, a method is necessary for classifying the segments as borders and non-borders. For boundary detection, a method similar to that of Chen and Lin [12] is used. The number of surfels along a strip is counted at both sides of the merged 2D lines on the planar surface (See Figure 3.8). If 80% of the strip at one side is full while 80% of the other side is empty, the 3D edge is labeled as border of the planar surface and the surfels in the neighborhood of the line are arranged accordingly.

Finally, the 2D convex hull of the matched features are calculated and the surfels that are outside the complex hull are removed, because the borders of real planar surfaces are, most probably, linear segments. As a result, surfels wrongly added due to weak texture are eliminated, in return, we take the risk of losing some correct surfels, if some border segments are not matched.

### 3.10 Surface Growing and Frequently Used Constraints

Smoothness, uniqueness, compatibility and orderedness are constraints frequently used in image matching as explained in Chapter 2. The planar surfaces are smooth by definition, so such a constraint is not used explicitly. Since the

surfels on different surfaces do not interact at all, the destructive effects of regularization around discontinuities are totally avoided. Uniqueness constraint is satisfied because surfels belonging to different surfaces are not allowed to project onto the same pixel. Compatibility is automatically satisfied by planar homographies. Finally, orderedness constraint is not imposed because it may be violated in real scenes.

### 3.11 Algorithm Complexity

Let the number of images be  $n$  and the area of an image be  $m$ . Then the complexity of the surface growing algorithm, except the feature matching and feature match function computation, is  $O(nm)$ . In our experiments, the algorithm spent only a small fraction of the computation power in the feature matching part, however, the complexity of feature matching is  $O(nl)$  and that of feature match function is  $O(n^2l^2)$  where  $l$  is average number of edges in an image. If it is assumed that  $l$  is proportional to  $m$ , their complexities become  $O(nm)$  and  $O(n^2m^2)$ , respectively.

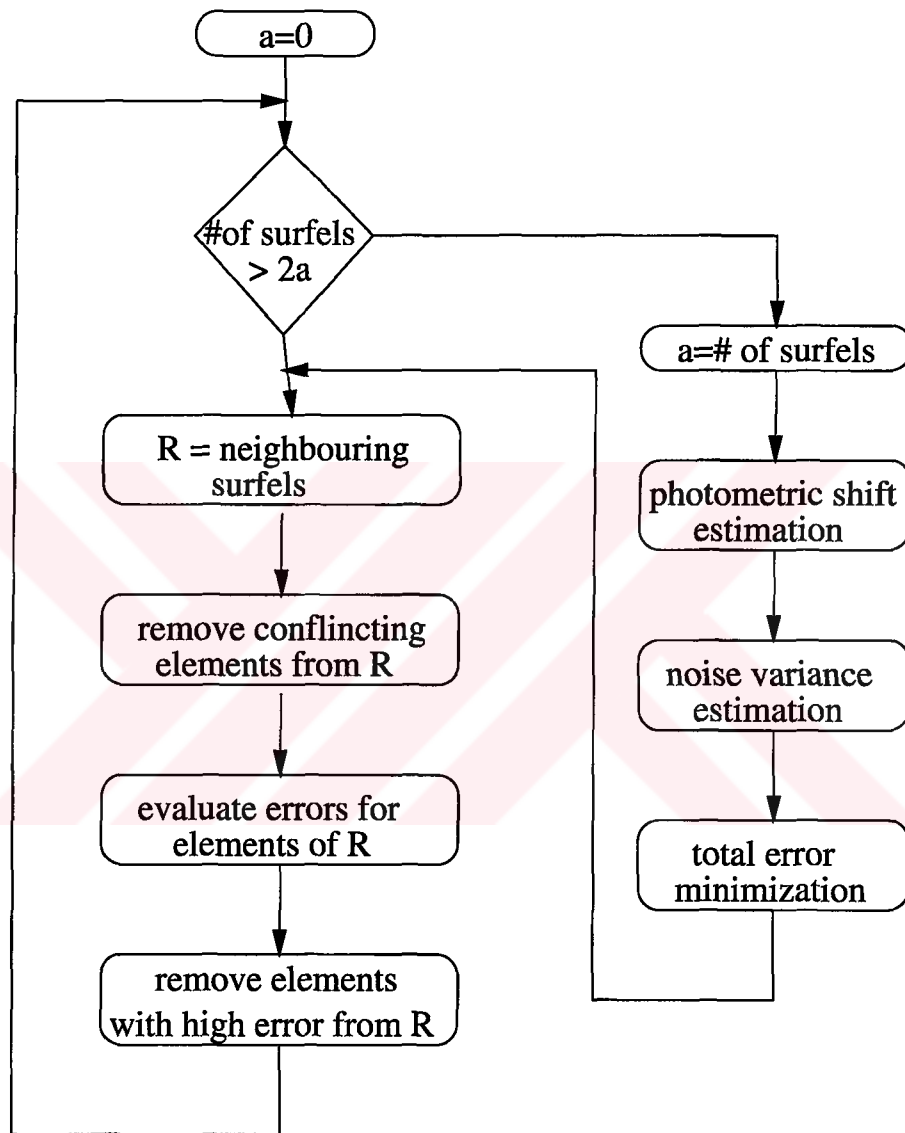


Figure 3.6: Flow diagram of the surface growing algorithm.

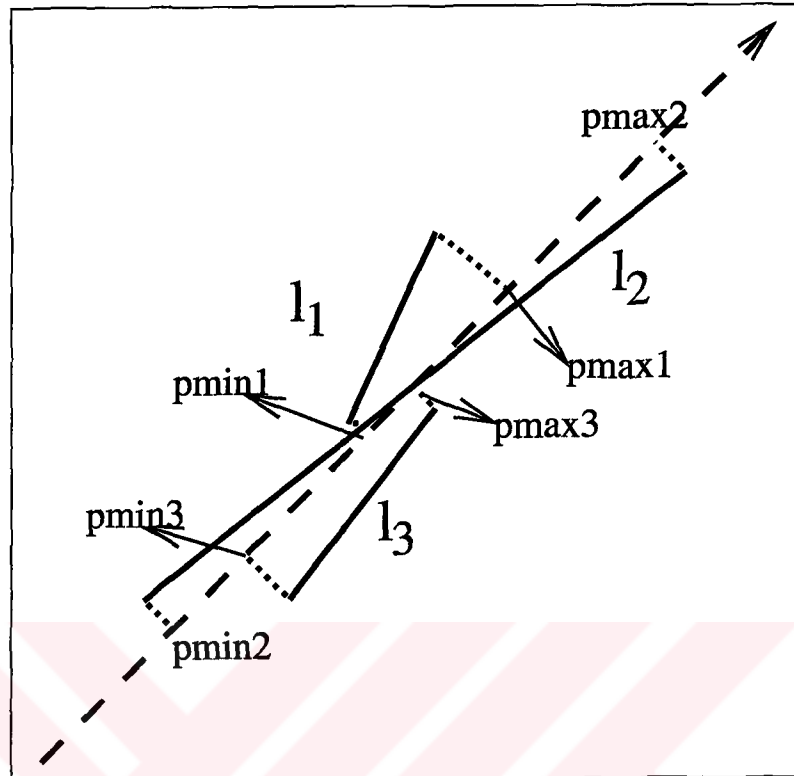


Figure 3.7: Projections of three line segments on the 2D infinite line.

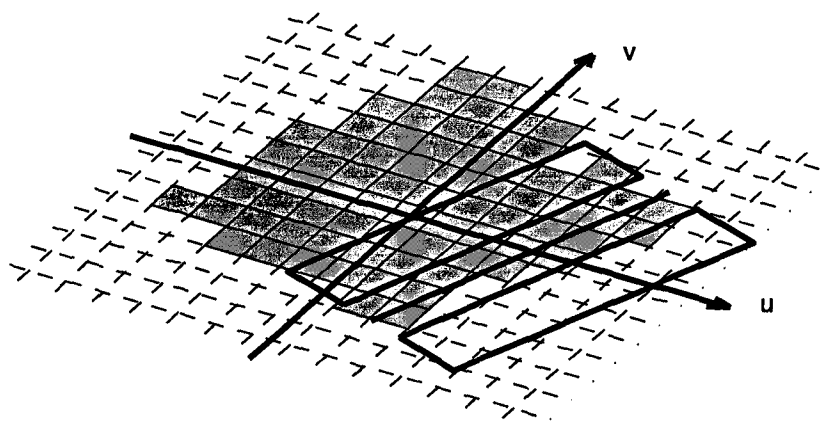


Figure 3.8: Two strips at both sides of the line is used to detect surface boundaries.



## CHAPTER 4

### EXPERIMENTAL RESULTS

The algorithm proposed is implemented in the C++ programming language. The program package is developed on both Linux and Solaris operating systems. The VRML language is used for visualization of the surfaces generated in 3D.

Two sets of high-resolution aerial color images of urban areas are used for testing the algorithm. The scenes contain mostly man-made objects like buildings, so planar approximation is reasonable. On the other hand, 3D modelling is difficult because of large textureless areas, frequent and abrupt depth changes, occlusions, glossy surfaces and moving cars. The first set of images are taken by Eurosense in Brussels and they are fully calibrated. Manually obtained VRML models of some buildings are also available. The ground resolution of the images are approximately 8.5 cm. We have chosen the surfel size as 7 cm accordingly. In Figure 4.1, three images that we have used are

shown. The second data set is the AMOBE dataset from ETH Zurich. Besides the internal and external camera parameters, the models of buildings in a few formats and a digital terrain model are available. The ground resolution is approximately 7.5 cm.

Feature extraction program FEX [26] developed by the University of Bonn is used for linear edge segment detection. FEX is preferred because, it can work on multi-channel data, there are small number of parameters and includes an information preserving filter.

An example of a surface patch constructed using the algorithm is shown in Figure 4.2. Subfigures 4.2.a-c show three views of the roof used as example while Subfigures 4.2.d-f show detected line segments. In 4.2.g, the surface patch is projected on the image in 4.2.a. The generating seed is painted to a different color. 4.2.h shows matched line segments and 4.2.i shows the border segments with corrected surface patch.

In Figure 4.3, all the constructed surfaces are re-projected on the first image shown in Figure 4.1. Figure 4.4 and Figure 4.5 depict two different 3D renderings of the surfaces, with and without texture, respectively. Figure 4.6 shows a reconstructed roof with the VRML model obtained manually.



Figure 4.1: Three Brussels images used from the IMPACT dataset. (Courtesy of Eurosense.) The roof shown in Figure 4.2 is marked with a rectangle in the first image.

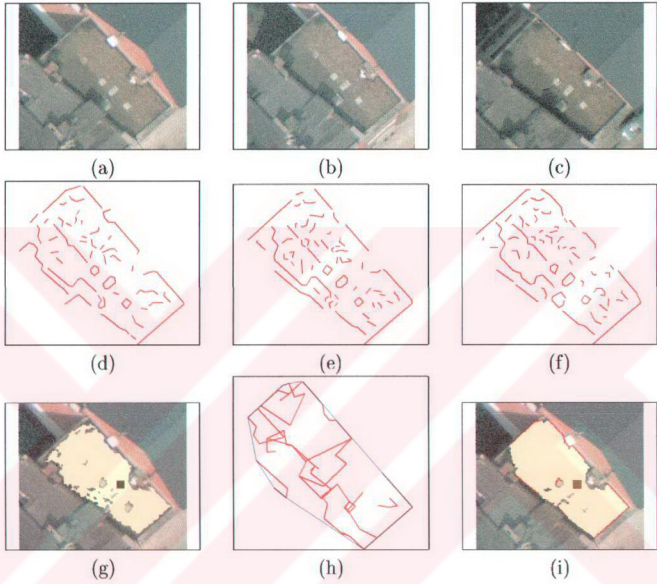


Figure 4.2: Example of a growing surface: (a)(b)(c) are views of the same roof in three images. (d)(e)(f) are the detected linear segments in those images. (g) is the projection of the corresponding surface in the first image. (h) is all the matched segments with the convex hull of the segments. (i) is the surface after correction with detected border segments.

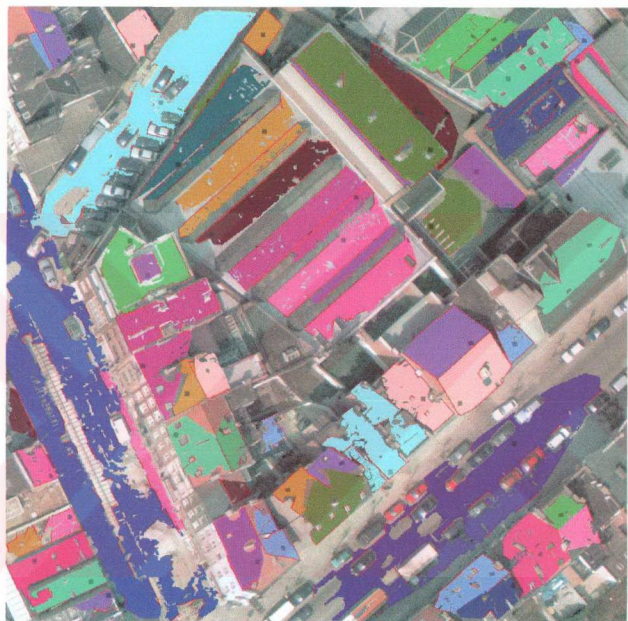


Figure 4.3: Reconstructed surfaces re-projected on one of the Brussels images. Each surface is shown in a different color.



Figure 4.4: 3D rendering of the surfaces obtained from the Brussels images (With texture).



Figure 4.5: 3D rendering of the surfaces obtained from the Brussels images (Without texture).

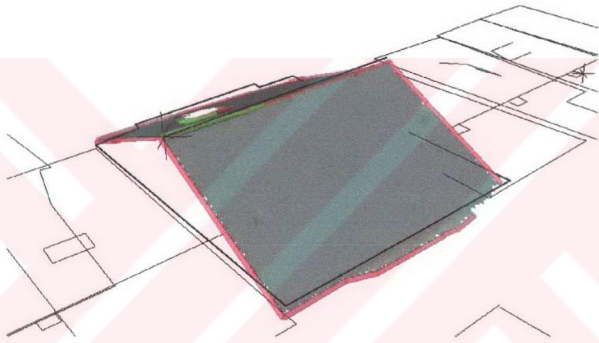


Figure 4.6: 3D rendering of a roof with the VRML model obtained manually.

The seed shown in Figure 4.2.g is generated by minimization of the error function on three images. In Figure 4.7, the total projection error is drawn as a function of height and angle. As one can see, the minimum is very shallow along the direction of the angle. That is why the surface parameters are tuned during surface growing. Inspecting the evolution of the error function and the feature match function during the surface growing process shows how the parameters are tuned better as the surface gets larger. For that purpose, the largest reconstructed surface is used. Figure 4.8 shows four stages of surface growing coded in different colors. In Figure 4.9, the re-projection is drawn as a function of two rotation angles around the unit vectors  $\mathbf{u}$  and  $\mathbf{v}$ . In Figure 4.10, the feature match function is drawn in the same way. It is clearly seen that the peak of the function is getting sharper and getting closer to the correct value in both cases as the surface grows, so the surface parameters can be estimated more precisely for larger surfaces.

For analyzing the error distributions and photometric distortion, two surfaces are used: *i)* The surface on the horizontal roof shown in Figure 4.2 and *ii)* The surface shown in Figure 4.11.

In Figure 4.12 and Figure 4.13, the error distributions, that is, the distributions of  $\mathbf{c}_{ji}^k - \hat{\mathbf{c}}_{ji}$  values in each channel of each image, are shown for two different surfaces. The additive constants (mean values) are clearly seen. A comparison of Table 4.2 and Table 4.1 justifies the use of different additive



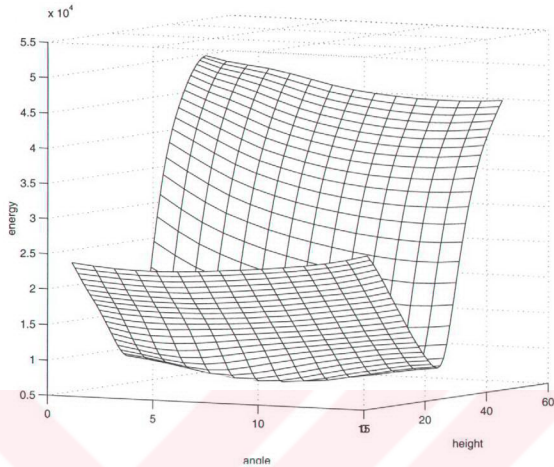


Figure 4.7: Total re-projection error as a function of depth and angle around the unit vector  $\mathbf{u}$ .

constants for each surface. Some researchers estimate global correction factors for each image. Since correction factors change from surface to surface, global correction will not work. Most image matching algorithms use normalized cross correlation for image matching, so the correction factors are estimated for each correlation window. In that case, the area is too small for accurate estimation of the parameters. Planar surfaces are the right scale between these two extremes. There are two possible explanations of why the parameters are almost constant in a planar surface patch and varies across different patches: Firstly, a surface is made of similar material in general. Secondly, the viewing angle does not change for points on the same planar patch.



Figure 4.8: Four stages of surface growing coded in different colors. The evolution of the error function is inspected during the surface growing process.

In Figure 4.14 three Zurich images from the AMOBE dataset are shown. The reconstructed surfaces re-projected on the first image are shown in Figure 4.15. Figure 4.16 and Figure 4.17 show 3D renderings of the reconstructed surfaces.

Table 4.1: Error means and variances for the surface shown in Figure 4.11.

	Camera 1	Camera 2	Camera 3
Red channel	$m = 4.7, \sigma^2 = 26.2$	$m = -2.6, \sigma^2 = 26.9$	$m = -2.1, \sigma^2 = 37.7$
Green channel	$m = 4.1, \sigma^2 = 29.2$	$m = -0.9, \sigma^2 = 26.5$	$m = -3.1, \sigma^2 = 36.5$
Blue channel	$m = 4.6, \sigma^2 = 23.9$	$m = 0.7, \sigma^2 = 21.4$	$m = -5.3, \sigma^2 = 31.6$

Table 4.2: Error means and variances for the surface shown in Figure 4.2.

	Camera 1	Camera 2	Camera 3
Red channel	$m = 2.5, \sigma^2 = 47.3$	$m = 4.9, \sigma^2 = 46.4$	$m = -7.4, \sigma^2 = 74.3$
Green channel	$m = 1.8, \sigma^2 = 28.2$	$m = 6.0, \sigma^2 = 34.2$	$m = -7.8, \sigma^2 = 42.9$
Blue channel	$m = 1.0, \sigma^2 = 30.2$	$m = 8.9, \sigma^2 = 33.3$	$m = -9.9, \sigma^2 = 52.5$

## 4.1 Quality Assessment

To estimate the reliability of surfaces, 74 surfaces obtained for the IMPACT dataset are classified as true or false manually. 67 surfaces are classified as true and 4 surfaces as false. Remaining 3 surfaces are true but they extend into other surfaces. 3.41% of all the surfels belong to false surfaces, while 91.45% of the surfels belong to true surfaces. Correctness of the remaining 5.14% cannot be determined using this method.

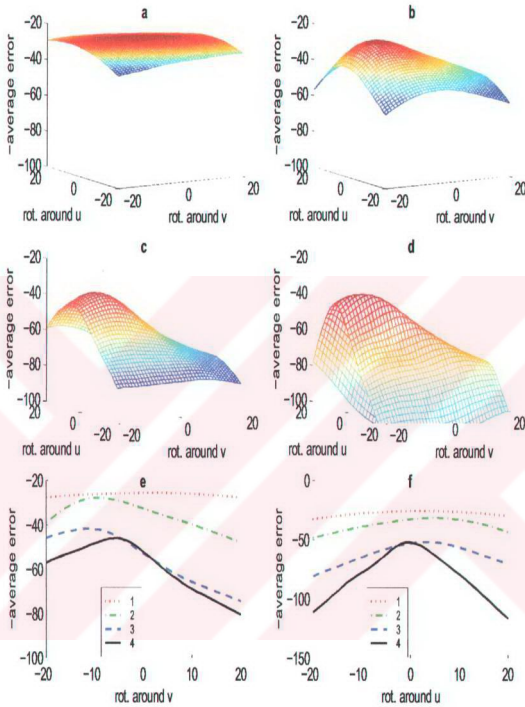


Figure 4.9: From *a*) to *d*): The evolution of the average re-projection error as the surface grows. The error is drawn as a function of rotation angles around the unit vectors  $\mathbf{u}$  and  $\mathbf{v}$ . *e*) and *f*): The error functions drawn as a function of rotation angles around the unit vectors  $\mathbf{u}$  and  $\mathbf{v}$ , respectively.

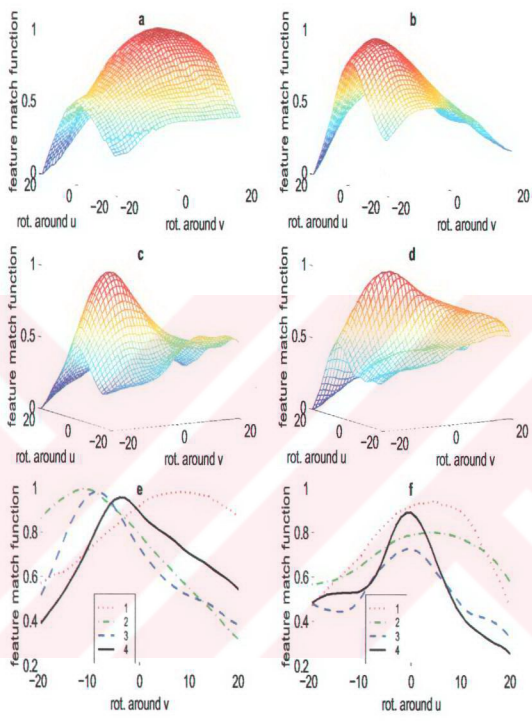


Figure 4.10: From *a*) to *d*): The evolution of the feature match function as the surface grows. Feature match is normalized to unity and is drawn as a function of rotation angles around the unit vectors  $\mathbf{u}$  and  $\mathbf{v}$ . *e*) and *f*): The feature match function drawn as a function of rotation angles around the unit vectors  $\mathbf{u}$  and  $\mathbf{v}$ , respectively.



Figure 4.11: The largest surface patch projected on one of the Brussels images. This surface is used for analyzing the error distributions and photometric distortion.

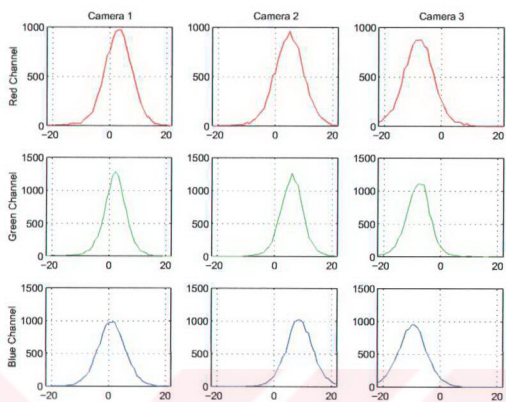


Figure 4.12: The error distributions in each channel of each camera for the surface in Figure 4.2.

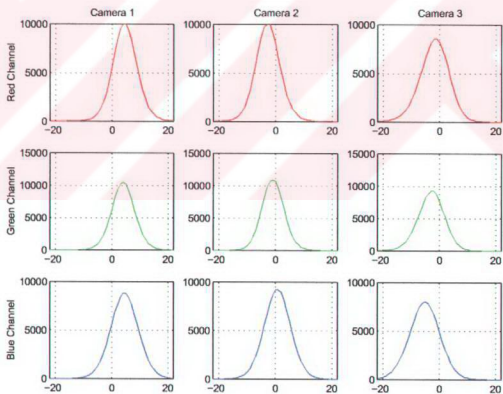


Figure 4.13: The error distributions in each channel of each camera for the surface in Figure 4.11.



Figure 4.14: Three images used from the AMOBE dataset. (Courtesy of IGP at ETH Zürich.)





Figure 4.15: Reconstructed surfaces re-projected on Zürich image. Each surface is shown in a different color.



Figure 4.16: 3D rendering with texture (AMOBÉ Image).



Figure 4.17: 3D rendering without texture (AMOB Image).

## CHAPTER 5

### CONCLUSION

In this thesis, we present a new algorithm for reconstructing surfaces directly from multiple cameras at arbitrary positions. A parameterized planar surface is used for surface representation in the object space. The surface model constrain the solution so that surfaces can be obtained in areas with weak texture. On the other hand, the planar surface patches are flexible enough to model man-made environments.

By employing an attention mechanism, the surfaces are reconstructed one by one. A surface growing algorithm is developed to obtain surfaces from seeds, that is, points of attention. As a result, the surfaces are correctly segmented and the unwanted effects of smoothing are avoided. The algorithm is both feature and area based. The re-projection error as well as a novel feature match function guide the surface growing. Besides guiding process, line segments are used for verifying the reconstructed surfaces and for detecting

surface boundaries. During surface growing, photometric correction parameters are estimated for each surface. It is shown that the planar surface is the right scale for photometric correction parameter estimation.

The decision on accepting or rejecting a surface is based on converging evidence from both re-projection error and groups of matched features. As a result, acceptance of a wrong match is very unlikely. The proposed algorithm is robust and it can successfully match even almost textureless areas. The algorithm does not require a matching phase and the results are directly available as models.

The performance of the algorithm is demonstrated on real images of complex environments, like urban areas. Most of the surfaces like roofs and roads are successfully reconstructed.

## 5.1 Future Work

- In this thesis, the surfaces are modeled by planar patches. But, the algorithm can easily be generalized to higher order surfaces. As a result, curved surfaces can be approximated better.
- Although our example images are all aerial images of urban areas, the algorithm is not specifically designed for building reconstruction. The algorithm can be tailored to consider specific properties of buildings (like symmetric roofs, vertical facets) so that the performance increases in the

domain of 3D building reconstruction. In such areas, vegetation can also be detected and seed points in the vegetation can be discarded.

- In the current implementation, only the surfaces that are seen by all the cameras are reconstructed. However, it is possible to reconstruct parts of the surfaces that can be seen in at least two images by making a second surface growing attempt. In this second growing process, the cameras that can not see the candidate surfels can be determined from already reconstructed surfaces and can be excluded from the error calculation.
- Implementing computer vision problems in a hierarchical structure is common practice. An image pyramid is formed for each image by low-pass filtering and subsampling. The algorithm works in a coarse-to-fine manner so that the solution at a level constrains the solution at the next finer level. This implementation both speeds up the process and increases the accuracy. The surface growing algorithm that we propose can also be implemented in a hierarchical system.
- An error-correction mechanism can be implemented with inspiration from binocular rivalry phenomenon, which suggests that the human visual system tries continuously new hypotheses until it resolves all conflicts and reaches a convincing solution. Similarly, if a new-generated surface fits the data better than former surfaces, it can grow into those surfaces, removing the conflicting surfels.

## REFERENCES

- [1] B. Ameri. Feature based model verification (FBMV): a new concept for validation in building reconstruction. In *XIXth Congress of The ISPRS*, volume XXXIII, pages 24–35, Amsterdam, Netherlands, 2000.
- [2] C. Baillard and A. Zisserman. A plane-sweep strategy for the 3D reconstruction of buildings from multiple images. In *19th ISPRS Congress and Exhibition*, Amsterdam, July 2000.
- [3] S. T. Barnard and M. A. Fischler. Computational stereo. *ACM Computing Surveys*, 14(4):553–572, 1982.
- [4] J. R. Beveridge, E. M. Riseman, and C. R. Graves. How easy is matching 2D line models using local search. *IEEE-PAMI*, 19(6), 1997.
- [5] N. Bhat and S. K. Nayar. Ordinal measures for image correspondence. *IEEE-PAMI*, 20(4):415–423, 1998.
- [6] N. Bhat and S. K. Nayar. Stereo and specular reflection. *International Journal of Computer Vision*, 26(2):91–106, 1998.
- [7] F. Bignone. Segment stereo matching and coplanar grouping. Technical Report BIWI-TR-165, ETH Zurich, Communication Technology Lab., Computer Vision Group, Switzerland, 1995.
- [8] Y. Boykov, O. Veksler, and R. Zabih. A variable window approach to early vision. *IEEE-PAMI*, 20(12):1283–1294, December 1998.
- [9] L. G. Brown. A survey of image registration techniques. *ACM Computing Surveys*, 24(4):325–376, December 1992.
- [10] D. Canu, N. Ayache, and J. A. Sirat. Accurate and robust stereovision with a number of aerial images. In *SPIE European Symposium on Satellite Remote Sensing II*, pages 1–9, Paris, France, 1995.

- [11] D. Canu, J.-P. Gambotto, J. A. Sirat, and N. Ayache. Reconstruction of buildings from multiple high resolution images. In *International Conference on Image Processing*, Lausanne, Switzerland, September 16–19 1996.
- [12] L.-H. Chen and W.-C. Lin. Visual surface segmentation from stereo. *Image and Vision Computing*, 15:95–106, 1997.
- [13] Y. Chen and G. Medioni. Description of complex objects from multiple range images using an inflating balloon model. *CVIU*, 61(3):325–334, May 1995.
- [14] Y.Q. Cheng, R.T. Collins, A.R. Hanson, and E.M. Riseman. Triangulation without correspondences. In *ARPA Image Understanding Workshop 94*, volume II, pages 993–1000, Monterey, CA, USA, November 1994.
- [15] R. T. Collins. A space-sweep approach to true multi-image matching. In *CVPR'96*, pages 358–363, San Fransisco, CA, USA, June 18–20, 1996.
- [16] R. T. Collins. Multi-image focus of attention for rapid site modeling. In *CVPR'97*, San Juan, Puerto Rico, June 1997.
- [17] N. D'Apuzzo. Motion capture by least squares matching tracking algorithm. In *AVATARS'2000*, Lausanne, Switzerland, 2000.
- [18] O. de Joinville, H. Maitre, G. Maillet, and M. Roux. How to design DEM's assessment maps. In *GISRUK1*, Glamorgan, UK, April 2001.
- [19] O. de Joinville, H. Maitre, D. Piquet Pellorce, and M. Roux. How to design DEM assessment maps. In *1st International Workshop on Pattern Recognition in Remote Sensing, PRSS'2000*, Andorra La Vella, Andorra, September 2000.
- [20] U. R. Dhond and J. K. Aggarwal. Structure from stereo – A review. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1489–1510, November/December 1989.
- [21] C. R. Dyer. Volumetric scene reconstruction from multiple views. In L. S. Davis, editor, *Foundations of Image Understanding*, pages 469–489. Kluwer, 2001.
- [22] O. Faugeras and R. Keriven. Complete dense stereovision using level set methods. In *ECCV98*, University of Freiburg, Germany, 2–6 June, 1998.
- [23] M. Fradkin, M. Roux, H. Maître, and U. M. Leloğlu. Surface reconstruction from multiple aerial images in dense urban areas. In *CVPR96*, pages 358–363, San Fransisco, CA, USA, June 18–20, 1996.

- [24] P. Fua. From multiple stereo views to multiple 3-D surfaces. *International Journal of Computer Vision*, 24(1):19–35, 1997.
- [25] P. Fua and Y.G. Leclerc. Using 3-dimensional meshes to combine image-based and geometry-based constraints. In *ECCV94*, pages B:281–291, 1994.
- [26] C. Fuchs and Stephan Heuel. Feature extraction. In W. Förstner, editor, *Proc. of Third Course in Digital Photogrammetry*, Bonn, Germany, 1998.
- [27] A. Fusiello. Uncalibrated Euclidean reconstruction: A review. *Image and Vision Computing*, 18(67):555–563, May 2000.
- [28] A. Fusiello, E. Trucco, and A. Verri. A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications*, 12(1):16–22, 2000.
- [29] S. Girard, P. Guerin, H. Maître, and M. Roux. Construction d’une carte de disparité dense pour la reconnaissance de bâtiments en imagerie aérienne. In *RFIA ’98*, Clermont-Ferrand, France, February 1998.
- [30] V. Gouet, P. Montesinos, and D. Pelé. Stereo matching of color images using differential invariants. In *International Conference on Image Processing 98*, Chicago, Illinois, US, October 4–7 1998.
- [31] R. I. Hartley. In defence of the eight-point algorithm. *IEEE-PAMI*, 19(6):580–593, 1997.
- [32] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049, 2000.
- [33] P. Havaldar and G. Medioni. Full volumetric descriptions from three intensity images. *IEEE-PAMI*, 20(5):540–545, 1998.
- [34] H. Heipke. Overview of image matching techniques. In *OEEPE Workshop on the Application of Digital Photogrammetric Workstations*, pages 81–88, Lausanne, Switzerland, March 4–6 1996.
- [35] U. V. Helava. Object-space least-squares correlation. *Photometric Engineering and Remote Sensing*, 54(6):711–714, June 1988.
- [36] E. Hemayed, A. Sandbek, A. Wassal, and A. Farag. Investigation of stereo-based 3D surface reconstruction. In *Proceedings of SPIE*, volume 3023, May 1997.
- [37] O. Henricsson, F. Bignone, W. Willuhn, F. Ade, O. Kubler, E. Baltsavias, S. Mason, and A. Grun. Project AMOBE: Strategies, current status, and



- future work. In *International Archives of Photogrammetry and Remote Sensing*, volume 31, pages 321–330, 1996.
- [38] A. Hilton, A. J. Stoddart, J. Illingworth, and T. Windeatt. Reliable surface reconstruction from multiple range images. In *4th European Conference on Computer Vision*, pages 117–126, April 1996.
- [39] A. Hilton, A. J. Stoddart, J. Illingworth, and T. Windeatt. Implicit surface-based geometric fusion. *Computer Vision and Image Understanding*, 69(3):273–291, 1998.
- [40] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganised points. In *Computer Graphics 26: SIGGRAPH'92 Conference Proceedings*, volume 26, pages 71–78, 1992.
- [41] B. Jia, Y. J. Zhang, and X. G. Lin. Stereo matching using both orthogonal and multiple image pairs. In *ICASSP'2000*, pages 2139–2142, Istanbul, Turkey, June 5–9 2000.
- [42] J. R. Jordan and A. C. Bovik. Using chromatic information in edge-based stereo correspondence. *CVGIP: Image Understanding*, 54(1):98–118, July 1991.
- [43] S. B. Kang and R. Szeliski. 3-D scene data recovery using omnidirectional multibaseline stereo. *International Journal of Computer Vision*, 25(2):167–183, 1997.
- [44] T. Kim and J. Muller. Automated urban area building extraction from high resolution stereo imagery. *Image and Vision Computing*, 14:115–130, 1996.
- [45] S. Kiriwara and H. Saito. Shape modeling from multiple view images using GAs. In *ACCV'98, Lecture Notes in Computer Science 1352*, volume II, pages 448–454, January 1998.
- [46] R. Koch. Surface segmentation and modeling of 3-D polygonal objects from stereoscopic image pairs. In *International Conference on Pattern Recognition 96*, Vienna, Austria, 1996.
- [47] C. D. Kuglin and D. C. Hines. The phase correlation image alignment method. In *IEEE International Conference on Cybernetics and Society*, pages 163–169, New York, USA, 1975.
- [48] K. N. Kutulakos and S. M. Seitz. A theory of shape by space carving. Technical Report CS-692, University of Rochester, USA, May 1998.

- [49] Y. Leclerc, Q. Luong, and P. Fua. Measuring the self-consistency of stereo algorithms. In D. Vernon, editor, *Proc. 6th European Conf. Computer Vision, ECCV'2000*, volume II, pages 282–298, Dublin, Ireland, July 2000.
- [50] S.H. Lee and J.J. Leou. A dynamic programming approach to line segment matching in stereo vision. *PR*, 27:961–986, 1994.
- [51] U. M. Leloğlu and Uğur Halıcı. Multi-image region growing for integrating disparity maps. In Franc Solina and Aleš Leonardis, editors, *Computer Analysis of Images and Patterns, CAIP'99*, pages 403–410, Ljubljana, Slovenia, 1999.
- [52] M. Lemmens. A survey on stereo matching techniques. In ???, volume 27, pages 11–23, 1998.
- [53] M. Lhuillier and L. Quan. Robust dense matching using local and global geometric constraints. In *IAPR International Conference on Pattern Recognition*, Barcelone, Spain, September 3–8, 2000.
- [54] C. W. Liao and G. Medioni. Surface approximation of a cloud of 3D points. *GMIP*, 57(1):67–74, January 1995.
- [55] H. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, (293):133–135, 1981.
- [56] Q. T. Luong. *Matrice Fondamentale et Calibration Visuelle sur l'Environnement - Vers une Plus Grande Autonomie des systèmes Robotiques*. PhD thesis, Université de Paris-Sud, Centre d'Orsay, France, 1992.
- [57] J. L. Martin and J. Crowley. Experimental comparison of correlation techniques. In *3rd Int. Sym. on Intelligent Robotic Systems'95*, pages 287–294, Pisa, Italy, July 10–14 1995.
- [58] G. Medioni and R. Nevatia. Segment-based stereo matching. *CVGIP*, 31:2–18, 1985.
- [59] J. P. Mellor. Automatically recovering geometry and texture from large sets of calibrated images. Technical Report AI-1674, Massachusetts Institute of Technology, USA, 1999.
- [60] R. Mohan, G. Medioni, and R. Nevatia. Stereo error detection, correction and evaluation. *IEEE-PAMI*, 11(2):113–120, February 1989.
- [61] P. Narayanan, P. Rander, and T. Kanade. Constructing virtual worlds using dense stereo. In *The Proceeding of International Conference on Computer Vision, ICCV'98*, pages 3–10, Bombay, India, 1998.

- [62] N. M. Nasrabadi and C. Y. Choo. Hopfield network for stereo vision correspondence. *IEEE Transactions on Neural Networks*, 3(1):5–13, January 1992.
- [63] M. Okutomi and T. Kanade. Multiple-baseline stereo. *IEEE-PAMI*, 15(4):353–363, 1993.
- [64] G. P. Otto and T. K. W. Chau. Region-growing algorithm for matching of terrain images. *Image and Vision Computing*, 7(2):83–94, 1989.
- [65] N. Paparoditis and G. Maillet. Improving aerial image matching techniques in urban areas using a new true multi-stereo approach guided from object space. In *3rd Int. Sym. on Mobile Mapping Technology*, Cairo, Egypt, 2001.
- [66] H. J. Park and P. Zimmermann. Colour image matching for DTM generation and house extraction. In *XIXth Congress of The ISPRS*, volume XXXIII, pages 679–704, Amsterdam, Netherlands, 2000.
- [67] F. Pedersini, P. Pigazzini, A. Sarti, and S. Tubaro. 3D area matching with arbitrary multiview geometry. *Signal Processing: Image Commun.*, 14:7–19, 1998.
- [68] A. Redert, C.-J. Tsai, E. Hendriks, and A. K. Katsaggelos. Disparity estimation with modeling of occlusion and object orientation. In *Proceedings of SPIE VCIP'98*, San Jose, CA, USA, January 24–30 1998.
- [69] D. Rosenholm. Multi-point matching using the least-squares technique for evaluation of three-dimensional models. *Photometric Engineering and Remote Sensing*, 53(6):621–626, June 1987.
- [70] F. Ade S. Scholze, T. Moons and L. Van Gool. Exploiting color for edge extraction and line segment stereo matching. In *ISPRS*, pages 815–822, Amsterdam, Netherlands, 2000.
- [71] C. Schmid and A. Zisserman. Automatic line matching across views. In *CVPR97*, San Juan, Puerto Rico, June 17-19, 1997.
- [72] H. Schultz. Shape reconstruction from multiple images of the ocean surface. *Photogrammetric Engineering and Remote Sensing*, 62(1):93–99, 1996.
- [73] J. Shao, R. Mohr, and C. S. Fraser. Multi-image matching using segment features. In *XIXth Congress of The ISPRS*, pages 837–844, Amsterdam, Netherlands, 2000.

- [74] C. Silva and J. Santos-Victor. Intrinsic images for dense stereo matching with occlusions. In *Proc. ECCV'2000*, Dublin, Ireland, July 2000.
- [75] O. Skrinjar, H. Tagare, and J. Duncan. Surface growing from stereo images. In *Conference on Computer Vision and Pattern Recognition (CVPR 2000)*, volume II, pages 571–576, Hilton Head Island, SC, USA, June 2000.
- [76] G. Slabaugh, B. Culbertson, T. Malzbender, and R. Schafer. A survey of methods for volumetric scene reconstruction from photographs. In *International Workshop on Volume Graphics*, page to appear, Stony Brook, New York, USA, 2001.
- [77] C. C. Slama, C. Theurer, and S. W. Henriksen, editors. *Manual of Photogrammetry*. American Society of Photogrammetry, ISBN: 0-937294-01-2, 1980.
- [78] B. J. Super and W. N. Klarquist. Patch-based stereo in a general binocular viewing geometry. *IEEE-PAMI*, 19(3):247–253, 1997.
- [79] R. Szeliski. Scene reconstruction from multiple cameras. In *International Conference on Image Processing (ICIP-2000)*, volume I, pages 13–16, Vancouver, Canada, 2000.
- [80] R. Szeliski and R. Zabih. An experimental comparison of stereo algorithms. In *Workshop on Vision Algorithms*, pages 1–19, Kerkyra, Greece, 1999.
- [81] Richard Szeliski and David Tonnesen. Surface modeling with oriented particle systems. *Computer Graphics*, 26(2):185–194, 1992.
- [82] J.-P. Tarel. Global 3D planar reconstruction with uncalibrated cameras and rectified stereo geometry. *Machine Graphics and Vision*, 6(4):393–418, 1997.
- [83] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment – A modern synthesis. In W. Triggs, A. Zisserman, and R. Szeliski, editors, *Vision Algorithms: Theory and Practice*, LNCS, pages 298–375. Springer Verlag, 2000.
- [84] R. Y. Tsai. Multiframe image point matching and 3-D surface reconstruction. *IEEE-PAMI*, 5(2):159–174, 1983.
- [85] Radim Šára and Ruzena Bajcsy. Fish scales: Representing fuzzy manifolds. In *The Proceeding of International Conference on Computer Vision, ICCV'98*, pages 811–817, Bombay, India, January 1998.

- [86] Z.F. Wang and N. Ohnishi. Intensity based stereo vision: from 3D to 3D. In *Intelligent Robots and Computer Vision XIII: 3D Vision, Product Inspection and Active Vision, Proc. SPIE 2354*, pages 434–443, Boston, MA, USA, 1994.
- [87] Z.F. Wang and N. Ohnishi. Deformable template based stereo. In *Proceedings of 1995 IEEE International Conference on System, Man and Cybernetics*, pages 3884–3889, Vancouver, British Columbia, Canada, 1995.
- [88] L. Zhang and S. M. Seitz. Image-based multiresolution modeling by surface deformation. Technical Report CMU-RI-TR-00-07, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, March 2000.
- [89] Z. Zhang. Estimating motion and structure from correspondences of line segments between two perspective images. *IEEE-PAMI*, 17(12), 1995.
- [90] A. Zisserman and C. Baillard. the IMPACT project. In *ISPRS Conference on Automatic Extraction of GIS Objects from Digital Imagery*, Germany, September 8–9, 1999. Technische Universität München.
- [91] V. Zyka and R. Sara. Polynocular image set consistency for local model verification. In *Application of 3D-Imaging and Graph-based Modeling 2000, 24th Workshop of the Austrian Association for Pattern Recognition*, pages 81–88, Villach, Austria, 2000.

## VITA

Uğur Murat Lelođlu is born in Erzurum on November 14, 1967. He left the Medical School of Hacettepe University after completing the first two years of the program successfully. He received his B.S. degree in Electrical and Electronics Engineering from Middle East Technical University in 1991. He has received his M.Sc. degree from the same department in 1994. He has been working for Information Technologies and Electronics Research Institute of the Scientific and Technical Research Council of Turkey since 1991 with the exception of one year visiting researcher position in Ecole Nationale Superieure des Télécommunications in Paris between July 1997 and July 1998.