Surface Reflectance and Shape from Images Using a Collinear Light Source

Jiping Lu Jim Little

Technical Report 97-5

April 1997

Laboratory for Computational Intelligence

Department of Computer Science

The University of British Columbia

Vancouver BC Canada V6T 1Z4

e-mail: {jplu, little}@cs.ubc.ca

Abstract

The purpose of computer vision is to extract useful information from images. Image features such as occluding contours, edges, flow, brightness, and shading provide geometric and photometric constraints on the surface shape and reflectance of physical objects in the scene. In this thesis, two novel techniques are proposed for surface reflectance extraction and surface recovery. They integrate geometric and photometric constraints in images of a rotating object illuminated under a collinear light source (where the illuminant direction of the light source lies on or near the viewing direction of the camera). The rotation of the object can be precisely controlled. The object surface is assumed to be C^2 and its surface reflectance function is uniform.

The first technique, called the *photogeometric technique*, uses geometric and photometric constraints on surface points with surface normal perpendicular to the image plane to calculate 3D locations of surface points, then extracts the surface reflectance function by tracking these surface points in the images. Using the extracted surface reflectance function and two images of the surface, the technique recovers the depth and surface orientation of the surface simultaneously.

The second technique, named the *wire-frame technique*, further exploits geometric and photometric constraints on the surface points whose surface orientations are coplanar with the viewing direction and the rotation axis to extract a set of 3D curves. The set of 3D curves comprises a wire frame on the surface. The depth and surface orientation between curves on the wire frame are interpolated by using geometric or photogeometric methods.

The wire-frame technique is superior because it does not need the surface reflectance

function to extract the wire frame. It also works on piecewise uniform surfaces and requires only that the light source be coplanar with the viewing direction and the rotation axis. In addition, by interpolating the depth and surface orientation from a dense wire frame, the surface recovered is more accurate.

The two techniques have been tested on real images of surfaces with different reflectance properties and geometric structures. The experimental results and comprehensive analysis show that the proposed techniques are efficient and robust. As an attempt to extend our research to computer graphics, work on extracting the shading function from real images for graphics rendering shows some promising results.

Table of Contents

List of Tables

List of Figures

Acknowledgement

The Ph.D program is a long way to go. I felt lost several times on the way. I am very grateful to my supervisor Jim Little for his encouragement, supervision and continuous support. Without him, I would not go back on the right track.

I am grateful to the other members of my supervisory committee: Dr. Alain Fournier, Dr. David Lowe, Dr. Bob Woodham for their constructive comments and valuable suggestions on my research. I would like to give special thanks to Dr. Bob Woodham. My work is made possible by using the facilities and objects he designed for photometric stereo. I am also grateful to Dr. Dinesh Pai, Dr. Gunther Schrack and Dr. Mabo Ito for their participation on the examination committee.

I thank Rob Barman, Stewart Kingdon and Dan Razzell for their continuous hardware, software support, and computer resource management.

I thank my fellow students, Jeffrey Beis, Andreas Siebert, Cristina Siegerist, Richard Pollock, Jane Mulligan, Michael Horsch, Paul Lalonde, Bob Lewis and many others for their help and friendship. I give special thanks to Bob Price and Daniel McReynolds for their proofreading of this thesis.

I acknowledge the financial assistance I received from the Institute for Robotics and Intelligent Systems (IRIS), one of the Canadian Networks of Centres of Excellence and from the Natural Science and Engineering Research Council of Canada in the form of research grants to Jim Little.

I thank my wife Lorna Zhang and my daughter Lucy Lu for their love, patience and understanding.

This thesis is dedicated to my parents, Tinxin Lu and Ma Xiufang.

Chapter 1

Introduction

1.1 Background and Objectives

Computer vision is the process of using computers to extract useful information about the physical world from images. The information to be extracted can be color, shape, texture, motion, or depth as well as surface reflectance, orientation and location. To extract the right information, a set of constraints has to be identified and used to interpret images unambiguously. These constraints can be features in images such as contours, surface marks, edges, image brightness and shading. They can also be knowledge and assumptions about the objects in the scene and the imaging conditions, such as smoothness of the object surface, the reflectance properties of the object surface, motion pattern of the object, and the properties of the light source.

Image features provide geometric and photometric constraints on the surface shape and reflectance of the physical objects in the scene. These constraints can be used to extract the surface reflectance function and surface shape. Conventional stereo techniques, which use only geometric constraints such as surface marks and edges, cannot recover objects of uniform reflectance. Shape from shading techniques which use only one image are usually computationally intensive and non-robust.

The major objective of this thesis is to develop robust techniques for surface reflectance extraction and surface recovery in a controlled environment by using both geometric and photometric constraints. The secondary objective is to apply the extracted reflectance function to computer graphics for realistic rendering.

1.2 Our Approach

To achieve our objectives, we explore and integrate geometric and photometric constraints on images of a rotating object illuminanted by a collinear light source (where the illuminant direction of the light source lies on or near the viewing direction of the camera) for extracting surface reflectance function, and surface shape. The images are taken under orthographic projection. The object surface is C^2. The reflectance of the surface is assumed to be uniform. The rotation axis of the object is perpendicular to the viewing direction, and the rotation angle of the object can be controlled.

Although the surface is smooth and uniform, there are still a number of features such as singular points (image points of maximum brightness), occluding contour points (image points projected from surface points with surface normal perpendicular to the viewing direction), and brightness in the images. Each individual image feature constrains the surface shape and reflectance. Of critical importance is that these image features are not independent and their integration can yield more useful constraints about the surface. For example, under a collinear light source and orthographic projection, a singular point in one image is related to an occluding contour point in another image taken after $\pi/2$ rotation of the object. From the relation, the 3D location and surface orientation at the singular point can be determined.

Under the illumination of a collinear light source and orthographic projection, the surface reflectance function, which describes the relation between the image brightness and the surface orientation at a surface point, is a function only of the emergent angle, the angle between the viewing direction and the surface normal. Because of this simplicity, the surface reflectance function can be determined by tracking brightness and surface orientations of some surface points on a rotating object.

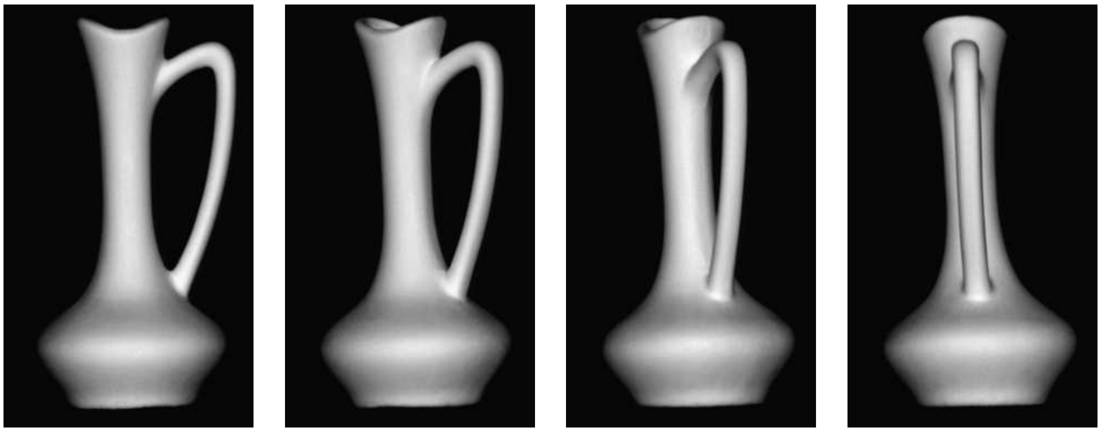
Assuming the reflectance function is strictly monotonic, then its inverse exists. The inverse of the extracted surface reflectance function can be used to determine the orientation at a surface point. The image brightness value of a surface point provides one constraint on its surface normal. The brightness values of a surface point in two images taken after the object rotation by two different angles will fully determine its surface normal. To extract the brightness values of a surface point in two different images, the projections of the surface point in the two images must be known. The projections can be computed from the 3D location of the surface point.

For a C^2 surface, the locations of surface points can be integrated from surface orientations. If surface orientation is known everywhere on the surface, then what is needed is the location of a starting surface point for the integration. As an image is a discrete 2D spatial array, the integration at each step is approximated by the first-order Taylor series.

Suppose the location of a surface point is known. Its surface orientation can be determined from its brightness values in two images. Then the locations of its neighboring surface points can be approximated by the first-order Taylor series. For each neighboring point, after its location has been obtained, its projections on images can be located and its surface orientation can be determined from its brightness values in the images. The location and surface orientation at each neighboring point can be used to calculate the locations of the surface points around each neighboring point. In this way, the 3D location and surface and surface orientation can be recovered over the entire visible surface.

By further exploring and integrating the geometric and photometric constraints, we find that the location and surface orientation on a $p = 0$ curve (a curve on which the surface normal at every point is coplanar with the rotation axis and viewing direction) can be determined without the surface reflectance function. Each point on a $p = 0$ curve gives the same brightness value in the two images taken after the object has been rotated by the same angle but in the two different directions. The disparity of the projections of a surface point in the two images depends only on the depth of the surface point. The depth (the distance from a surface point to the plane which is parallel to the image plane and contains the rotation axis) on a $p = 0$ curve can be measured from the corresponding occluding contour in the image taken after $\pi/2$ rotation of the object. Using these constraints, the projections of a $p = 0$ curve in the two images are found by a search method. Then the 3D location of the $p = 0$ curve is calculated from the projections. Rotating the object by an angle, another 3D curve on the surface will become a $p = 0$ curve. The 3D location of the curve can be found by the same method. In this way, a set of curves on a surface can be obtained.

The surface reflectance function in computer vision and the shading function in computer graphics both describe the relation between brightness and surface orientation in the image formation process. A typical shading function has a set of parameters and these parameters are usually determined heuristically. The surface reflectance function extracted from images directly comes from the image formation process. In theory and practice, to achieve realistic graphics rendering, the reflectance function extracted from images will give better results than the heuristic shading function. The shading function is intended to render objects under arbitrary viewing and illuminant directions while the surface reflectance function extracted from real images only gives the relation under a collinear light source. However, the surface reflectance function can be considered as a shading function under a collinear light source and the parameters of the shading function can be estimated from the surface reflectance function. Then the estimated shading function can be used to generate synthetic images of surfaces to achieve realistic rendering.



a. 0 rotation b. $\pi/6$ rotation c. $\pi/3$ rotation d. $\pi/2$ rotation

Figure 1.1: Four images from an image sequence of a rotating vase.

1.3 Experimental Results

The techniques have been tested on real images of several objects with different surface reflectance properties and surface geometry structures. Figure 1.1 shows four of a nineteen image sequence of a rotating clay vase. The solid line in Fig. 1.2 displays the surface reflectance function extracted from the image sequence. Figures 1.3 and 1.4 show the depth and surface orientation recovered from images using the photogeometric technique.

Figure 1.5 shows six of a thirty six image sequence of a rotating porcelain cup. Figure 1.6 shows the 3D wire frame extracted from the image sequence using the wire-frame technique. Figure 1.7 shows the recovered surface of the cup.

The preliminary experiments on estimating the shading function from real images for graphics rendering show that the technique is feasible. The dashed line in Fig 1.2 represents the shading function estimated from the surface reflectance function. Figure 1.8 is a group of synthetic images generated by rendering the surface orientation data, recovered by the photogeometric technique, with the shading function. Each image in the group of

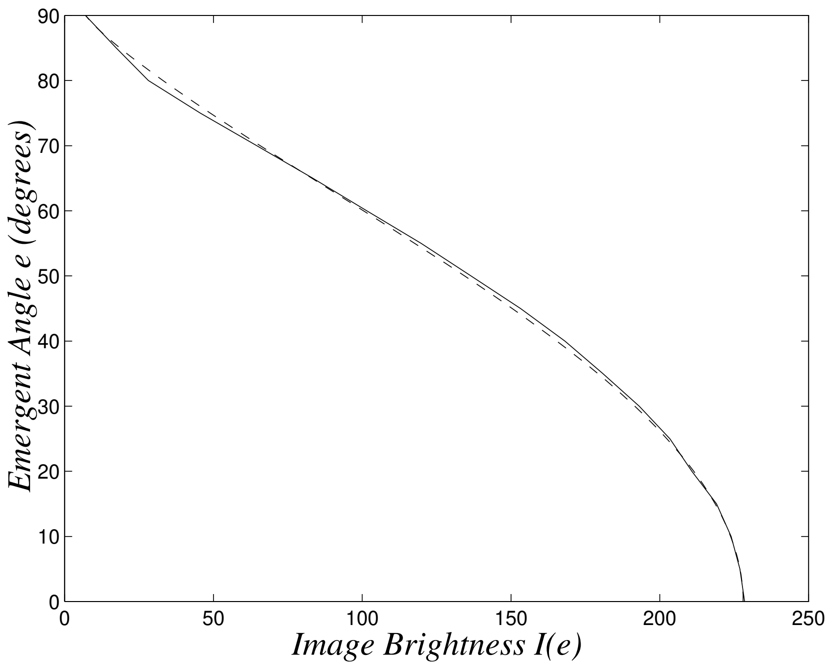


Figure 1.2: The extracted reflectance function (dashed line) and the estimated shading function (dashed line) of the vase.

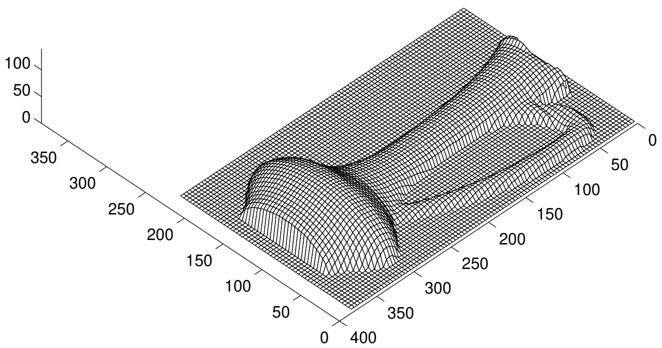


Figure 1.3: Surface depth plot of the recovered vase.

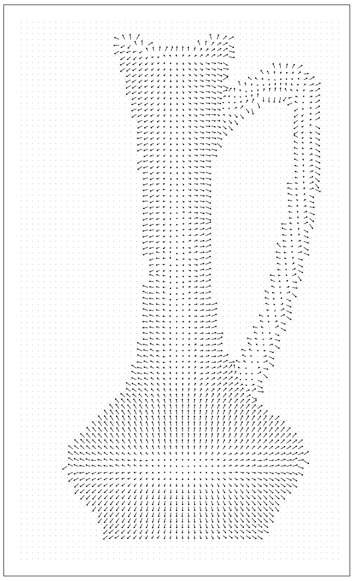


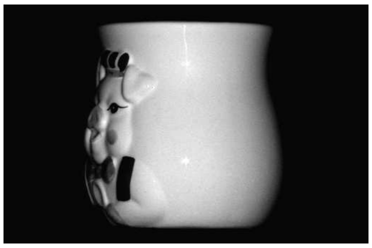
Figure 1.4: Recovered surface orientation.

0 rotation

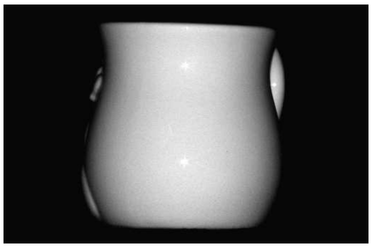
$\pi/3$ rotation

$2\pi/3$ rotation

π rotation



$4\pi/3$ rotation



$5\pi/3$ rotation

Figure 1.5: Six images from an image sequence of a rotating cup. The label indicates the rotation angle of the object. The pig figure violates the smooth and uniform surface assumption.

Figure 1.6: A 3D wire frame extracted from images of a cup

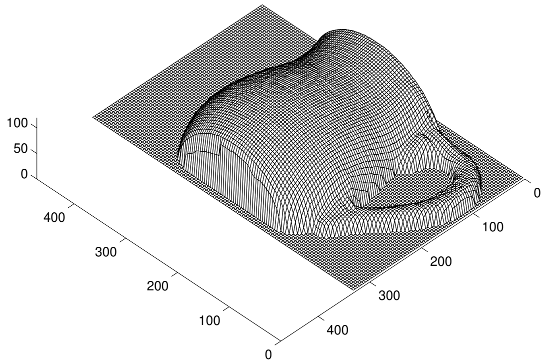
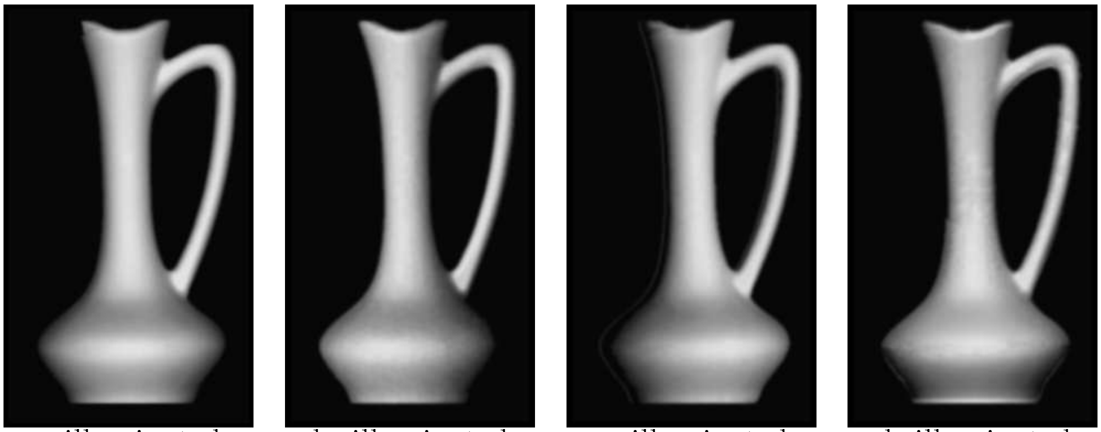
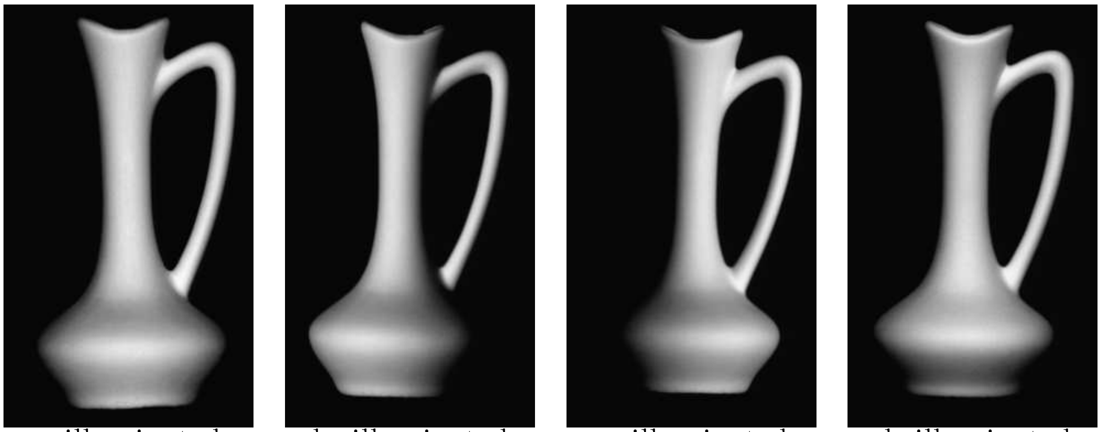


Figure 1.7: The surface depth plot of the recovered cup



a. illuminated at $(0, 0, 1)$	b. illuminated at $(-\tan\frac{\pi}{6}, 0, 1)$	c. illuminated at $(\tan\frac{\pi}{6}, 0, 1)$	d. illuminated at $(0, \tan\frac{\pi}{10}, 1)$

Figure 1.8: Synthetic images of the vase rendered with the estimated shading function.



a. illuminated at $(0, 0, 1)$	b. illuminated at $(-\tan\frac{\pi}{6}, 0, 1)$	c. illuminated at $(\tan\frac{\pi}{6}, 0, 1)$	d. illuminated at $(0, \tan\frac{\pi}{10}, 1)$

Figure 1.9: Real images of the vase viewed and illuminated under the similar conditions.

real images (see Fig. 1.9) corresponds to an image in the group of synthetic images. The corresponding images are illuminated and viewed under the same condition so that the visual comparison can be made to verify the similarity between the real and synthetic images.

1.4 Thesis Outline

The rest of this thesis is organized as follows:

Chapter 2 gives a comprehensive survey of the work closely related to this thesis.

Chapter 3 explores geometric and photometric constraints as well as their relations in an image sequence of a rotating object illuminated under a collinear light source.

Chapter 4 proposes what we call the photogeometric technique, which extracts the surface reflectance function from an image sequence and then recovers the surface depth and orientation using the surface reflectance function.

Chapter 5 describes what we call the wire-frame technique, which extracts a 3D wire frame on an object surface from the image sequence of the object without using the surface reflectance function.

Chapter 6 discusses the practical problems with the experimental conditions and the measures used to overcome these problems. This chapter can be skipped without affecting the understanding of the rest of this thesis.

Chapter 7 evaluates the proposed techniques and analyzes the errors caused by various factors by using real and synthetic images of an object with ground truth.

Chapter 8 presents preliminary work on estimating a shading function from real images for realistic graphics rendering.

Chapter 9 summarizes this thesis, discusses the strengths and limitations of our approach, and suggests future research directions.

Chapter 2

Previous Research

This chapter is a general overview of surface recovery from brightness in images. The overview also includes some related work in surface recovery by using or integrating other image features with brightness. Surface recovery refers to the process of reconstructing the 3D shape of an object from one or more images. Surface recovery from brightness is based on the fact that the surface reflection is determined by surface geometric and reflectance properties when the illuminant and viewing conditions are fixed. The survey starts with the various surface reflectance models that have been used in computer vision and computer graphics. Techniques for estimating the surface reflection function are presented in Section 2. Theories and techniques for surface recovery from brightness are reviewed in Section 3. An investigation of surface recovery techniques from relative rotation between an object and a camera is given in Section 4. A brief overview of shape from contour methods is presented in Section 5. The strategies and techniques of integrating different image features or different vision modules are viewed in Section 6. A summary of the survey is given in the final section.

2.1 Surface Reflectance Models

The foundation of surface recovery from image brightness is the constraint on the relation between surface geometry and its surface reflectance. This constraint has been formulated for different reflectance models. Three representations can be used to express these reflectance models. First, surface reflectance can be expressed in terms of BRDF [30], which stands for Bidirectional Reflectance Distribution Function. Let the light source direction be (θ_i, ϕ_i) and the viewing direction be (θ_e, ϕ_e) in a local coordinate system specified by polar angle and azimuth. The BRDF, expressed as $f(\theta_i, \phi_i; \theta_e, \phi_e)$, is the ratio of $\delta E(\theta_i, \phi_i)$, the radiance falling on the surface from the direction (θ_i, ϕ_i), to $\delta L(\theta_e, \phi_e)$, the irradiance of the surface as seen from the direction (θ_e, ϕ_e). For an isotropic surface, its surface reflectance at a surface point is invariant to the rotation around the surface normal at the point, so its surface reflectance can be expressed as $\phi(i, e, g)$ in an object centered coordinate system [28]. The incident angle i is the angle between the incident ray and the surface normal, the emergent angle e is the angle between the emergent ray and the surface normal, and the phase angle g is the angle between the incident and emergent rays. When both the illuminant and viewing directions are fixed, the surface reflectance can be expressed as a reflectance map $R(p, q)$, with p and q as the surface slopes in the x and y direction [29, 30]. The BRDF is the most general and powerful. The reflectance function $\phi(i, e, g)$ and the reflectance map $R(p, q)$ can be be calculated from the BRDF [34]. While the reflectance map is the most restrictive, the viewer centered representation makes it amenable to applications in physics-based vision, especially for surface recovery from image brightness.

2.1.1 Lambertian model

The Lambertian reflectance model [30] is the most widely used reflectance model in computer vision. In the Lambertian model, the intensity of the reflected light only depends on the incident direction and the intensity of the light source. The change of the viewing direction will not affect the observed intensity of the reflectance. The reflectance function of a Lambertian surface can be expressed as $E = \frac{\rho}{\pi} \cos i$, where i is the incident angle and ρ is the surface's albedo. The albedo of a surface is the fraction of the light incident on the surface that is reflected over all the directions.

2.1.2 Torrance-Sparrow model and Beckmann-Spizzichino model

The Torrance-Sparrow and Beckmann-Spizzichino models are originally from the optics community and have been extensively used in computer vision and computer graphics to derive other reflectance models. Both describe primarily the phenomenon of specular reflection. In the Beckmann-Spizzichino model [2], the surface height is assumed to be normally distributed, and the light reflectance is described as the reflection of plane electromagnetic waves. The wavelength of the incident light is small compared to the undulation of the surface, and the surface is assumed to be a perfect conductor, such as copper and aluminum. The reflectance function of a metal surface is modeled by two components: a specular component which consists of a broadly distributed lobe with significant specular reflection into many different directions, and a specular spike which is a mirror-like reflection and can only be seen in the specular direction. The magnitudes of the two components change as the surface roughness changes. In the Torrance-Sparrow model [69], the surface is treated as a collection of planar micro-facets, and the light reflection is described as the reflection of directional rays from surface. The model assumes that the wavelength of the light is much smaller than the size of the micro-facets. A specular reflection component is used to represent the direct reflection from the incident direction to the viewing direction by the micro-facets on the surface. The magnitude of the specular reflection is determined by the angle between the viewing direction and the vector which bisects the incident direction and the viewing direction. A diffuse component is used to model the multiple reflection and internal scattering of among the micro-facets on the surface. This diffuse component is modeled as Lambertian. Unlike the Beckmann-Spizzichino model, the Torrance-Sparrow model can predict reflections from both conductors and non-conductors.

2.1.3 Diffuse-reflection model

Subsurface multiple reflection and internal scattering are usually termed diffuse reflection and considered as Lambertian which depends only on the incident direction. By observation of experimental data, Wolff [74, 75, 76] pointed out that for dielectric surfaces, such as surfaces of plastic and ceramic, the Lambertian reflectance model is only valid when both incident angle and emergent angle are less than 50°. Using results of radiative-transfer theory and considering surface boundary effects on subsurface scattering, he derived a new diffuse reflection model for smooth inhomogeneous dielectric surfaces. In his model, the diffuse reflection is a function of both viewing angle and incident angle and the diffuse albedo is calculated from physical parameters of the dielectrical surface.

Oren and Nayar [51, 52] also pointed out that the Lambertian model can prove to be a very inaccurate approximation to the diffuse component. They described a phenomenon in which the brightness value of a rough diffuse surface increases as the viewer approaches the source direction. They presented a different diffuse model for rough diffuse surfaces. In their model, the diffuse reflection also depends on both the incident direction and the viewing direction. Compared to Wolff's diffuse model, their model is not limited to surfaces of dielectric materials and can be applied to both isotropic and anisotropic rough surfaces.

2.1.4 Three component reflection model

Based on the comparison between Torrance-Sparrow model and Beckmann-Spizzichino model, Nayar *et al.* [48] proposed a surface reflectance model of three components: a diffuse lobe, a specular lobe and a specular spike. The diffuse lobe is considered as Lambertian. The specular lobe is expressed by the specular component in the Torrance-Sparrow model. The specular spike is expressed by a modified specular spike in the Beckmann-Spizzichino model. This three component reflection model unifies the Torrance-Sparrow and Beckmann-Spizzichino models and can be applied to a wide class of surfaces.

2.1.5 M-lobed reflectance model

The m-lobed reflectance model [66] was proposed by Tagare and deFigueiredo. Tagare and deFigueiredo observed that most previous reflectance models have “lobes” and these lobes have a common mathematical structure. They formulated these properties and constructed a class of reflectance functions called m-lobed reflectance maps. They define the plane of incident direction and viewing direction as the principal plane and a set of vectors which lie on the principal plane and between the incident direction and the viewing direction as principal vectors. In an m-lobed reflectance map, except for a constant term, each lobe is associated with a principal vector and each lobe is a monotonic function of the angle between the surface normal and the associated principal vector. The multiple lobe reflectance model is not derived from physical principles and it is only a mathematical abstraction. However, this model proved to be useful in the theoretical analysis of non-Lambertian photometric stereo [66].

2.1.6 Reflectance map

The reflectance map was introduced by Horn [29, 30]. It relates the orientation of the surface normal to the image intensity caused by that orientation. A surface z is expressed as a height function on the image plane in coordinates x, y. The surface orientation is determined by the gradient (p, q) with $p = \partial z/\partial x$ and $q = \partial z/\partial y$. When the surface type and illuminant configuration are given, the image intensity is a function of p and q. The function is called reflectance map and written as $I = R(p, q)$. The reflectance map can be calculated from the BRDF and the surface reflectance function $\phi(i, e, g)$ [28, 34, 29, 79, 30].

2.1.7 Reflectance models in computer graphics

In computer graphics, a number of reflectance models have been developed for rendering a surface to achieve visual reality [19]. One of the widely used model is the Phong-Blinn model of three components: the ambient component, diffuse component, and specular component. The ambient component is the reflectance caused by illumination from the light reflected from surfaces present in the environment. The diffuse component is Lambertian. The specular component is represented by the Torrence-Sparrow model. An alternative for the specular component is Phong's model [19] in which the specular reflection is approximated by $\cos^n \alpha$. The angle α is the angle between the viewing direction and specular direction. The exponent n is the roughness of the surface. The above models are based on hueristic approaches. Some physics-based reflectance models [16, 25] have been developed for realistic rendering.

2.2 Estimating the Surface Reflectance Function

The BRDF of a surface can be determined experimentally by using goniometers. A goniometer has two axes of rotation so that the device or a sample surface mounted on it can be accurately adjusted in any direction. This procedure is tedious and impractical [30] since at least three variables are involved. The surface reflectance can also be calculated from physical parameters of the surface material. However, the physical parameters of a surface material are generally not available. Even if these parameters are known, the calculation is still very complicated since the reflectance properties of a surface are also related to the properties of the light source such as polarization and wave length. The reflectance function can also be measured or computed from the images of a surface. In general, precisely calculating the reflectance function of a surface under arbitrary illuminant and viewing directions is impossible and sometimes unnecessary. For example, in photometric stereo, the surface reflectance maps obtained under a fixed viewing direction with three different illuminant directions are sufficient for extracting a surface shape.

2.2.1 Estimating reflectance parameters from one image

Under the assumption of Lambertian reflectance or Lambertian reflectance plus a constant component, it is possible to estimate jointly the composite albedo (product of albedo and the illuminant intensity) and the constant component with surface orientation from just one image. Zheng and Chellappa [86] showed that if the surface normals have a certain kind of distribution, the composite Lambertian albedo and the constant component as well as the illuminant direction of the light source can be derived from the first and the second statistical moments of the brightness values in one image. Lee and Rosenfeld [42] showed that after the tilt and slant at a surface point have been found, the composite albedo at a surface point can be calculated from its image intensity value and the image intensity values of its two opposite neighbor points in the gradient direction. In [8], Brooks and Horn describe a regularization method which estimates the surface orientation and the composite albedo at the same time in an iterative computation. Under the spherical point (point on a sphere) approximation and Lambertian reflection, Pentland [54] proved that the composite Lambertian albedo, surface orientation, illuminant direction and curvature can be calculated from the image intensity and the first and second directives of the image intensity at a surface point.

Estimating the surface reflectance from one image only applies to surface of simple reflectance properties. Other restrictive assumptions have to be made and the results are usually not accurate.

2.2.2 Joint estimation using photometric stereo images

Some techniques attempt to avoid the calibration process in Woodham's photometric stereo. A common point among these techniques is that they assume that the surface reflectance function has some particular form with undetermined parameters and these parameters are jointly estimated with surface orientation. Coleman and Jain [14] developed four light-source photometric stereo to extract the shape of textured and specular surfaces. In their work, they assume that the textured surface has Lambertian reflectance $E = R\cos i$ with varying reflectance factor R, and four light sources are distributed in a way so that at any surface point, no more than one light source would cause a specular reflection. From every triple of the four intensity values of a surface point, a value for the reflectance factor is calculated and a relative deviation of the four values for the reflectance factor is computed. If the relative deviation is smaller than a threshold, a specular component is not considered to be present and the reflectance factor is computed as the average of the four. If the relative deviation is greater than the threshold, the reflectance factor is considered as the smallest of the four.

Solomon and Ikeuchi [60] also use four light source photometric stereo. They assume a simplified Torrance-Sparrow reflectance model for a uniform surface. They divide the surface into regions illuminated by different numbers of light sources. In the four light source illuminated area, they use a method, which is similar to that of Coleman and Jain [14], to compute the Lambertian albedo and the surface normal and to segment image pixels into Lambertian and specular. They propagate the results to the area illuminated by three light sources and then to the area illuminated by two light sources to compute surface normals and segment the image pixels. To estimate the specular roughness and the specular strength, an optimization procedure is called to search for the values of the reflectance parameters which fit the brightness values of the specular image pixels into a simplified Torrance-Sparrow reflectance model.

Tagare and deFigueiredo [65] use eight light sources and an a priori estimate of parameters of a reflectance function with Lambertian and specular components to compute the reflectance function and the surface orientation by using regularization.

Estimating the surface reflectance from photometric stereo images can deal with more complicated surface reflectance and obtain more accurate results than methods that rely on a single image.

2.2.3 Photometric sampling

Photometric sampling [50] is a method for obtaining shape and reflectance from a large number of images. Each image is taken with a different position of an extented light source. A hybrid reflectance model of a Lambertian component and a specular spike component is assumed and the image intensity at a surface point is sampled at a minimum sampling frequency determined by the angle width of specular component of the extended source radiance function. The extraction of the surface orientation, the strength of the Lambertian component and the strength of the specular component are all based on two constraints: the unique surface orientation constraint and minimum sampling frequency constraint. The algorithm tries to separate the intensity values containing only a Lambertian component from the intensity values containing both Lambertian and specular components. The intensity values containing only a Lambertian component are used to compute the surface normal, and the Lambertian strength by least-squares fitting. Then the Lambertian intensity component is deducted from pixels and the resulting intensity values are used to compute the specular strength and the surface normal.

Photometric sampling is capable of estimating complicated surface reflectance functions. However it uses more complicated experimental setting and more images than the other methods.

2.2.4 Measuring the reflectance map by calibration

Photometric stereo was first introduced by Woodham and later on developed by a number of researchers. In Woodham's photometric stereo, the reflectance function is assumed to be known [78] or obtained from a calibration procedure [80, 82]. A calibration object, of known shape and with the same reflectance properties as the object to be recovered, is used to measure the reflectance function of the surface material. Three images of the calibration object are taken under the same viewing and illuminant conditions as that for the object to be recovered. Since the shape of the calibration object is known, at each image pixel, the corresponding surface gradient (p, q) can be calculated and the relation between gradient and image intensity value can be determined. Because the reflectance map is measured empirically from images and no assumptions on the reflectance of the surface are made, the reflectance function obtained is accurate and the technique can be applied to a wide class of surfaces of uniform reflectance.

2.2.5 Estimating reflectance from range and brightness images

Ikeuchi and Sato [37] use a range image given by a range finder and a brightness image given by a TV camera to estimate the reflectance property of an object surface. They assume a simplified Torrance-Sparrow reflectance model, and they extract Lambertian strength, specular strength, and specular sharpness, as well as light source direction. Since the surface normal can be calculated from the range image and the number of the reflectance parameters is far smaller than the size of the brightness image, least-squares fitting is used to estimate the parameters. In the first stage, the light source direction and the Lambertian strength are calculated by an iterative least-squares fitting method. During the iteration, specular and interreflection pixels are discarded by comparing the observed intensity value with the intensity value calculated with the estimated parameters at each pixel. Therefore only Lambertian pixels are picked for determining the Lambertian strength and light source direction in the final iteration. With the estimated light source direction and surface orientation, the specular pixels are extracted from the brightness image. With these specular pixels, the strength and sharpness of the specular reflectance are also estimated by an iterative least-squares fitting method. Ikeuchi and Sato's method requires both a range image and a brightness image and works only on a uniform surface.

2.3 Shape Recovery from Image Brightness or Shading

The reflectance models provide a constraint on the relation between surface orientation and image brightness. Since surface orientation has two degrees of freedom, the surface orientation at a surface point is not fully constrained by its image brightness in one image. Additional assumptions or more images must be introduced to make the surface orientation fully constrainted. Sometimes, special consideration have to be made for coping with image noise and making the computation on the surface orientation robust and accurate. In this section, we review different approaches to surface recovery from image brightness or shading, the variation of image brightness.

2.3.1 Horn's shape from shading

The characteristic strips method developed by Horn [27] was the first algorithm for general shape from shading. In this approach, the problem is described by a set of five coupled ordinary differential equations and then solved along a set of curves with the initial starting values. Horn and Brooks [32] also developed a parallel and iterative method which is based on a variational principle. The constraints on the irradiance equation, surface smoothness and integrability are reflected in a cost function which is a functional of surface orientation. The Euler equation which minimizes the integral of the functional over the object image is found and solved with boundary conditions. The solution of the Euler equation gives a surface gradient field which best matches the image intensity values. Later on, Horn [31] refined and improved the variational approach. Using the sum of the squared differences between the first derivatives of surface height and the estimates for surface gradients as the integrability constraint, he derived a coupled system of second-order partial differential equations on surface gradient and surface height. Thus the solution of this system gives an estimation of surface gradient and surface height at the same time. In order to get a fast and correct solution, Horn introduced an adaptive regularization parameter for the smoothness departure term and used a linearization of reflection map.

2.3.2 Other global shape from shading techniques

In recent years, other global shape from shading techniques have been developed. Leclerc and Bobick [41] formulated the cost function directly with respect to the surface height on a regular grid. In the cost function, the first and second finite difference of surface height is used to approximate the surface gradient and the partial derivatives of the surface gradient. The problem then can be considered as minimizing the cost function with respect to the surface height at each grid point. The conjugate gradient method is used to find an optimal solution. They pointed out that their algorithm can fuse stereo data. Lee and Kuo [43] also directly use the surface height in the cost function. They use a triangular mesh surface model with a linearized reflectance map to directly relate image intensity with the surface height at the three vertices of a triangular surface element. The minimization of the cost function is turned into the solution of a system of linear equations. Using an iterative method, the linear approximation of the reflectance map and surface height are updated until a stable state is achieved.

Dupuis and Oliensis [17] adapted the dynamic programming approach in optimal control to the shape from shading problem. They showed that the surface shape within an image region containing a singular point can be recovered without using regularization, and that the algorithm is fast and provably convergent. Based on Dupuis and Oliensis's work, Bichsel and Pentland [3] developed a simpler algorithm. The algorithm is described by a minimum downhill principle which reveals the direct connection between their algorithm and Horn's characteristic strip approach. They reported that the algorithm is fast and converges to the right solution in fewer than 10 iterations.

Vega and Yang [71] presented a heuristic approach to recover shape from shading. The heuristics are derived based on the geometrical interpretation of the Brooks and Horn's algorithm [8]. They use weighted neighbor operations to enforce a smoothness constraint, a minor slant adjustment operation to minize the brightness error, and a normal complementary operation to change the convexity of the solution to avoid local minima. They claim that although their algorithm is heuristically based, it has better performance than Brooks and Horn's original algorithm.

Pentland [55] developed an algorithm based on the linearization of the reflectance map. He shows that under some conditions, the reflectance map on an image region can be approximated by a linear function, and through Fourier transform, the surface height can be directly recovered on the image region.

2.3.3 Local shape from shading

Local shape from shading was first formulated by Pentland [54]. Under the assumption that a surface point can be approximated as a point on a sphere and that the surface reflectance is Lambertian, he derived a set of equations which uses the image intensity and the first and second directives of image intensity at a surface point to estimate the surface normal at that point. Since the first and second directives of the image intensity are not reliable, the estimation can be very inaccurate as pointed out by Andrew Blake [5]. In an improved local shading method, [42], under the same assumption made by Pentland but by using the light source coordinate system, Lee and Rosenfeld proved that the surface normal can be estimated from the first order derivatives of image intensity. This leads a more robust surface orientation estimator than that using the first and second derivatives of image intensity.

2.3.4 Photometric stereo

Shape from shading using one image is usually an ill-posed problem. To solve the problem, additional constraints must be applied and sophisticated mathematical methods must be used. In photometric stereo, multiple images taken under different illuminant directions are used to make the problem overdetermined and only simple computation is used. The basic idea of photometric stereo was first described by Woodham [78, 79], and first implemented by Silver [59]. Multiple images of an object are taken under different illuminant directions with a fixed viewing direction. Assuming that the reflectance function of the object surface is known, two images of a surface are sufficient to determine the surface orientation at each surface point. However, in general, since the reflectance function is nonlinear and more than one solution is possible, a third image is used to overdetermine the surface orientation. Woodham proved that for a Lambertian surface three images taken under three non-coplanar light sources are necessary and sufficient to determine surface orientation and albedo at each surface point. In the implementation of photometric stereo [59, 80, 82], a calibration procedure is used to obtain the reflectance of the surface from images. Using photometric stereo, Woodham shows that intrinsic surface curvature can also be locally estimated from the local intensity derivatives [80]. First the Hessian matrix at a surface point is estimated from intensity derivatives of this point in three images. Then from the Hessian matrix and the surface normal, the principal curvatures and their associated directions are determined. Recently, Woodham described a near-real-time implementation of photometric stereo which includes confidence measurements for local gradient and curvature estimate [82]. He showed that triples of intensity values of a uniform surface are confined on a 2D surface in the 3D space of intensity values. Because of interreflection and cast shadows, the measured triple of intensity values may not lie on the 2D surface. The distance between the measured triple and the 2D surface is used as a confidence measure for surface orientation and curvature. The near-real-time implementation of photometric stereo makes it appealing for practical applications.

Color photometric stereo [10] recovers the shape of a colored object from two or more color images of the object. The method is based on the fact that color provides more constraints on the surface orientation than grey level brightness. As a consequence, the surface shape can be recovered more accurately.

2.3.5 Variations of photometric stereo

Since Woodham's photometric stereo was introduced, variations of photometric stereo have been developed. Tagare and deFigueiredo [66] conducted a theoretical study on photometric stereo with an m-lobed reflectance function model. They proved that only three light sources are needed to derive surface orientation uniquely under the m-lobed reflectance model. They also discussed the problem of how many light sources are needed to derive a complete surface orientation on a hemispherical surface in the general case. In the work described in another paper [65], they used a 3-lobed reflectance function and regularization to recover surface orientation and albedo simultaneously. Four light source photometric stereo techniques [15, 60] for jointly estimating the surface reflectance function and surface orientation have been discussed in the previous section, so we will not talk about them here again.

Hayakawa [24] uses an arbitrary moving light source to illuminate a Lambertian surface with varying albedo. The computation is based on factorizing an image intensity matrix of multiple pixels into two parts: one is the product of surface albedos and surface normals and the other is the product of light intensities and light source directions. The surface normals, albedos, light strength and light directions are obtained from the fact that surface normals and light source directions are unit vectors. Iwahori *et al.* [39] illuminate a uniform Lambertian surface with a point light source undergoing controlled motion in 3D space. Using perspective projection and inverse square law for illuminance, they recover surface orientation as well as absolute depth of an object. Like Iwahori *et al.*, Clark [13] also uses a moving light source to recover the absolute depth, but he assumes a general surface reflectance function. The light source moves around a small area, and the image intensities and the derivatives of the image intensities with respect to the changes in light position are used to compute depth at each surface point. To get an accurate estimate of depth, seven images and a least-squares estimate are used.

2.4 Shape from Rotation

To observe different aspects of an object for surface recovery, we can change either the camera viewing direction or the object orientation. To change camera direction, we usually have to use some calibration technique to determine the precise location of the camera. In most cases, fixing the camera and controlling the orientation of the object is easier. Thus it is natural to consider rotating an object in front of a camera. In photometric stereo, in order to recover the surface orientation of the whole object, Woodham puts the object on a rotary table [82]. In Szeliski's shape from rotation [61, 64], Szeliski place the object on a spring-wound microwave turntable. The turntable is taped with a position encoding grid on its side so that its rotation angle can be detected from the image sequence. In Szeliski's work, only optical flow or contour is used to derive the 3D model of the surface. In Zheng's 3D model acquisition [83], Zheng did some experiments on both plaster models and real people. For plaster models, he put the object on a turntable controlled by a computer. For real people, he put the subject on a swivel chair and attach some markers to the subject's shoulder to estimate the rotation angle. In his work he used only contours to recover the surface shape. Zheng [84] also has considered shape recovery from the highlight of a rotating object. His technique works well a specular surface. Boyer [7] also uses a turntable to rotate an object and extracts the contours in images for surface recovery. Although there are different image features available, such as brightness, contour, optical flow and texture, from the images of a rotating object, the works mentioned above just used one of them.

2.5 Shape from Contour

When an object is viewed from different directions, contours in images directly reflect the geometry of the object. A mathematical approach to shape from contour is described by Giblin and Weiss [23]. The object is modeled as the envelope of its tangents from its profiles. An algorithm is used to reconstruct the object and compute surface curvature from profiles under the assumption that the viewing directions are in a plane. Some experiments on planar curves verified their algorithm. Like Giblin and Weiss, Zheng [83] also assumes that the viewing directions are in a plane. Zheng pointed out that some regions, such as concave regions, can not be exposed as contours in images. He presented a method for detecting and locating the unexposed regions and extended this method to both orthographic and perspective projection.

Unlike Giblin and Weiss, Vaillant and Faugeras [70] assume more general camera motion. Under arbitrary change of the camera viewing direction, they derived a set of equations for computing the 3D shape as well as curvatures from occluding contours (profiles). They model the object as the envelope of its tangents and use Gauss map to compute surface depth and curvature. They showed some experiments on calculating principal curvatures and directions and reconstructing local surfaces.

Cipolla and Blake [11] use epipolar parametrization for surface recovery from occluding contours in the image sequence. They assume arbitrary nonplanar and curvilinear camera motion under perspective projection. They derived a set of formulas for computing depth, surface orientation and surface curvature from contour motion in three images taken with slightly different view positions. Since computing surface curvature involves the first and second order differential quantities of 2D location of the contours and these quantities are usually not reliable, they use the difference of the image motions of a pair of surface points; one is a surface marking and the other is a point on an occluding boundary. In the implementation they use B-spline snakes for image contours, and use a parabolic curve to fit contour points tracked over epipolar planes. They demonstrated the power of this technique for robot application in circumnavigating curved obstacles and picking up a curved surface object under visual guidance. Boyer [7] extended Zheng [83], Cipolla and Blake [11]'s work and used a regularization method to get a smooth surface from image contours of a rotating object.

Szeliski and Weiss [64] also use epipolar planes for surface recovery from contours in image sequence. But they are mainly concerned with 3D surface location. They derived a very simple linear equation for fitting a circular arc to the points on the extremal boundary points tracked over epipolar planes. Then they use different techniques such as least-squares estimate, Kalman filter and linear smoother (a generalization of the Kalman filter to update previous estimates) for fitting piecewise circular arcs to the surface by using contours in an image sequence of a rotating object. They showed some expirements with both synthetic and real images.

There is an inevitable problem in all shape from contour methods. It is that the object regions which are not exposed as contours in images can not be recovered. Therefore other image features have to be used to recover these regions.

2.6 Integration of Vision Cues and Modules

When an object is moving or rotating relative to a camera, there are different vision cues available, such as brightness, texture, flow, and contour. Some vision modules just use one of above vision cues and work under some specific assumptions. It is possible to integrate different vision cues or different vision modules to achieve more reliable and robust solutions. Woodham [78, 79] compared photometric stereo with binocular stereo and pointed out they can be considered complementary to each other. Thus it is natural to consider integrating photometric stereo and binocular stereo. Indeed some work has been done on this subject. Here we give a general review on integrating different vision cues and integrating different vision models.

2.6.1 Integrating photometric stereo with binocular stereo

Ikeuchi [36] combined photometric stereo with binocular stereo by using a dual photometric stereo system. Instead of one camera in the conventional photometric stereo, there are two cameras with different viewing directions in the system. Two sets of photometric images with different viewing directions are used to produce a pair of surface orientation maps. The surface orientation maps are segmented into different regions. Using epipolar, area, and surface orientation constraints, region matching is done from the coarse level to fine level between the two orientation maps. The disparity of the mass center of corresponding regions determines the absolute depth. The experiment demonstrates that the system can work in a scene of several objects.

Lee and Brady [44] reported a system which helps physicians to monitor the development of glaucoma, a complex eye disease. In the first stage, the system uses a pair of binocular cameras to get a pair of stereo images of the optic disk. The edge-based stereo algorithm is used to match blood vessels and the edge of the optic disk to get a sparse depth estimation. In the next stage, a pair of photometric images of the optic disk is taken under different illuminant directions. Under the Lambertian surface assumption and the coplanar light source and viewing direction, the photometric image pair is used to compute the surface gradient component p at every point. The field of the gradient component p is smoothed by regularization and then is integrated, starting from a profile supplied by binocular stereo, to get a dense depth map. An elastic plate model is used to combine the two estimates obtained from photometric stereo and binocular stereo. The initial shape is determined by photometric stereo and then is deformed so that the corresponding surface points conform to the depths given by binocular stereo.

Recently Wolff and Angelopoulou [77] present a new method for depth recovery by using the photometric stereo ratio in two sets of photometric images taken by a binocular camera. The key point in their method is that the surface reflectance function can be factorized into two different parts: one is the function of the incident angle only and the other is the function of the emergent angle only. So the photometric ratio, the ratio of the brightness value of a surface point illuminated by a light source in one direction to the brightness value of the surface point illuminanted by the light source in another direction, is an invariant with respect to the viewing direction. Wolff [75] derived a surface reflectance function which can be factorized into the two different parts for dielectric surface and verified that the model is more accurate than Lambert's law. Since the photometric ratio is viewing direction invariant, the ratio can serve as surface marks for searching for stereo correspondence. They report that in their experiments the surface depth (the distance from the surface to the camera) can be recovered within 1% accuracy.

2.6.2 Integrating binocular stereo with shading

Blake *et al.* [5] studied the uniqueness of the solution when stereo is supplied with analysis of shading in a special case. They discussed the construction of a C^2 surface patch with a continuous boundary obtained from binocular stereo. If the boundary is piecewise differentiable and the normal component of surface orientation on the boundary is not equal to zero, if there is at most one singular point in the image of the patch, and if the surface can be recovered, the recovered surface is unique up to an inversion.

Hougen and Ahuja [35] integrate shading and stereo at the module level. Their system consists of several modules. An initial depth map is obtained from a pair of stereo images by a stereo module using an area-comparison based method. The regions of uniform albedo in one of the images are segmented by a segmentation module and the albedo for each region is computed by an albedo module. The light source direction is estimated by a source module with the initial estimate on the depth and the surface albedo. The final surface shape is refined by a shape from shading module which uses Leclerc and Bobick's method [41].

Chiaradia *et al.* [9] developed an approach to surface recovery by integrating stereo and shading. A sparse depth map is obtained by edge-based binocular stereo technique. A local shading analysis method is applied to one of the two images to get an estimation of surface orientation in terms of a raw needle map. To integrate the sparse depth map with the raw needle map, the raw needle map is first segmented into connected regions corresponding to continuous surfaces of the object. For each segmented region, by using the known sparse depth values within that region, the surface is interpolated by polynomial approximation. This interplation process will give a good approximation under the assumption that each region contains enough points of known depth.

2.6.3 Integrating multi-cues and multi-modules

Markov Random Fields (MRFs) can be used as a scheme for integrating different vision cues or modules. Poggio *et al.* [56] implemented a parallel integration of vision modules based on coupled MRF lattices on the Connection Machine. In this approach, the assumptions and constraints on different vision cues can be embedded into an *a priori* probability distribution of the field. A discontinuity is handled by explicitly adding a line process to MRF field. In their implementation, several early vision modules including depth, motion, texture and color, are coupled to discontinuity in surface properties. The problem then is to find the maximum of the posterior distribution which is the best estimate of the surface for the input data. The solution can be obtained by means of Monte Carlo techniques.

Fua and Leclerc [22] integrate different information cues for surface recovery by using a 3D triangular mesh. The framework of their approach is combining different constraints on the surface into a cost function. The cost function is a weighted sum of several components corresponding to the different constraints. The minimization process searches for surface shape which satisfies the constraints by deforming the 3D mesh. Information cues such as silhouettes, 3D points, 3D lines, stereo, shading, can be integrated in the cost function. Since the whole process is an optimization process, the method works well when a good initial estimation of the surface shape is available.

Zheng [85] proposed the problem of combining different visual cues into a 3D model. He addressed the problem by analyzing an image sequence of a rotating object. He described how to classify different cues, such as contour, edge, flow, shadow and highlight in the image sequence so that different methods can be applied to different cues. He also described the strategy for selecting different vision modules and combining the estimation obtained by the different modules. Finally, he pointed out that the only way to recover the shape in a concave area without distinct image features is using a shape from shading method.

Pankanti *et al.* [53] discussed an implementation of the integration of different vision modules in the context of the quality of surface reconstruction. The models used in the reconstruction are perceptual organization, stereo, shape from shading and line labeling. In their integration scheme, the interaction between shape from shading and stereo modules is a two-way and tightly coupled interaction. One depth map is determined by linear interpolation of the sparse depth obtained from stereo module. The other depth map is integrated from surface normals produced by shape from shading module. A corrected depth map is computed by summing up the two depth maps with different weights. Then the corrected map is passed to both modules to compute a refined sparse depth and surface normals. The interactions between other modules are one-way and straight forward.

Clark and Yuille [12] took a constraint based approach to integrate various information sources obtained from sensory data. They considered integrating information sources (or data fusion in their literature) as the embedding of multiple constraints on information sources into a solution process. They described two methods, a Bayesian method and a regularization method as well as their incorporation strategy for embedding the constraints. They presented a group of examples to show how the constraint based approach is used in solving problems in computational vision. Their approach provides a mathematical foundation on which integration algorithms can be constructed and analyzed.

2.7 Summary

A brief survey on surface reflectance extraction and shape recovery from images has been presented. Extracting the surface reflectance function from images is not trivial. The surface reflectance function is usually assumed to be Lambertian or have a particular form with undetermined parameters. This is not true in the general case. So far the most simple and robust method for extracting the surface reflectance function is using a calibration object of known shape. The method requires that the calibration object have the same surface reflectance as the object to be recovered and the calibration object be imaged under the same conditions as the object to be recovered.

In general, shape from shading information in one image is an under-constrained problem. The solution to this problem is usually computationally intensive, non-robust and non-unique [33, 57]. Local shape from shading relies on the assumption that a surface patch can be approximated as a spherical surface and uses the derivatives of image brightness. The results obtained can be inaccurate. Photometric stereo is the first technique using image brightness in multiple images for surface recovery. The computation in this technique is simple and robust. However it requires a calibration procedure to obtain the surface reflectance function and works on objects of uniform reflectance. The variations of photometric stereo try to overcome these drawbacks. Some of them use optimization or regularization to estimate jointly the surface reflectance function and surface orientation. Thus they lose the essence of the computational simplicity and robustness in the original photometric stereo.

While photometric stereo fixes the camera and the object but changes the illuminant direction, other techniques fix the illuminant and viewing directions but change the orientation of the object. A commonly used scheme is rotating an object in front of a fixed camera. In this case, there are several image features such as contour, brightness, stereo, texture and optical flow available for surface recovery. The technique using contour can only obtain a surface of limited resolution and can not recover concave regions. The technique using flow or stereo requires surface marks on the object surface and cannot work on uniform surfaces. So far, no other work on integrating several information cues from the image sequence of a rotating object has been reported.

Chapter 3

Geometric and Photometric Constraints

One of the primary tasks in computer vision is to find a set of constraints that allow a computer to extract useful information from images unambiguously. Features in images provide constraints on geometric and photometric properties of the object surfaces in the scene. Knowledge and assumptions of the objects in the scene as well as the imaging conditions provide other constraints on how the image features are generated. These constraints are not independent. Exploring and integrating these constraints yield solutions for vision problems. In a controllable environment, by properly setting the imaging condition and controlling the object motion, the constraints as well as the relations among these constraints are made simple. However, even in a controlled environment, identifying the constraints and finding the relations among the constraints are not trivial. This chapter describes the constraints we used in our photogeometric and wire-frame techniques.

Section 3.1 of this chapter describes the imaging apparatus and assumptions. Section 3.2 investigates geometric constraints, which are mainly related to the geometric property of the object in the scene. Section 3.3 explores photometric constraints, which are mainly related to the photometric property of the object in the scene. Section 3.4 gives a summary of this chapter. The relations among these constraints will not be described in a separate section since they are introduced in the descriptions of the constraints or used in the following chapters.

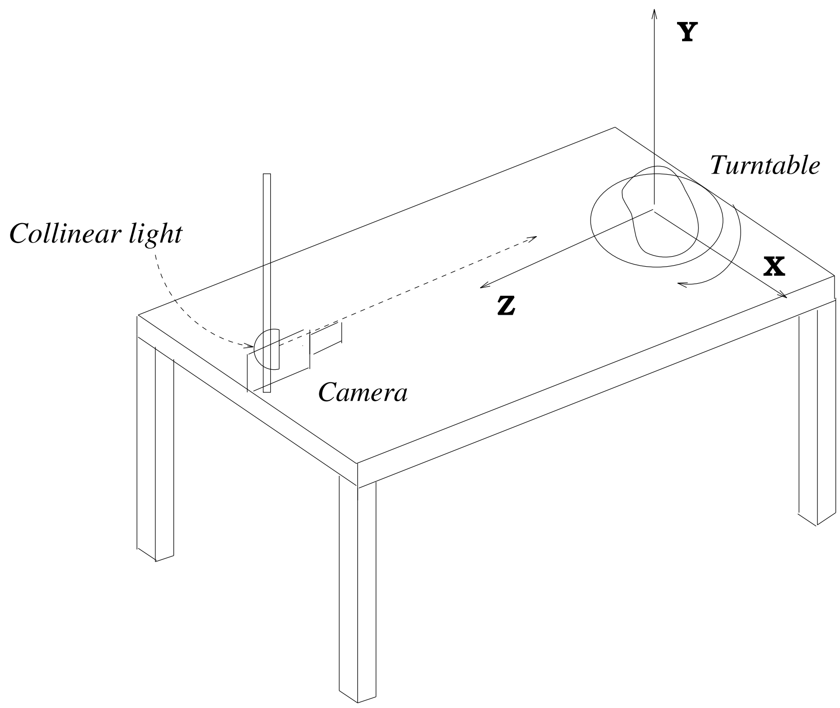


Figure 3.10: Imaging Apparatus. The light source direction and viewing direction are collinear.

3.1 Imaging Apparatus and Assumptions

The imaging apparatus is shown in Fig. 3.10. The object is on a turntable whose rotation angle can be precisely controlled. The Y axis coincides with the rotation axis of the turntable. A parallel projected light source is used to illuminate the object. It gives uniform radiance over time and the illuminated area. Both the illuminant direction of the light source and the viewing direction of the camera are aligned with the Z axis. Since the illuminant direction and the viewing direction are collinear, the light source is called a *collinear light source*. As the camera is far away relative to the size of the object, the projection from the object on the image plane is assumed to be orthographic. The origin of the coordinate system is projected on the center of an image. Images are taken when the object rotates around the Y axis in the direction from the X axis to the Z

axis. The projection of the rotation axis of the object in the images can be determined by a very simple method (see Section 6.5). The actions of taking images and rotating the object are synchronized by a computer program.

The surface is assumed to have the form $z = f(x, y)$ and to be C^2. The surface reflectance function is isotropic, uniform, and strictly monotonic. It has the maximum brightness value when the surface normal is in the viewing direction. These assumptions are true for most opaque surfaces. The surface shape is generic so that there is at least one point of surface normal parallel to the viewing direction. This assumption is necessary for extracting surface reflectance function from images(see Section 4.2). The whole surface is enclosed in each image taken during object rotation. Another assumption is that surface has low curvature, especially in the x direction.

3.2 Geometric Constraints

3.2.1 Rotational transformation

Since the surface is C^2, its second partial derivatives exist and are continuous. Since the coordinate system is defined by the viewing direction and the rotation axis, a 3D point on a surface changes its coordinates as well as its surface orientation during object rotation. The rotation angle of an object is described relative to an initial state of the object in which the rotation angle of the object is zero. In the general case, when we talk about the 3D coordinate and surface orientation without specifying the rotation angle, we refer to the initial state. In this thesis, we use the rotation angle as a subscript to denote the values of all variables after the object has been rotated by that angle. For example, (x_0, y_0, z_0) and $(x_\alpha, y_\alpha, z_\alpha)$ are the coordinates of a 3D point on an object before and after the object has been rotated by α; $image_0$ and $image_\alpha$ are the images taken before and after the object has been rotated by α. Let (x_0, y_0, z_0) be a 3D point on a surface. After the object has been rotated by α, the coordinate $(x_\alpha, y_\alpha, z_\alpha)$ of this 3D point is

$$x_\alpha = x_0 \cos\alpha - z_0 \sin\alpha, \tag{3.1}$$

$$y_\alpha = y_0, \tag{3.2}$$

$$z_\alpha = x_0 \sin\alpha + z_0 \cos\alpha. \tag{3.3}$$

The surface orientation is defined as $(-p, -q, 1)$ with $p = \frac{\partial z(x,y)}{\partial x}$ and $q = \frac{\partial z(x,y)}{\partial y}$. Thus the unit surface normal is $\hat{\mathbf{n}} = \frac{(-p,-q,1)}{\sqrt{(p^2+q^2+1)}}$ when $p \neq \pm\infty$ and $q \neq \pm\infty$. Let $(-p_0, -q_0, 1)$ be the surface orientation of a 3D point. Considering the surface orientation at the 3D point as a vector in 3D space, after the object has been rotated by α, the three components of the vector are

$$\begin{aligned} -u &= -p_0 \cos\alpha - \sin\alpha, \\ -v &= -q_0, \\ w &= \cos\alpha - p_0 \sin\alpha, \end{aligned}$$

respectively. Scaling the three components to make the component in the z direction be 1, the surface orientation after α rotation of the object is $(-p_\alpha, -q_\alpha, 1)$ with

$$p_\alpha = \frac{p_0 \cos\alpha + \sin\alpha}{\cos\alpha - p_0 \sin\alpha}, \tag{3.4}$$

$$q_\alpha = \frac{q_0}{\cos\alpha - p_0 \sin\alpha}. \tag{3.5}$$

These two equations describe the surface orientation changes during object rotation. Since the brightness value of a surface point is determined by its surface orientation, these equations also provide constraints on the brightness changes during object rotation. In surface recovery, the depth, the z coordinate, and the surface orientation are recovered at every point in an image.

3.2.2 Image correspondence

Since orthographic projection is used, a 3D point of coordinates (x, y, z) is projected on an image plane with coordinate (x, y). For convenience, the coordinates for both 3D and image points are measured in image pixels. In the case of no ambiguity, when we talk about surface orientation at image point (x, y) we mean the surface orientation at point (x, y, z), and when we talk about the depth at image point (x, y) we mean the z coordinate at point (x, y, z). A 3D surface point on an object surface will appear in several images taken during object rotation. Let (x_0, y_0, z_0) be a 3D surface point d. In the images taken after the object has been rotated by α_0, α_1, ..., α_n, the image locations of d will be $(x_{\alpha_0}, y_{\alpha_0})$, $(x_{\alpha_1}, y_{\alpha_1})$, ..., $(x_{\alpha_n}, y_{\alpha_n})$ and the locations can be computed by using Equations 3.1–3.3 if the initial coordinate (x_0, y_0, z_0) of d is known. Since rotating the object in front of a fixed camera is equivalent to rotating the camera around a fixed object, the epipolar and disparity constraints used in conventional stereo still apply to the images taken during object rotation. As the rotation is around the y axis and the image projection is orthographic, all the image points of a 3D surface point will have the same y coordinate. In our imaging condition, the epipolar and disparity constraints are simpler than those used in the general stereo techniques.

3.2.3 Occluding contour

An occluding boundary (contour generator, extremal boundary, rim) consists of a set of surface points which are tangent to the viewing direction. For a generic smooth object, the occluding boundary is a part of a smooth 3D curve on the surface. The projection of an occluding boundary on the image plane is an occluding contour (apparent contour, extremal contour, silhouette). Occluding contours in images give rich information on the geometry at the corresponding occluding boundaries.

At a surface point on an occluding boundary, $p = \pm\infty$ and $q = \pm\infty$. The surface normal at the surface point can be directly determined from the corresponding occluding contour in the image. Under orthographic projection, the surface normal is perpendicular to the viewing direction and to the tangent at the corresponding point on the occluding contour. The x and y coordinates of an occluding boundary can be directly determined from its occluding contour. However, the z coordinates on an occluding boundary are lost under the projection from 3D to 2D.

There are two kinds of occluding contours in images: background-occluding contours and self-occluding contours. A background-occluding contour is caused by the occlusion of the object surface against the background. Since a black curtain is used as the background in our experiment, the brightness of the background is dark and uniform. A background-occluding contour can be extracted by thresholding the background brightness. A self-occluding contour indicates that one part of the object surface is occluded by another part of the object surface. Since both the occluded part and the occluding part are brighter than the background, a self-occlusion contour cannot be extracted simply by thresholding the background brightness, but it can be extracted by using the Canny edge detector. Figure 3.12 shows the background-occluding contour on the left side and the self-occluding contour on the handle in an image of a cup (see Fig. 3.11). The protrusion and colors on the right side of the cup violates our smooth and uniform surface assumption so that the occluding contour on the right side cannot be detected as a continuous curve. The self-occluding contour is detected by using the Canny edge detector with $\sigma = 2$ and threshold 20 for gradient. The self-occluding contour caused by the handle is well detected and can be seen as connected edge points. Sometimes, a segment of an occluding contour is a background-occluding contour and the other segment is a self-occluding contour. So both thresholding and edge detection have to be used. Then the different segments of an occluding contour are connected by using the

Figure 3.11: An image of a cup.

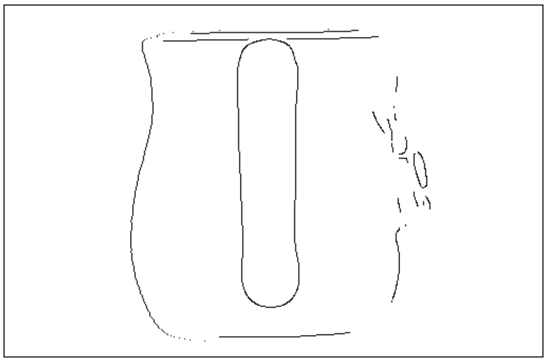


Figure 3.12: Occluding contours obtained by thresholding and edge detection.

continuous surface assumption.

3.2.4 $p = 0$ curve

Under orthographic projection, the surface orientation of a 3D point of $p = 0$ ($\partial z/\partial x = 0$) is parallel to the plane of the viewing direction and the rotation axis. For a C^2 surface, the points of $p = 0$ consists of a smooth 3D curve. We define the curve as a $p = 0$ *curve*. The image of a $p = 0$ curve on a smooth surface does not give any feature which can be used to determine the location of the curve.

Although there is no distinguishing image feature for a $p = 0$ curve, after $\pi/2$ or $-\pi/2$ rotation, a $p = 0$ curve will become an occluding boundary and the projection of the curve will be an occluding contour if the curve is not occluded by the other part of the object after the rotation. Since occluding contours cause sharp brightness changes, they can be detected and located in images.

Let a 3D surface point (x_0, y_0, z_0) have surface orientation $(-p_0, -q_0, 1)$ with $p_0 = 0$. From Equation 3.1–3.3, after $\pi/2$ rotation of the object, we have

$$x_{\pi/2} = -z_0, \quad y_{\pi/2} = y_0, \quad z_{\pi/2} = x_0, \quad \& \quad \frac{dx}{dy}(x_{\pi/2}, y_{\pi/2}) = q_0.$$

In the above equations, $\frac{dx}{dy}(x_{\pi/2}, y_{\pi/2})$ is the tangent on the occluding contour at point $(x_{\pi/2}, y_{\pi/2})$. These equations tell us the geometric relation between a $p = 0$ curve and its corresponding occluding contour. The relation is used to compute the 3D coordinate and surface orientation on a $p = 0$ curve.

3.2.5 Disparity

For a surface point (x_0, y_0, z_0), after α_0 and α_1 rotation of the object, we have

$$x_{\alpha_0} = x_0 \cos\alpha_0 - z_0 \sin\alpha_0, \tag{3.6}$$

$$z_{\alpha_0} = x_0 \sin\alpha_0 + z_0 \cos\alpha_0, \tag{3.7}$$

$$x_{\alpha_1} = x_0 \cos\alpha_1 - z_0 \sin\alpha_1, \tag{3.8}$$

$$z_{\alpha_1} = x_0 \sin\alpha_1 + z_0 \cos\alpha_1. \tag{3.9}$$

Subtracting Equation 3.8 from Equation 3.6,

$$x_{\alpha_0} - x_{\alpha_1} = x_0(\cos\alpha_0 - \cos\alpha_1) - z_0(\sin\alpha_0 - \sin\alpha_1). \tag{3.10}$$

The quantity $x_{\alpha_0} - x_{\alpha_1}$ is called *the disparity* of the two image points, $(x_{\alpha_0}, y_{\alpha_0})$ and $(x_{-\alpha_1}, y_{-\alpha_1})$. The disparity depends on the x and z coordinates of the surface point. However, when $\alpha_1 = -\alpha_0$,

$$x_{\alpha_0} - x_{\alpha_1} = -z_0(\sin\alpha_0 - \sin\alpha_1) = -2z_0 \sin\alpha_0. \tag{3.11}$$

This means that for a surface point, the disparity of its two image points in the two images taken after the object has been rotated by α and $-\alpha$, respectively, depends only on the depth of the surface point and the rotation angle. For convenience, we define α and $-\alpha$, which represents the same amount of rotation but in the two different directions, as *two symmetric angles*.

3.2.6 Surface approximation

For a C^2 surface $z = f(x, y)$, its first and second partial derivatives exist, and are continuous. If $p(x, y) = \frac{\partial z(x,y)}{\partial x}$ and $q(x, y) = \frac{\partial z(x,y)}{\partial y}$ are known everywhere on the surface and the z coordinate z_0 is known at a starting point (x_0, y_0), the z coordinate $z(x, y)$ at another point (x, y) can be computed by integrating the surface orientation along a path which connects (x_0, y_0) to (x, y) as

$$z(x, y) = z_0 + \int_{x_0}^{x} p(x, y)dx + \int_{y_0}^{y} q(x, y)dy.$$

For a C^2 surface, the computed $z(x, y)$ is independent of integration path since p and q are integrable. In computer vision, the data obtained or computed are usually at image points on a uniform grid. Assuming $p(x, y)$ and $q(x, y)$ are known at every image point and the z coordinate is known at a starting point (x_0, y_0), the surface can be approximated by

$$z(x, y) = z_0 + \sum_{x_0}^{x} p(x, y)\Delta x + \sum_{y_0}^{y} q(x, y)\Delta y, \tag{3.12}$$

where Δx and Δy are the distances between the two adjacent image points in the x and y direction respectively. On a uniform image grid, $\Delta x = \Delta y$. As the coordinates are measured in pixels, Δx and Δy have unit length. Generally, p and q computed from images are not integrable. Therefore $z(x, y)$ may be different if a different integration path is used. An elegant way to enforce integrability is to project a normal map onto a "nearest" integrable normal map [21].

3.3 Photometric Constraints

3.3.1 Reflectance function under a collinear light source

In the general case, the image brightness of a surface point under a distant light source is determined by the reflectance function $R(i, e, g)$ [78]. As shown in Fig. 3.13, the

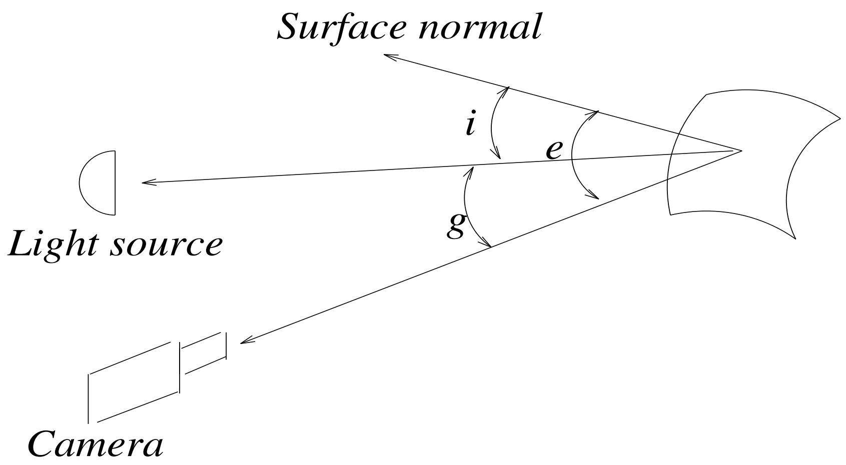


Figure 3.13: The image brightness is a function of the angles i, e and g. The incident angle i is from the illuminant direction to the surface normal. The emergent angle e is from viewing direction to the surface normal. The phase angle g is from the illuminant direction to the viewing direction.

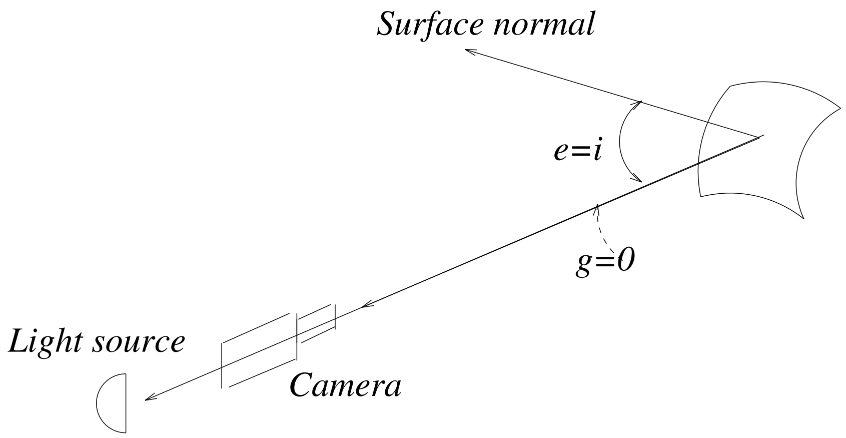


Figure 3.14: The image brightness is a function only of the emergent angle e under a collinear light source.

incident angle i is the angle between the illuminant direction and the surface normal, the emergent angle e is the angle between the viewing direction and the surface normal, and the phase angle g is the angle between the illuminant direction and viewing direction. Under a collinear light source, as shown in Fig. 3.14, the phase angle g becomes zero and the emergent angle e becomes the same as the incident angle i. In this case, all the components of the reflectance such as the specular component, diffuse component and other components defined in Tagare's paper [66] are functions only of the emergent angle e. Thus for a uniform surface, its image brightness value E at a surface point can be written as

$$E = R(e) \tag{3.13}$$

where e, $0 \leq e \leq \pi/2$, is the emergent angle at the surface point. While the emergent angle e is defined in an object centered coordinate system, the surface orientation $(-p, -q, 1)$ is defined in a viewer centered coordinate system. The relation between the emergent angle e and the surface orientation at a point can be expressed as

$$\cos e = \frac{1}{\sqrt{p^2 + q^2 + 1}}. \tag{3.14}$$

Since the surface reflectance function is strictly monotonic, the inverse of the surface reflectance function $e = R^{-1}(E)$ exists. For convenience, sometimes we express the surface reflectance function as $E = Q(\cos e)$. It is easy to show that the inverse surface reflectance function, $\cos e = Q^{-1}(E)$, also exists. In surface recovery, the inverse function $Q^{-1}(E)$ is used to compute the surface normal from the image brightness values of a surface point.

Under a collinear light source, the surface reflectance function is simplified to a function only of one variable. This simplification is crucial in surface reflectance extraction and surface recovery.

3.3.2 Singular points

A singular point is an image point which has a particular brightness value (maximum) and this particular brightness value corresponds uniquely to a particular surface orientation. Under a collinear light source, the particular surface orientation is the direction normal to the image plane. This property can be directly derived from the surface reflectance function of an object under a collinear light source.

The x and y coordinates of a singular point can be directly detected from images by searching for points of the maximum brightness value. The depth at a singular point can be determined from the corresponding contour point. If there is no occlusion, after $\pi/2$ rotation of the object, the image of the surface point corresponding to the singular point will be a contour point in $image_{\pi/2}$. Let the surface point corresponding to the singular point in $image_0$ be (x_0, y_0, z_0). After $\pi/2$ rotation of the object, from Equation 3.1, we have

$$-x_{\pi/2} = z_0 \tag{3.15}$$

The quantity $-x_{\pi/2}$ can be measured as the distance from the corresponding contour point to the projection of the rotation axis. It is easy to find the correspondence between a singular point and a contour point since the two image points have the same y coordinates and the tangent on the occluding contour at the contour point is vertical in $image_{\pi/2}$.

Once the 3D location of a singular point is known, the projections of the corresponding surface point on the images taken during object rotation can be calculated. The image brightness values of the surface point are used to build the surface reflectance function. The singular points are also used as starting points for surface recovery.

3.3.3 Equi-brightness constraint on $p = 0$ curve

The surface orientation at a surface point changes during object rotation. So does the image brightness value of the surface point. Let the surface orientation at a surface point on a $p = 0$ curve be $(-p_0, -q_0, 1)$ with $p_0 = 0$. After the object has been rotated by α, $(\alpha < \pi/2)$, from Equations 3.4–3.5, the surface orientation of the point is

$$\begin{aligned} p_\alpha &= \tfrac{\sin\alpha}{\cos\alpha} = \tan\alpha, \\ q_\alpha &= \tfrac{q_0}{\cos\alpha}. \end{aligned}$$

The cosine of the emergent angle at the surface point is

$$\cos e_\alpha = \frac{1}{\sqrt{p_\alpha^2 + p_\alpha^2 + 1}} = \frac{1}{\sqrt{\tan^2\alpha + (\frac{q_0}{\cos\alpha})^2 + 1}}.$$

If the object has been rotated by $-\alpha$, instead of α, the surface orientation at the point after the rotation is

$$\begin{aligned} p_{-\alpha} &= \tfrac{\sin -\alpha}{\cos -\alpha} = -\tan\alpha \\ q_{-\alpha} &= \tfrac{q_0}{\cos -\alpha} = \tfrac{q_0}{\cos\alpha} \end{aligned}$$

The cosine of the emergent angle at the surface point is

$$\cos e_{-\alpha} = \frac{1}{\sqrt{p_{-\alpha}^2 + p_{-\alpha}^2 + 1}} = \frac{1}{\sqrt{(-\tan\alpha)^2 + (\frac{q_0}{\cos\alpha})^2 + 1}} = \cos e_\alpha.$$

Let the surface reflectance function be $E = Q(\cos e)$. The brightness value of this surface point after α rotation is $E_\alpha = Q(\cos e_\alpha)$. The brightness value of the same surface point after $-\alpha$ rotation is $E_{-\alpha} = Q(\cos e_{-\alpha})$. Since $\cos(e_{-\alpha}) = \cos(e_\alpha)$, we have

$$E_\alpha = E_{-\alpha}.$$

This equation tells us that under a collinear light source, each point on a $p = 0$ curve gives rise to the same brightness in two images taken after the object has been rotated by two symmetric angles. The equi-brightness constraint comes from the fact that for a $p = 0$ point, $e_\alpha = e_{-\alpha}$.

3.3.4 Surface orientation from brightness values

Under a collinear light source, the surface reflectance function has the form $E = R(e)$ or $E = Q(\cos e)$ with $\cos e = 1/\sqrt{p^2+q^2+1}$. When an object rotates, the surface orientation of a surface point on the object changes and so does the image brightness value of this point. Let a surface point T have surface orientation $(-p_0, -q_0, 1)$, and the image brightness value $E_0 = Q(\cos e_0)$. After the object rotates by α, let the surface orientation of T be $(-p_\alpha, -q_\alpha, 1)$ and the brightness value of T be $E_\alpha = Q(\cos e_\alpha)$. The following calculation shows how we derive the surface orientation of a surface point from its two image brightness values, E_0 and E_α.

First, we show that the surface orientation of T can be calculated from $\cos e_0$ and $\cos e_\alpha$. From Equation 3.14,

$$\cos e_0 = \frac{1}{\sqrt{p_0^2+q_0^2+1}}, \tag{3.16}$$

$$\cos e_\alpha = \frac{1}{\sqrt{p_\alpha^2+q_\alpha^2+1}}. \tag{3.17}$$

Using Equation 3.4 and 3.5 to substitute p_α and q_α in Equation 3.17, we have

$$\cos e_\alpha = \frac{1}{\sqrt{1+(\frac{p_0\cos\alpha+\sin\alpha}{\cos\alpha-p_0\sin\alpha})^2+(\frac{q_0}{\cos\alpha-p_0\sin\alpha})^2}} = \frac{1}{\sqrt{p_0^2+q_0^2+1}}(\cos\alpha-p_0\sin\alpha). \tag{3.18}$$

By using Equation 3.16, this equation can be simplified to

$$\cos e_\alpha = \cos e_0(\cos\alpha - p_0\sin\alpha). \tag{3.19}$$

Solving the above equation for p_0,

$$p_0 = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{\cos e_\alpha}{\cos e_0}. \tag{3.20}$$

Applying the inverse of the surface reflectance function to the image brightness values, E_0 and E_α, to estimate $\cos e_0$ and $\cos e_\alpha$ and substituting the estimated values for $\cos e_0$

and $\cos e_\alpha$ in Equation 3.20, we get

$$p_0 = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{Q^{-1}(E_\alpha)}{Q^{-1}(E_0)}. \tag{3.21}$$

After p_0 has been calculated, q_0 can be solved from Equation 3.16 as

$$q_0 = \pm\sqrt{\frac{1}{\cos^2 e_0} - p_0^2 - 1} = \pm\sqrt{\frac{1}{Q^{-1}(E_0)^2} - p_0^2 - 1}. \tag{3.22}$$

The geometric explanation for the solution of p_0 and q_0 is shown in Fig. 3.15. Since

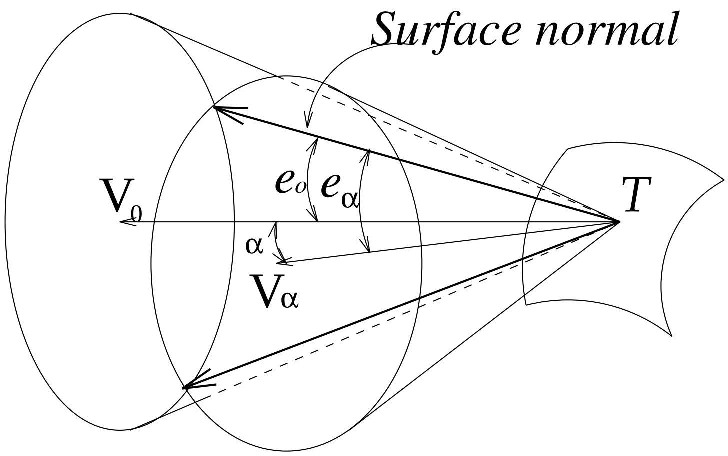


Figure 3.15: The solution for surface orientation is the intersection of two cones.

rotating an object in front of a collinear light source is equivalent to rotating the collinear light source around a fixed object, we assume E_0 is the image brightness of T illuminated and viewed in direction V_0 and E_α is the image brightness of T illuminated and viewed in direction V_α. The value of $\cos e_0$ determined from the brightness E_0 gives the emergent angle e_0 and all the surface normals with angle e_0 to V_0 form a cone with V_0 as its axis. The value of $\cos e_\alpha$ determined from the brightness E_α gives the emergent angle e_α and all the surface normals with angle e_α to V_α form the other cone with V_α as its axis. The intersection of the two cones generates two vectors which are the solutions for the surface orientation at T. The q components of the two vectors have the same magnitude but opposite signs just as indicated by Equation 3.22. Since only one of the two solutions is valid, there is a two-way ambiguity. This ambiguity can be removed by using other constraints.

3.4 Summary

In this chapter, we have described the geometric and photometric constraints in images of a rotating object under a collinear light source. We have also implicitly introduced the relations among these constraints. Under orthographic projection and the illumination of a collinear light source, both geometric and photometric constraints are simplified and easy to identify from images. Also the relations among these constraints are easy to explore. These constraints and their relations are the theoretical foundation for our work.

In the following chapters, we will further explore the relations among these constraints and integrate these constraints for surface reflectance extraction and shape recovery. In Chapter 4, we present the photogeometric technique. The technique extracts the surface reflectance function directly from images of a rotating object and then applies the extracted surface reflectance function to surface recovery. In Chapter 5, we describe the wire-frame technique which extracts a set of 3D curves on the object surface without the knowledge of the surface reflectance function.

Chapter 4

Photogeometric Technique

In this chapter we describe a new technique, called *the photogeometric technique*, for surface reflectance extraction and surface recovery. We show that under a collinear light source, the surface reflectance function of a rotating object can be directly extracted from the image sequence of the rotating object. We also show that the depth and surface orientation of the object can be recovered simultaneously by using brightness in two images of a rotating object once the surface reflectance function is obtained. Experiments are conducted on real image sequences of objects of different surface reflectance and different geometric properties. The experimental results show that the technique is feasible and effective. Section 4.1 of this chapter describes the imaging apparatus. Section 4.2 presents a method for extracting the surface reflectance function directly from an image sequence of an object. Section 4.3 describes the surface recovery process. Section 4.4 present a method to remove the ambiguity in the q component of surface orientation. Section 4.5 describes the recoverable area in an image. Section 4.6 to Section 4.8 show some experimental results on real image sequences of different objects. The final section summarizes this chapter.

4.1 Experimental Conditions

The imaging apparatus for surface recovery has been described in Section 3.1. A calibrated image facility built in our lab controls the object rotation and the imaging apparatus (see Section 6.1 for further details). A DC powered and parallel projected light source is mounted on the top of the camera. The light source and the camera point in the same direction towards the object on a turntable. In practice, the radiance of the light source is not constant over the illuminated area. A uniform white board is used to calibrate the non-uniform illumination (see Section 6.3). Since the distance from the camera to the object is far larger than the size of the object, the camera is set to telephoto mode and orthographic projection is assumed as a reasonable approximation. The actions of rotating an object and taking images are synchronized by a computer.

4.2 Extracting the Surface Reflectance Function

As we have shown in Subsection 3.3.1, under the illumination of a collinear light source, the surface reflectance function of an object is simplified to a function only of the emergent angle e. Let the surface reflectance function be $E = R(e)$, $0 \leq e \leq \pi/2$. To build the surface reflectance function, we only need to find the correspondence from e to E. This is much easier than building a reflectance function of two variables.

The surface reflectance function can be extracted from any surface points of known 3D location and surface orientation. From the known 3D location, their image locations during rotation can be found and their image brightness values can be extracted. From the known surface orientation, their emergent angles during rotation can be calculated. Then the correspondence from the brightness values to the emergent angles can be built. Since it is easy to find the 3D locations and surface orientations of the surface points whose images are singular points, these surface points are used as sample points for building the surface reflectance function. Other surface points, such as $p = 0$ points, can be used to extract a pseudo surface reflectance function which can also be used for surface recovery (see Subsection 7.3.2).

The 3D coordinates of the sample points are calculated by the method described in

Subsection 3.3.2. Let (x_0, y_0, z_0) be the calculated coordinate of a sample surface point S. The projection (x_α, y_α) of S on $image_\alpha$, which is taken after α rotation of the object, can be computed by Equations 3.1–3.3. Thus the brightness value $E_\alpha(S)$ of S in $image_\alpha$ can be extracted. The projection of S on an image may be located between pixels and linear interpolation between image pixels is used to calculate the image brightness value. As the image of S in $image_0$ is a singular point, the initial emergent angle $e_0(S)$ of S is zero. It is easy to show that the emergent angle $e_\alpha(S)$ at S after the object rotation by α is α. Since the rotation angle of the object is precisely controlled, the rotation angle is taken as the emergent angle of S. Therefore the surface reflectance function we build is a function from the rotation angle of the object to the brightness value of S, that is

$$E_\alpha(S) = R(\alpha) \quad or \quad E_\alpha(S) = Q(\cos\alpha)$$

Since $e_\alpha(S) = \alpha$ during rotation, we have

$$E_\alpha(S) = R(e_\alpha(S)) \quad or \quad E_\alpha(S) = Q(\cos e_\alpha(S)).$$

As the surface reflectance is uniform, the function can be applied to any surface point. Then we have

$$E = R(e) \quad or \quad E = Q(\cos e).$$

The function we actually used for surface recovery is the inverse function $\cos e = Q^{-1}(E)$. The inverse function is linearly interpolated for every integer brightness value ranging from 0 to 255.

4.3 Surface Recovery

After the surface reflectance function has been obtained, surface recovery can be done by using any two images in the image sequence of the rotating object. Without losing generality, we use the first image, $image_0$, and the image taken after the object has been rotated by α, $image_\alpha$, $\alpha < \pi/2$. The depth and surface orientation are computed at every point in $image_0$. Three subprocedures are used in surface recovery.

The first subprocedure calculates the image location of the corresponding image point in $image_\alpha$ for an image point of known depth in $image_0$. Let $(x_0(S), y_0(S))$ be the image of a surface point S in $image_0$ with depth $z_0(S)$. The location of image point $(x_\alpha(S), y_\alpha(S))$ of S in $image_\alpha$ is computed by using Equations 3.1–3.2 as

$$\begin{aligned} x_\alpha(S) &= x_0(S)\cos\alpha - z_0(S)\sin\alpha; \\ y_\alpha(S) &= y_0(S). \end{aligned}$$

The second subprocedure determines the orientation $(-p_0(S), -q_0(S), 1)$ at an image point from its image brightness $E_0(S)$ in $image_0$ and the brightness of its corresponding image point $E_\alpha(S)$ in $image_\alpha$ by using Equation 3.20 and Equation 3.22,

$$\begin{aligned} p_0(S) &= \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{Q^{-1}(E_\alpha(S))}{Q^{-1}(E_0(S))}; \\ q_0(S) &= \pm\sqrt{\frac{1}{[Q^{-1}(E_0(S))]^2} - [p_0(S)]^2 - 1}. \end{aligned}$$

The third subprocedure does local expansion of depth from an image point of known depth and surface orientation by using the first-order Taylor series approximation. Let N be another surface point which is close to S so that the image of N in $image_0$ is $(x_0(N), y_0(N)) = (x_0(S)+\Delta x, y_0(S)+\Delta y)$. The depth $z_0(N)$ at image point $(x_0(N), y_0(N))$ can be calculated by using Equation 3.12 as

$$z_0(N) = z_0(S) + p_0(S)\Delta x + q_0(S)\Delta y.$$

The surface recovery procedure starts at the image points whose depths are known. These starting points could be the images of the sample surface points we used to extract the surface reflectance function. By iterating the three subprocedures, the computation can be extended over the whole object image to obtain the depth and surface orientation at the same time. At a background image point, the depth is set to zero and the surface orientation is set to a particular value. The surface recovery procedure can be described as follows.

Surface recovery procedure

1. Take a point in $image_0$ of known depth as an input point. For the input point, do step 2-4;
2. Use the first subprocedure to calculate the location of its corresponding image point in $image_\alpha$.
3. Use the second subprocedure to determine its surface orientation from its image brightness in $image_0$ and the brightness of the corresponding image point in $image_\alpha$.
4. Use the third subprocedure to compute the depth for its neighboring points.
5. Take each of the neighboring points as the input point and repeat step 2-4 until the computation is extended to every pixel in $image_0$.

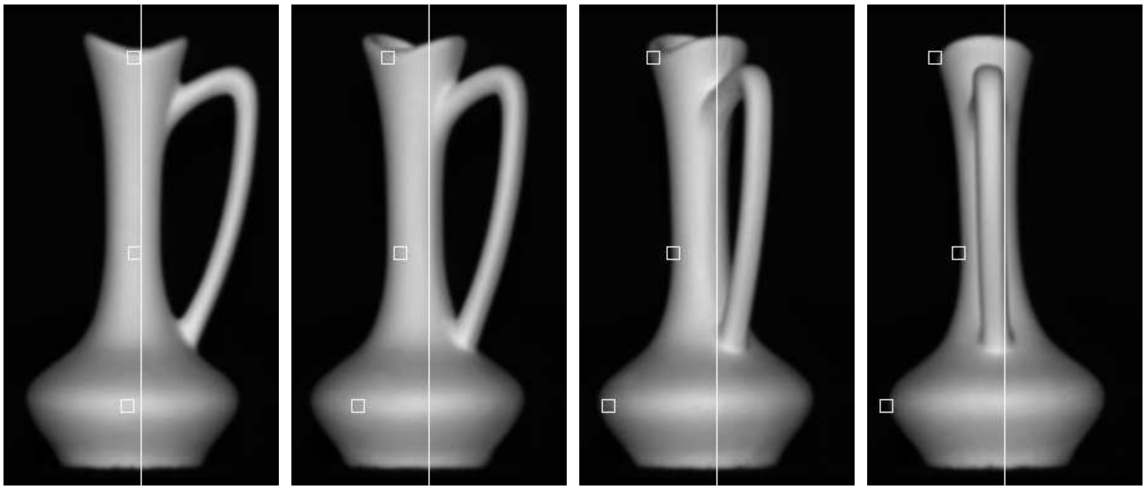
4.4 Removing Ambiguity in q

When we compute a new depth $z_0(N)$ for the next image point N by using the depth $z_0(S)$ at the current image point S in the y direction, we have $z_0(N) = z_0(S) + q_0(S)\Delta y$ with $q_0(S)$ as the q component of the surface orientation at S. An ambiguity in depth occurs as there are two solutions of different signs for q (see Subsection 3.3.4). There is no ambiguity when we extend the computation in the x direction as the solution for p is unique. The ambiguity in the depth caused by the ambiguity in q can be removed by the C^2 surface assumption and some image points of known depths. In the actual implementation of surface recovery, the computation for depth and surface orientation is extended along a $p = 0$ curve first by integrating the q component. After removing the depth ambiguity on the $p = 0$ curve, the computation is extended from every point in the $p = 0$ curve in the x direction. Assume that the surface recovery procedure starts from one image point of known depth to another image point of known depth along a non-horizontal path and that the q values along the path have the same sign. Then, the sign of the q component used to compute the depth along the path can be determined from the sign of the difference of the two depths at the two end points on the path. For example, if a path connects two closest singular points, then the sign of the q component along the path can be determined from the sign of the difference of the two depths at the two closest singular points.

4.5 Recoverable Area

In surface recovery, the depth and surface orientation could not be recovered at every point in $image_0$. The part of the surface that can be recovered must be visible in both images used for surface recovery. Some surface points visible in one image may not be visible in the other image. If $image_0$ and $image_\alpha$ are used, a surface point whose p component of surface orientation is larger than $\tan(\pi/2-\alpha)$ will not be visible in $image_\alpha$.

Gaussian filtering is applied to the input images to reduce image noise and quantization effects. At the image points close to the background, the brightness values are blurred with the background brightness by Gaussian filtering. The surface recovery at these image points is not accurate and reliable. So a brightness threshold, which is a little bit larger than the background brightness, is used to detect these image points. The depth and surface orientation are recovered only at image points where the brightness values are larger than the threshold. The threshold is empirically determined by the



a. $image_0$ b. $image_{\pi/6}$ c. $image_{\pi/3}$ d. $image_{\pi/2}$

Figure 4.16: Tracking sample points over image sequence. The center of a square box is the image of a sample point.

brightness value $E = Q(\cos \frac{85\pi}{180})$. Thus the surface can only be recovered at the surface points whose image brightness values in both images are larger than the threshold. For the surface points whose images are close to the occluding contours in $image_0$ or $image_\alpha$, their depths and surface orientations cannot be recovered.

The recoverable area in $image_0$ is determined by the surface orientation of the surface and the two images used for surface recovery. But the actually recovered area in $image_0$ may not be exactly the same as the recoverable area because of image noise, non-unform surface reflectance, and other factors. In addition, Gaussian filters of different sizes give different degree blurring effects. The actually recovered area may change with the sizes of Gaussian filters used.

4.6 An Experiment with Matte Surface

4.6.1 Surface reflectance function

To extract the surface reflectance of a vase, nineteen images of the vase are taken during $\pi/2$ rotation with each successive image taken after a $\pi/36$ successive rotation. Figure 4.16 shows four of the image sequence. Images (a), (b), (c), and (d) are taken after 0, $\pi/6$, $\pi/3$, and $\pi/2$ rotation, respectively. The white line in the middle of each image is the rotation axis of the object. Three sample surface points are used to extract the surface reflectance function. The image brightness and surface orientation of the sample points are tracked over the nineteen images. The initial 3D coordinates of the three sample surface points are obtained from the singular points in $image_0$ and the corresponding occluding contour in $image_{\pi/2}$. The center of a small square box in Fig. 4.16 marks the image of a sample point. The average of the brightness values of the three sample points is used to build the reflectance function. The solid line in Fig. 4.17 shows the surface

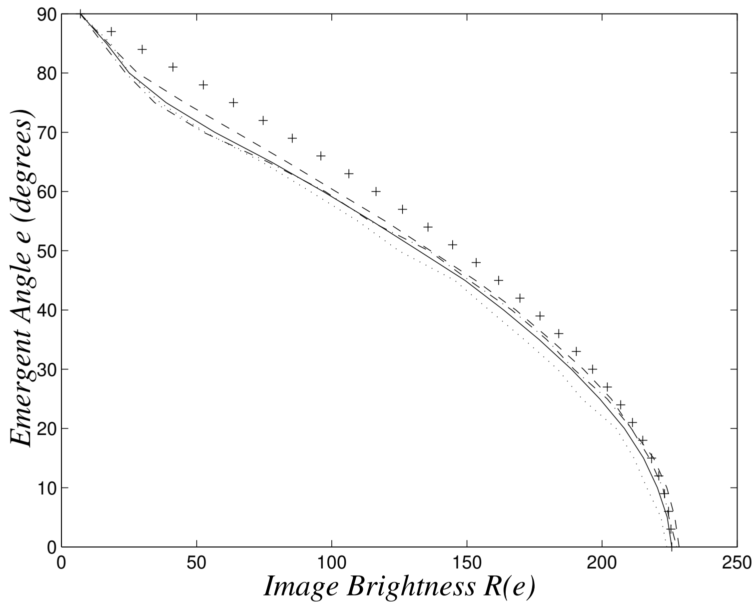


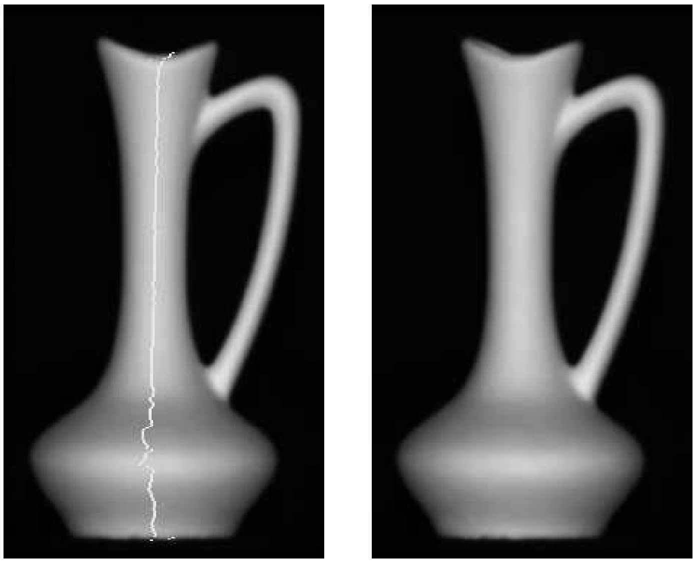
Figure 4.17: The solid line shows the reflectance function obtained by averaging the three reflectance functions. The plus line is Lambertian reflectance function of the same maximum brightness. The other three lines show the three reflectance functions extracted from the three sample points of the vase.

reflectance function extracted from images. The emergent angle is represented by the vertical axis in degrees for simplicity in plotting, and the image brightness is represented by the horizontal axis. The plus line in the figure is the Lambertian reflectance function which has the same maximum brightness as the surface reflectance function. The difference between the two reflectance function shows that the surface of the vase is not Lambertian although it is made of clay and considered as a diffuse surface.

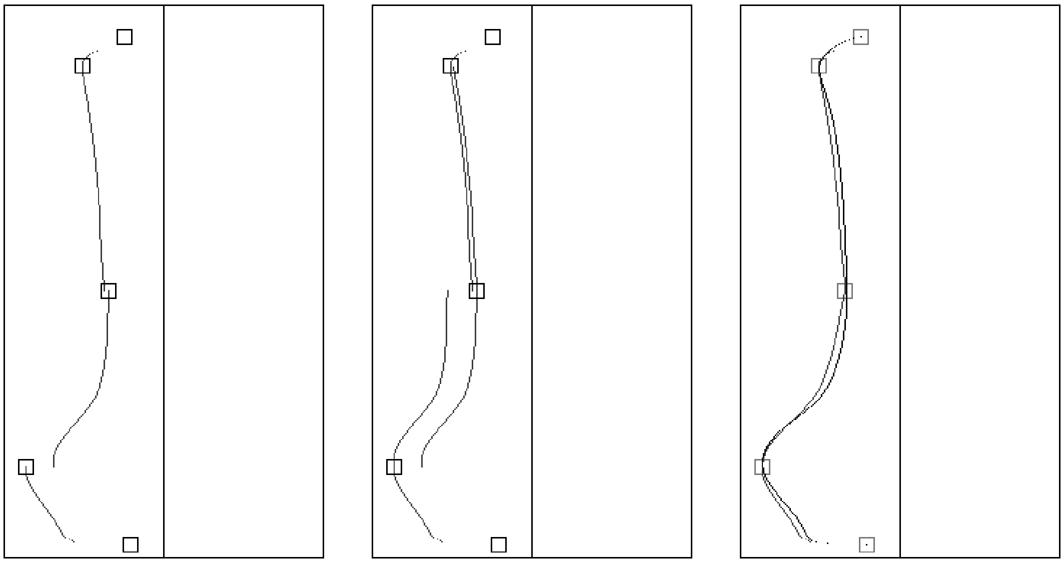
4.6.2 Surface recovery on a $p=0$ curve

In surface recovery of the vase, $image_0$ (Fig. 4.18(a)) and $image_{\pi/18}$ (Fig. 4.18(b)) are used. Depth and orientation are recovered at each point in $image_0$. Starting from each singular point of known depth, depth and surface orientation are recovered along a $p=0$ curve first. The depth and surface orientation could be first computed in the x direction along some $q=0$ curves and then extended in the y direction at every point on the $q=0$ curves. However, this requires solving the depth ambiguity on every path along the y direction. Also the depth computed along the y direction is less accurate since, at every image point, the computed q component is not as accurate as the computed p component (see error analysis in Chapter 7). A $p=0$ curve passes through some singular points and contour points in $image_0$. The depths at the singular points and contour points are obtained from the corresponding contour points in $image_{\pi/2}$. The sign of q on a path which connects a singular point to its closest singular point or contour point is determined by the sign of the depth difference at the two end points. To compute the depth and orientation on a path, take a singular image point in $image_0$ as an input point and do the following until the other end point is reached.

1. Starting from the input point, extend the computation on depth and surface in the two directions along the x axis in $image_0$ until a $p=0$ point is found. Output the 3D location of the $p=0$ point.



a. $image_0$ b. $image_{\pi/18}$



c. depth computed d. depth from two directions e. after weighted average

Figure 4.18: (a) and (b) are images for surface recovery. The white curve in (a) is a $p = 0$ curve. The square boxes denote the depths at singular or contour points. The distance from a point to the vertical black line is the depth. (c) the depth computed in one direction. (d) the depth computed in two directions. (e) the depth after the weighted average. The smooth curve is the real depth obtained from the corresponding contour.

2. At the $p = 0$ point, use the third subprocedure to calculate the depth for its adjacent image point in the next row. Take the adjacent point as the input point and go to step 1.

Figure 4.18(c) shows the depth computed on a $p = 0$ curve. The straight line in the middle of Fig. 4.18(c)-(e) represents the line of zero depth. The horizontal distance from a point to the line is the depth. The center of a small box in the diagrams represents the true depth at a singular point or a contour point in $image_0$, which is obtained from the corresponding contour point in $image_{\pi/2}$ and used to remove the ambiguity in q. The white curve on the body of the vase in Fig. 4.18(a) is the curve of $p = 0$ on which the depth and surface orientation are recovered first. The curve consists of several paths which connect a singular point to another singular point or a contour point. Since the depth computed at the end point of a path may not be exactly the same as the known depth at the end point because of image noise, non-uniform albedo and other factors, the depth on the $p = 0$ curve may not be continuous at the singular points (see Fig. 4.18(c)).

A *distance-weighted averaging* method is used to make the computed depth continuous on the curve. The idea is to make the depth at the two ends of a path equal to the known depths. For a path which connects two singular points (x_0, y_0) and (x_k, y_k) with $y_0 \neq y_k$, the computation on the path can start from either of the two points. Thus two sets of depths along a path can be obtained (see Fig. 4.18(d)). Let Z_i, $(i = (0, k))$ be a set of depths computed along the path starting from (x_i, y_i) and ending at (x_{k-i}, y_{k-i}). Let the known depth at the point (x_i, y_i) be z_i. Let point (x_j, y_j), $0 \leq j \leq k$, be a point on the path and let $z_{i,j}$ be a depth in Z_i and computed at point (x_j, y_j). Since (x_0, y_0) is the starting point for Z_0 and (x_k, y_k) is the starting point for Z_k, we have $z_{0,0} = z_0$ and $z_{k,k} = z_k$. In the general case, $z_{0,k} \neq z_k$ and $z_{k,0} \neq z_0$. A new depth $\tilde{z}_j$ is calculated at point (x_j, y_j), $0 \leq j \leq k$, by

$$\tilde{z}_j = \frac{(k-j)z_{0,j}}{k} + \frac{jz_{k,j}}{k} \tag{4.23}$$

It is easy to show that $\tilde{z}_0 = z_{0,0} = z_0$ and $\tilde{z}_k = z_{k,k} = z_k$. After recomputing depth for each path, we get continuous depth on the curve. The true depth (extracted from occluding contour in $image_{\pi/2}$) and the recomputed continuous depth are represented in one image. The two sets of depths are quite close.

We have shown that the depth and surface orientation on a $p = 0$ curve can be determined from a few singular and contour points of known depth and surface orientation. In fact, the depth on a $p = 0$ curve can be directly determined from the occluding contour in $image_{\pi/2}$ and the x location of the $p = 0$ curve in $image_0$ can be determined by a search method that we will describe in Subsection 4.7.2.

4.6.3 Extending surface recovery over the entire image

Using the points on the $p = 0$ curve as the starting points, we extend the computation on the depth and surface orientation in the x direction by $z(N) = z(S) + p(S)\Delta x$ with S as the current point and N as the next point. Since the p value is unique, there are no ambiguities when the computation is extended in the x direction. Some areas in the image may not be recovered by extending the computation in the x direction since the computation may stop at a background image point before reaching another image area. Figure 4.19 shows the depth plot of the partially recovered surface. The top of the body and the handle are not recovered by extending the computation from the $p = 0$ curve on the body. We assume a single smooth surface in the image so the image area of the surface is connected. These unrecovered areas must connect to the recovered area in the y direction since they do not connect the recovered area in the x direction. For each unrecovered area, a $p = 0$ curve is computed first in the area. The $p = 0$ curve is

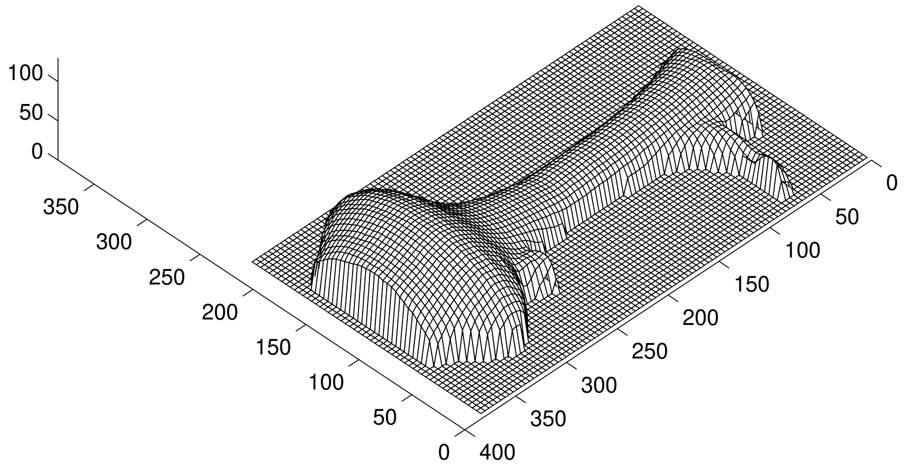


Figure 4.19: The surface depth plot of the partially recovered vase

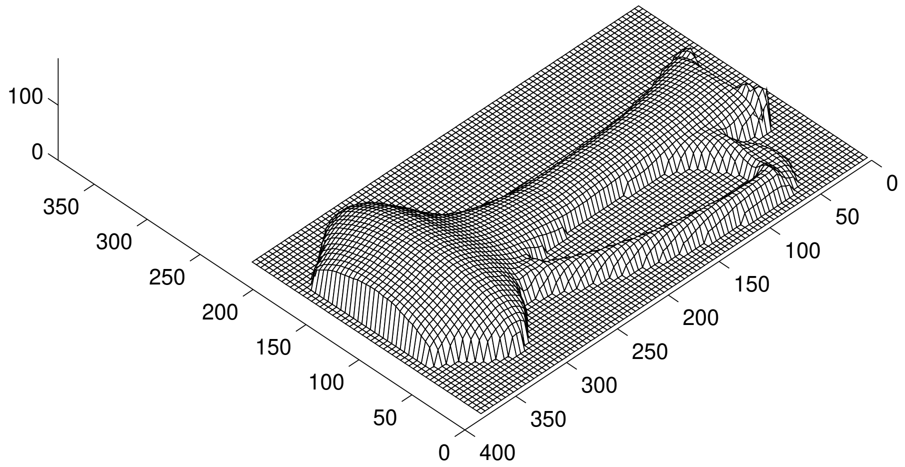


Figure 4.20: The surface depth plot of the recovered vase

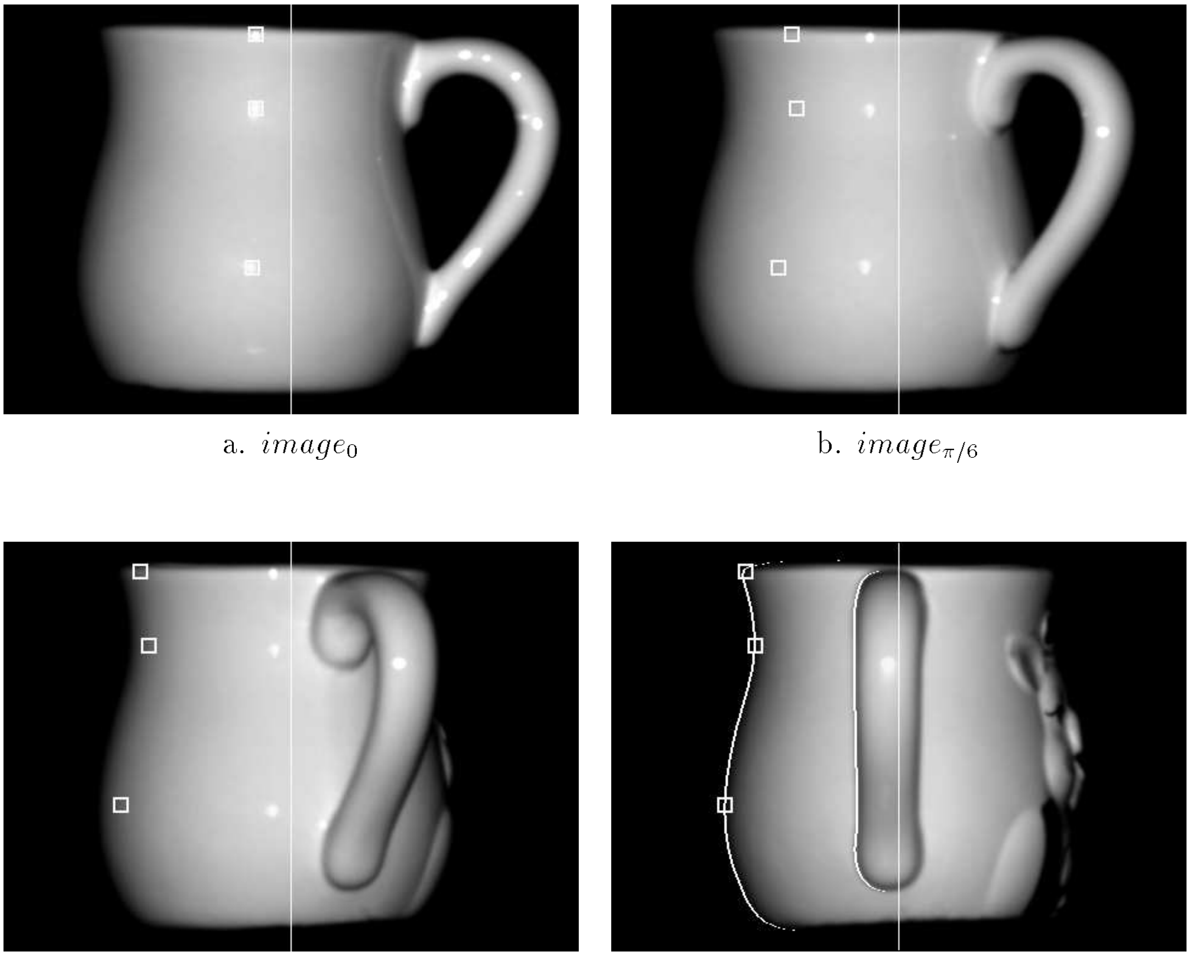
computed by using a $p = 0$ point at the boundary, which connects the unrecovered area, of an recovered area as the starting point. The depth ambiguity on the $p = 0$ curve is removed by using the depth at singular points or contour points. Then the depth and surface orientation are extended in the x direction. In this way, the entire visible surface can be recovered.

The depth and surface orientation are computed at every pixel in $image_0$. The brightness values in $image_{\pi/18}$ are linearly interpolated between pixels as the projection of a 3D point on $image_{\pi/18}$ may be located between pixels. The depth plot of the final recovered vase is shown in Fig. 4.20. The depth plots are displayed with Matlab [46]. We did not do any smoothing or regularization on the depth and surface orientation data.

4.7 An Experiment with Specular Surface

4.7.1 Surface reflectance function

We also experimented with a porcelain cup. Nineteen images are taken during $\pi/2$ rotation. The rotation angle between successive images is $\pi/36$. The reflectance of the cup presents a strong specular reflection so the brightness values at singular points are saturated. Figure 4.21 contains four images of the cup. Images (a) to (d) are the images of the cup taken after 0, $\pi/6$, $\pi/3$, and $\pi/2$ rotation. Three sample points are tracked over the image sequence to extract the surface reflectance function. In Fig. 4.22, the solid line represents the surface reflectance function of the cup, and the dash-dot line is the Lambertian reflection which has the same maximum brightness value as the extracted surface reflectance function. The difference between the two reflectance functions is mainly caused by the specular reflection of the surface. Also the diffuse reflection of the surface deviates from the Lambertian model when the emergent angle is large [75].



a. $image_0$ b. $image_{\pi/6}$

c. $image_{\pi/3}$ d. $image_{\pi/2}$

Figure 4.21: Tracking the sample points over the image sequence of a cup. The middle white line is the rotation axis, the center of a square box marks the image of a sample point. The white curves in $image_{\pi/2}$ are the detected occluding contours.

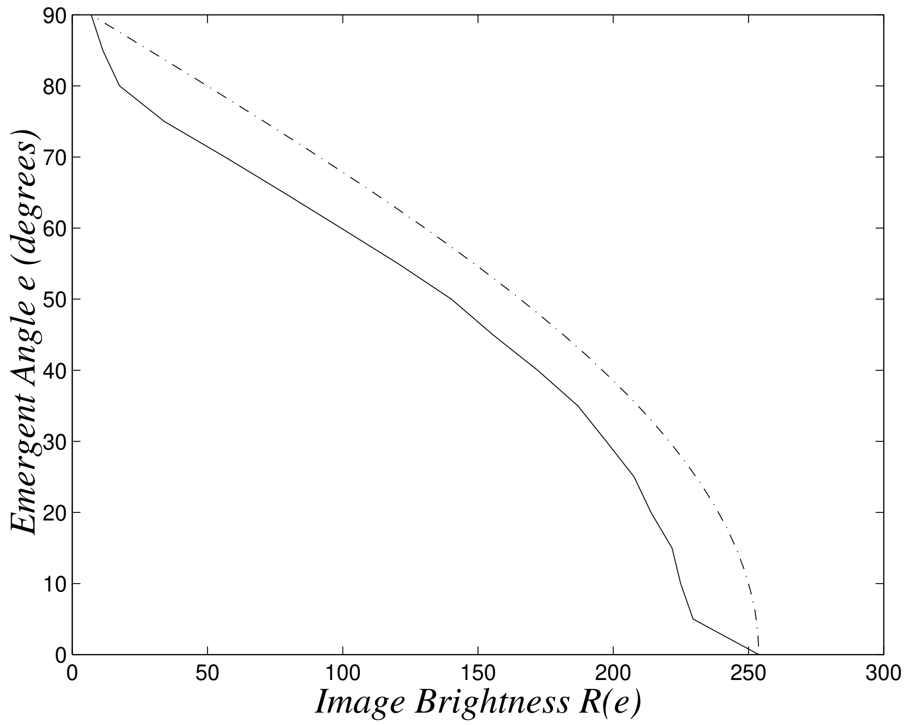


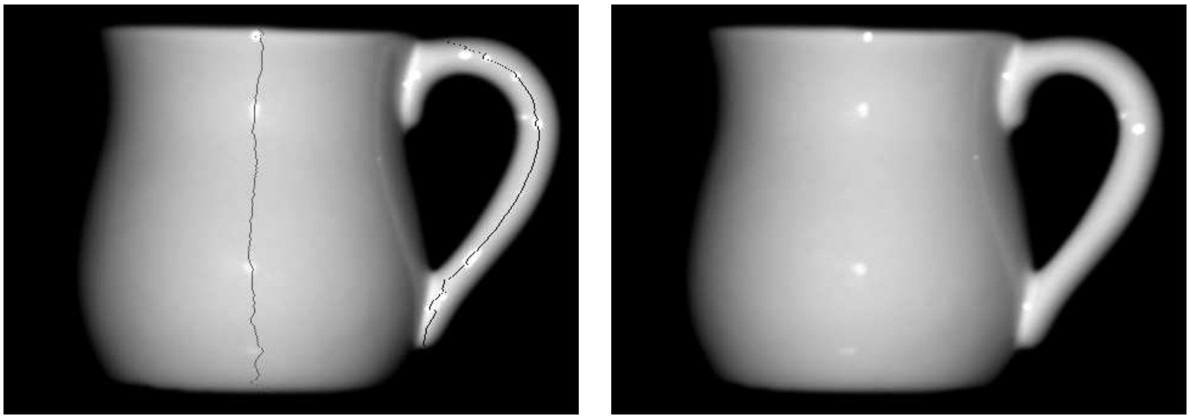
Figure 4.22: The surface reflectance function of the cup extracted from images(solid line). Lambertian reflectance function of the same maximum brightness (dash-dot line).

4.7.2 Alternative method for surface recovery on a $p = 0$ curve

In surface recovery of the cup, $image_0$ and $image_{\pi/18}$ are used. Like the surface recovery of the vase, the depth and surface orientation are recovered on a $p = 0$ curve first. However, for the cup, the depth on a $p = 0$ curve is determined directly from the occluding contours in $image_{\pi/2}$ of the cup. A $p = 0$ curve will project on $image_{\pi/2}$ as an occluding contour after $\pi/2$ rotation of the object (see Subsection 3.2.4). The horizontal distance from a point on the occluding contour to the image of the rotation axis gives the depth for the corresponding point on the $p = 0$ curve. The two white curves in Fig. 4.21 (d) are the occluding contours extracted from $image_{\pi/2}$. One occluding contour corresponds to the $p = 0$ curve on the body of the cup and the other corresponds to the $p = 0$ curve on the handle.

Although the depth on a $p = 0$ curve can be known from the corresponding occluding contour, the x coordinate on the $p = 0$ curve in $image_0$ still needs to be determined. To find the x coordinate, in each row in $image_0$, a search is carried on for an image point which has the measured depth and whose p component is zero. The search can be parallel by rows. To reduce the search space, the search is carried row by row starting from a singular point in $image_0$. Based on the C^2 surface assumption, the result in the current row is used to conduct the search in the next row so that the search space can be limited within a 1D window. The following procedure inputs the same row from the two images, the depth measured from the occluding contour, and a 1D window in $image_0$ and outputs the x location of a point on the $p = 0$ curve.

1. Take the first point in the window as an input point. For the input point do
2. Use the first subprocedure and the depth to calculate its correspondent in $image_\alpha$;
3. Use the second subprocedure to determine its surface orientation from its brightness and the brightness of the calculated correspondent. If the p component is zero,

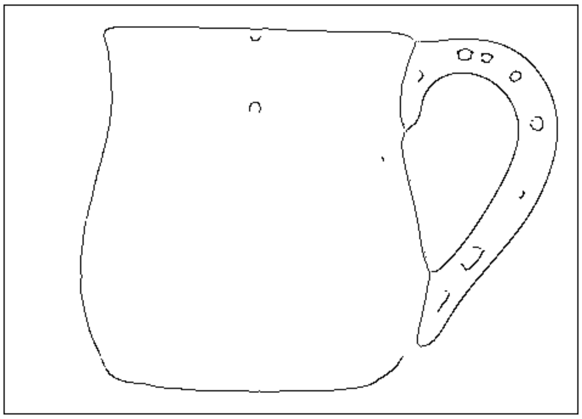


a. $image_0$ b. $image_{\pi/18}$

Figure 4.23: Images of the cup used for surface recovery. The black lines are $p = 0$ curves.

output the x location of the input point. Otherwise, take the next point in the window and goto step 2.

The two black curves in Fig. 4.23 (a) are the two $p = 0$ curves computed by using the search method. The 3D location of a $p = 0$ curve can also be obtained by the method we used in surface recovery of the vase. The method we used here has several advantages. First, the depth on a $p = 0$ curve is more accurate because the distance measured from occluding contour can be accurate within one pixel and there is no accumulated depth error on the curve. Second, the method does not have any ambiguity in depth. Finally, the depth given by the occluding contour is continuous. However, a $p = 0$ curve may not appear in $image_{\pi/2}$ as an occluding contour because of occlusion. In this case, a part of a $p = 0$ curve can be recovered by using the method described in this section and the rest of the curve can be recovered by the method used in the recovery of the $p = 0$ curve on the vase.



(a) Edges in $image_0$

(b) Edges used for recovery

Figure 4.24: Detecting surface orientation discontinuity

4.7.3 Detecting surface orientation discontinuity

The object surface is C^2. That means the surface orientation change is smooth. However, this assumption does not hold in general. Surface orientation discontinuities occur between different parts of an object. For the surface of the cup, the discontinuity appears at the joint between the body and handle. When surface recovery is extended over the discontinuity, the depth and surface orientation obtained will be inaccurate.

The surface orientation discontinuity should be detected in order to avoid surface recovery error. The discontinuity can be found by the Canny edge detector. Figure 4.24 (a) shows edges in $image_0$ of the cup. The edges are generated by the Canny edge detector with $\sigma = 2$ and threshold 20 for gradient. There are many image features which cause edges. Distinguishing the edges caused by occluding contours from the edges caused by surface orientation discontinuity can be done by tracking edges in three or more images [70] or using some other criteria. How to determine the type of an edge (e.g, surface mark, orientation discontinuity, shadow, occluding contour) is not the subject of this thesis so we just pick the edges caused by surface orientation discontinuity (see black lines in Fig. 4.24 (b)) by hand and use the edges for surface recovery.

4.7.4 Surface recovery of the cup

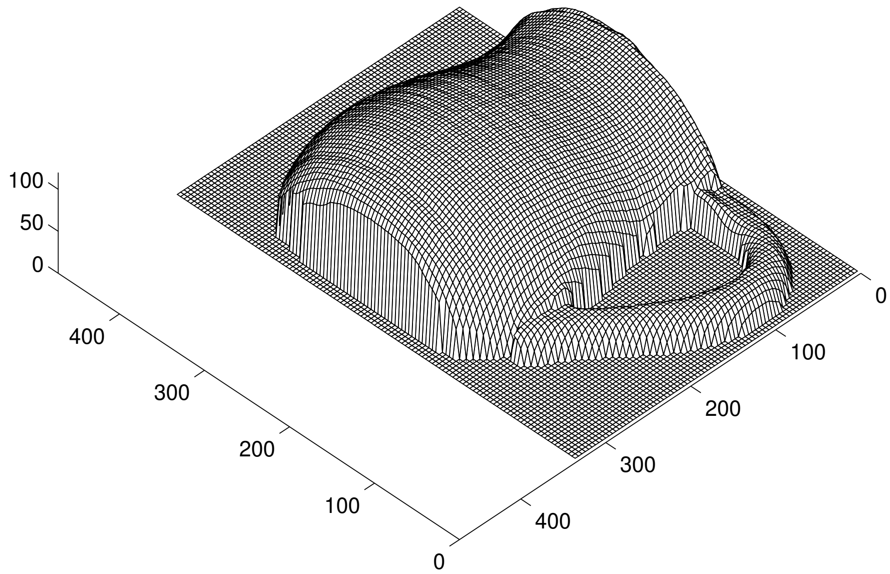


Figure 4.25: The depth plot of the partially recovered surface of the cup

The edges of orientation discontinuity are used as boundaries when the computation is extended on $image_0$. To recover the body or handle of the cup, starting from every point on the $p = 0$ curve, the computation for depth and surface orientation is extended in the x direction and stopped at the edges of orientation discontinuity or the occluding contour. Figure 4.25 is the partially recovered cup. What is left uncovered is the tip which is a part of the handle. A $p = 0$ point on the boundary, which connects to the unrecovered area, is used to compute a $p = 0$ curve on the tip. Then the surface recovery is extended from the $p = 0$ curve in the x direction. Figure 4.26 shows the depth plot of the entire visible part of the cup. The surface orientation is also recovered for the cup.

The sign of the q component is determined by using the sign of the difference of the depth in the y direction. Since the depth along the y direction is not smooth at some points because of surface recovery errors, the computed sign of q may not be be correct at these points. Under the C^2 surface assumption, the surface orientation should be smooth so the q component should be smooth. The sign of the q component is finally determined by taking the sign that the majority of its neighbor points have. Figure 4.27 is the needle map which represents the surface orientation of the recovered cup. The needle map is generated by projecting the surface normal of the cup on to a 2D plane. The normals are sampled from the points in every eighth row and column in $image_0$.

4.8 An Experiment with a Concave Object

We also conducted the experiment with a concave object nicknamed "pillow". The object is designed by Woodham and used by Ying Li [45] in her Ph.D work. The object is manufactured in a workshop and made of white clay. The images of the pillow are taken during $\pi/2$ rotation of the object. The rotation angle of the object between the two successive images in the sequence is $\pi/36$. Figure 4.28 shows four of the nineteen images of the pillow taken during $\pi/2$ rotation. We can see that the pillow has strong interreflection in the concave area. The 3D locations of the two sample points are calculated from the locations of the singular points in $image_0$ and its corresponding contour points in $image_{\pi/2}$. The images of the two sample points are labeled with small square boxes in Fig. 4.28. The surface reflection function shown in Fig. 4.29 is built by tracking the images of the two sample surface points in the image sequence. The occluding contour in $image_{\pi/2}$ (see white curve in Fig. 4.28 (d)) is used to compute the 3D location on the $p = 0$ curve.

The two images, $image_0$ and $image_{\pi/18}$, of the pillow are used for surface recovery.

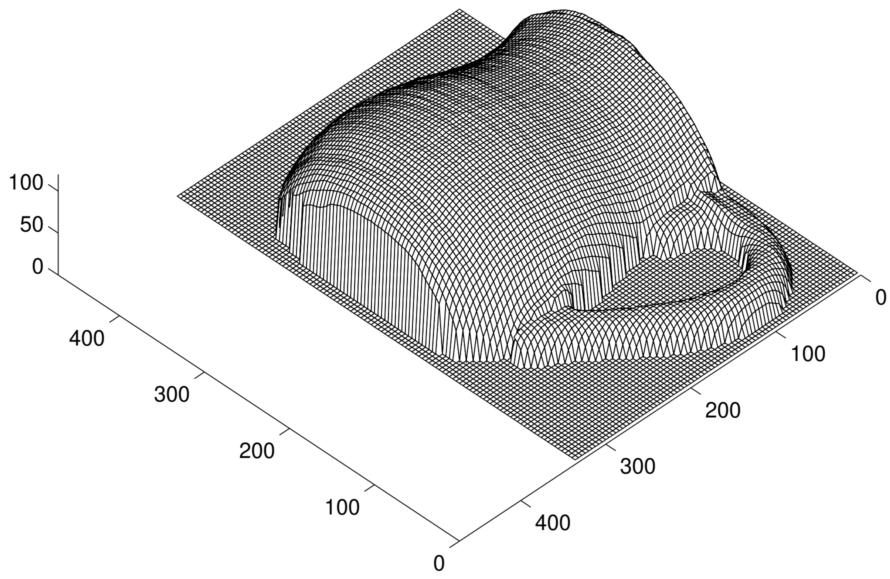


Figure 4.26: The depth plot of the final recovered cup

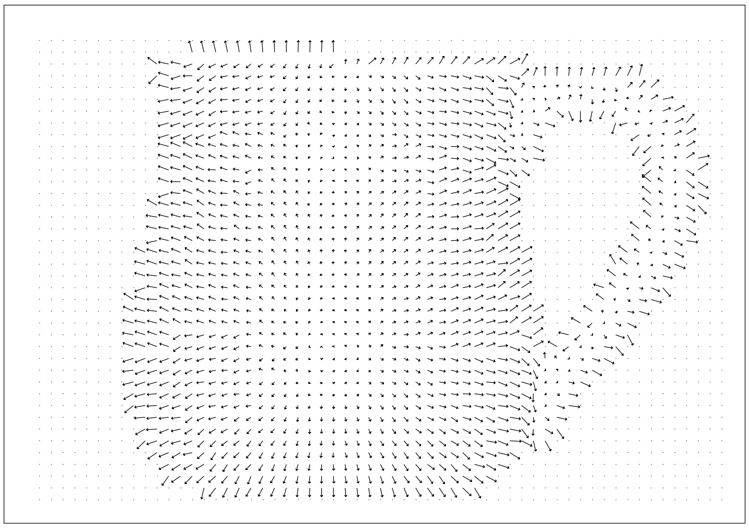
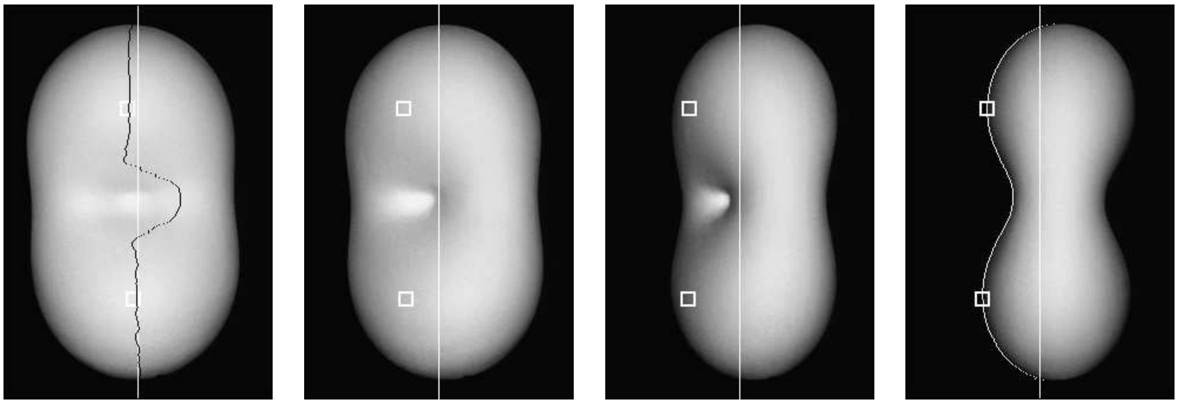


Figure 4.27: The surface orientation of the recovered cup



a. $image_0$ b. $image_{\pi/6}$ c. $image_{\pi/3}$ d. $image_{\pi/2}$

Figure 4.28: Four images from the image sequence of the pillow. The square box marks the image of a sample point. The white vertical line is the rotation axis. The white curve in (d) is the detected occluding contour. The black curve in (a) shows the $p = 0$ curve.

The depth and surface orientation on a $p = 0$ curve in $image_0$ (see black curve in Fig. 4.28 (a)) is recovered by using the search method which is used to recover the $p = 0$ curves on the cup. Then the depth and surface orientation is extended from every point on the $p = 0$ curve in the x direction. Figure 4.30 shows the recovered surface depth and orientation of the pillow. The surface orientation plot is the projection of the surface normals of the recovered surface. The sign for the q component is determined in the same way as we used for the q component of the cup. The surface normals are extracted from every fourth row and column in $image_0$. From the depth plot, we see that the surface of the pillow is recovered reasonably well in the area of strong interreflection although the recovered surface looks less concave than the manufactured pillow. The interreflection is considered the major cause which made the recovered surface less concave.

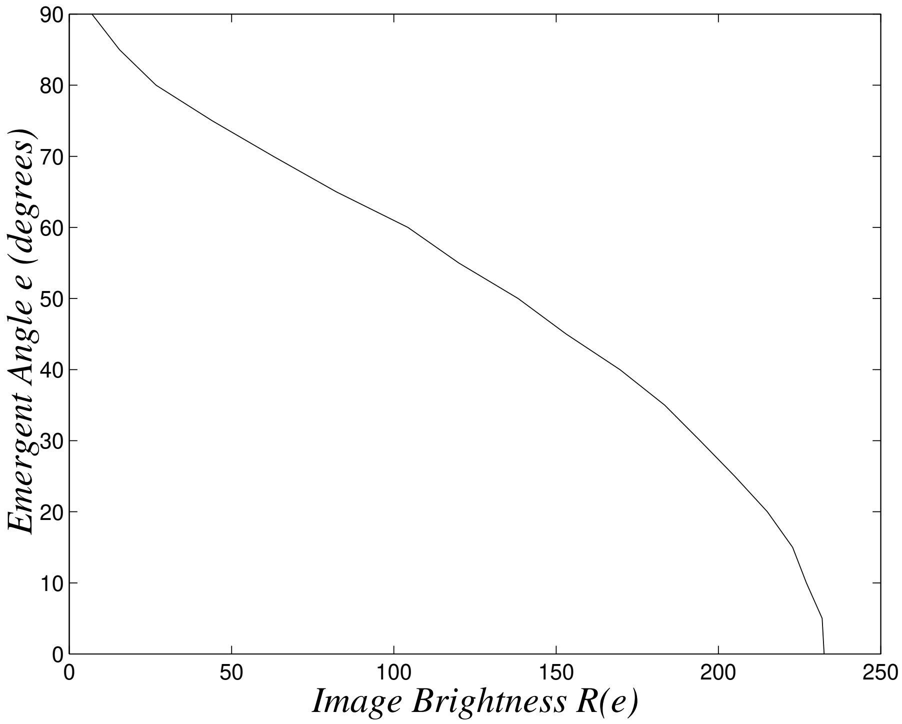


Figure 4.29: The surface reflectance function of the pillow.

4.9 Summary

We have presented the photogeometric technique for surface recovery from images of a rotating object illuminated under a collinear light source. The technique contains two parts: one is the surface reflectance function extraction and the other is surface recovery. We also have conducted some experiments with different objects. The results show that the technique is simple, feasible and effective.

The surface reflectance function under a collinear light source is easy to compute, easy to store and easy to use. Any surface point of known 3D location and surface orientation can be used as sample points for extracting the surface reflectance function. For the surface points whose images are singular points, it is easy to determine their 3D

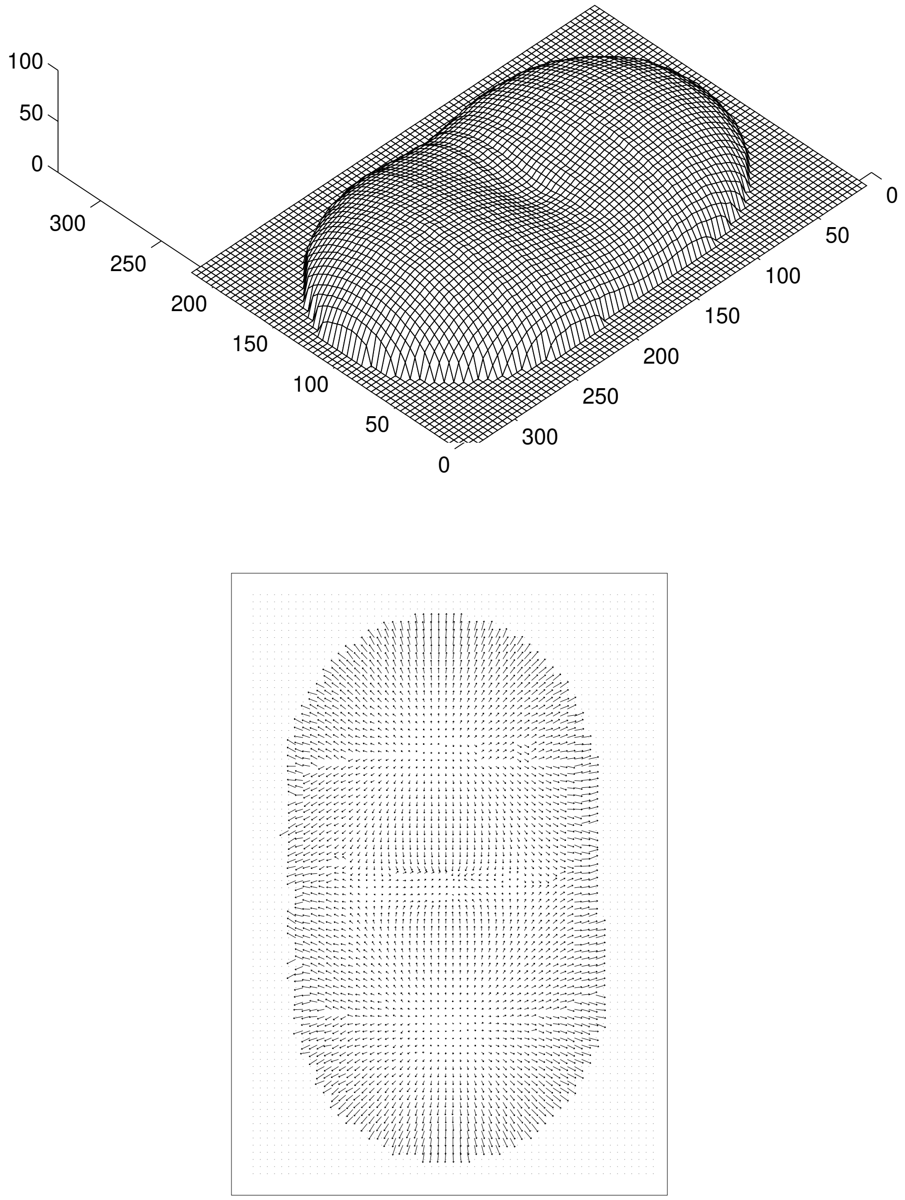


Figure 4.30: The surface depth and orientation of the recovered pillow.

locations and surface orientations. So these surface points are used as sample points. The search for singular points in images is linear with the number of pixels in an images. The maximum cost for finding the corresponding occluding contour of singular points is $O(mk)$ with m as the number of pixels in a row of an image and k as the number of the singular image points. The cost for extracting the brightness values of sample points from images is $O(vw)$ with v as the number of images and w as the number of the sample points used for extracting the surface reflectance function. For a uniform surface, only a few sample points and a limited number of images are needed to extract the surface reflectance function. So the total cost for extracting the surface reflectance function is linear with the image size. The cost can be reduced by using the results in the current row to conduct the computation in the next row. A look-up table of 255 entries can be used to store the inverse of the surface reflectance function. To compute the surface orientation from two brightness values, either the formulas 3.20 and 3.22 or a look-up table of $255 * 255$ entries can be used.

The surface recovery process consists of the three subprocedures. Starting from some points of known depth and iteratively applying the three subprocedures, the depth and surface orientation can be recovered on the whole object image. The surface recovery can start from any point of known depth. To make it easy to remove the depth ambiguity, surface is recovered on a $p = 0$ curve first. Two methods can be used to find the location, depth and surface orientation on a $p = 0$ curve. The first method only needs depths on some singular points and extends the computation from singular points along the y direction. The second method uses the corresponding occluding contour of a $p = 0$ curve as the depth for the curve and gives more accurate results. The two methods can be used at the same time for surface recovery on a $p = 0$ curve.

To recover surface orientation of a surface point, its two brightness values from two different images are required and the two image points corresponding to the surface point must be found. Since there is no surface marking on a uniform surface, to find the correspondence by matching points in two images is impossible. The matching is avoided by first computing the 3D coordinates of a surface point and then projecting the surface point on the two images.

When surface recovery is extended in the x direction from a $p = 0$ curve, there may be some areas which can't be recovered. To recover these areas, the depth and surface orientation on a $p = 0$ curve in an area are recovered and then extended in the x direction. If there is no $p = 0$ curve in the area, we can start from any point with known depth at the boundary or recover the corresponding surface by using the other images taken after the object has been rotated by different angles.

The size of the Gaussian filter used in surface recovery is small and about several pixels. Let n be the size of the Gaussian filter; then the time required for filtering one image of size N is $O(2nN)$. The Gaussian filtering can be implemented in real time by using hardware. The maximum computational cost in surface recovery on a $p = 0$ curve is linear with N. The maximum computational cost in extending the surface depth and orientation in the x direction is also linear with N. Therefore the total cost for surface recovery is linear with the image size N.

There are several factors such as noise, non-collinear illumination, non-uniform surface reflectance, and interreflection that may cause surface recovery error. Chapter 7 will give a detailed analysis of the effects of different factors on surface recovery.

Chapter 5

Wire-frame Technique

In Chapter 4, we described the photogeometric technique for surface recovery. The technique is capable of building the surface reflectance function of the object and recovering the depth and surface orientation of the object surface. The experimental results show that the technique is feasible and effective. The technique can be applied to any uniform surface of isotropic reflection. However, the assumption on the uniformity of the surface reflectance function is too restrictive to apply the technique for general surface recovery. Moreover, integrating surface gradients for depth recovery will accumulate errors caused by image noise, non-uniform albedo, and interreflection. To solve these problems, we further explore the constraints on image sequence of a rotating object. By integrating these constraints, we propose another technique, called wire-frame technique for surface recovery. In this technique, we extract a wire frame of 3D curves on the object surface. Then the whole object surface is recovered by interpolating the surface between curves on the wire frame. The interpolation can be done by using geometric or photogeometric methods. The photogeometric method is the application of the photogeometric technique we described in the previous chapter. The technique presented in this chapter does not need the surface reflectance function for extracting a wire frame. Moreover the technique makes it possible to deal with piecewise uniform objects by extracting the local reflectance function from the wire frame and then applying the local reflectance function for local surface recovery.

Section 5.1 in this chapter reviews the geometric and photometric constraints on a

$p = 0$ curve. Section 5.2 describes how to find the correspondence between a $p = 0$ curve and an occluding contour. Section 5.3 presents a search method which finds the 3D coordinates on a $p = 0$ curve by integrating the geometric and photometric constraints. Section 5.4 shows how to extract a wire frame from an image sequence. Section 5.5 and Section 5.6 demonstrates the experiments with surfaces of matte and specular reflectance. Section 5.7 summarizes this chapter.

5.1 Geometric and Photometric Constraints on a $p = 0$ Curve

In order to further explore the geometric and photometric constraints on the image sequence of a rotating object for surface recovery, we look at geometric and photometric constraints on a $p = 0$ curve. A $p = 0$ curve, after $\pi/2$ or $-\pi/2$ rotation of the object, will become an occluding boundary and be projected as an occluding contour on the image if there is no self-occlusion on the curve. The geometric information on the corresponding occluding contour provides the geometric information on the $p = 0$ curve (see Subsection 3.2.4). The x coordinates on the corresponding occluding contour give the depths on the $p = 0$ curve. To determine the 3D location of a $p = 0$ curve, we still need to find the x coordinate of each point on the curve. For a $p = 0$ curve on a surface illuminated under a collinear light source, each point on a $p = 0$ curve gives the same brightness value in two images taken after the surface has been rotated by two symmetric angles. This constraint is called *the equi-brightness constraint* and has been discussed in Subsection 3.3.3. Also, for any surface point, the disparity of its two image points in two images taken after the surface has been rotated by two symmetric angles can be calculated from its depth (see Subsection 3.2.5). The geometric or photometric constraint alone is not sufficient for determining the 3D location of a $p = 0$ curve. A search method which integrates both geometric and photometric constraints can be used to solve this problem.

5.2 Corresponding Occluding Contour of a $p = 0$ Curve

The correspondence from a $p = 0$ curve in $image_0$ to the occluding contour in $image_{\pi/2}$ can be found by using singular points in $image_0$ since a singular point in $image_0$ is a point on a $p = 0$ curve. As shown in Subsection 3.3.2, for a singular point, it is easy to find its depth and its corresponding occluding contour. The equi-brightness constraint can be used to check the correspondence from a singular point to its corresponding occluding contour point. If the correspondence is correct, the two image points, which are projected by using the depth at the singular point, should have the same brightness value in $image_\alpha$ and $image_{-\alpha}$. Under the C^2 surface assumption, a $p = 0$ curve and its corresponding occluding contour are smooth and continuous. So once the correspondence from a singular point to a contour point is built, the correspondence from a $p = 0$ curve to an occluding contour is determined.

5.3 Integrating Geometric and Photometric Constraints

Let (x_0, y_0, z_0) be a surface point on a $p = 0$ curve, and (x_α, y_α) and $(x_{-\alpha}, y_{-\alpha})$ be the images of the surface point in $image_\alpha$ and $image_{-\alpha}$, respectively. Then $y_\alpha = y_{-\alpha}$. In Section 3.2.5, we have shown that the disparity D of the two image points is

$$D = -2z_0 \sin\alpha, \tag{5.24}$$

and the depth z_0 can be measured from the corresponding contour point in $image_{\pi/2}$. The two image points, (x_α, y_α) and $(x_{-\alpha}, y_{-\alpha})$, must satisfy the two constraints:

$$x_\alpha - x_{-\alpha} = D \quad \& \quad E(x_\alpha, y_\alpha) = E(x_{-\alpha}, y_{-\alpha}).$$

Three images, $image_{\pi/2}$, $image_{\alpha}$ and $image_{-\alpha}$, are used in search for the x location of a $p = 0$ curve. The same row in the three images are used to search for the two image points, which satisfy the above two constraints, in $image_{\alpha}$ and $image_{-\alpha}$. Then the 3D location of the $p = 0$ point is calculated from the locations of the two corresponding image points. The search and calculation can be parallel by rows. But to reduce the search space, the result in the current row is used to conduct the search in the next row so that the search can be confined within a small window. The search starts from image points corresponding a singular point in $image_0$. The following procedure inputs the same row from the three images and outputs the location of a $p = 0$ point with subpixel accuracy. The procedure assumes that, for the two corresponding image points which satisfy the two constraints, the point in $image_{\alpha}$ is within a 1D window. For each row in the three images, do

1. In $image_{\pi/2}$, find the occluding point which corresponds to the $p = 0$ point. Measure the distance from the occluding point to the rotation axis. The distance gives the depth z_0 for the $p = 0$ point. Compute disparity D by using Equation 5.24. Pick the first image point in the window in $image_{\alpha}$;

2. For the image point in $image_{\alpha}$, use the disparity D calculated in step 1 to locate the corresponding image point in $image_{-\alpha}$;

3. Compute the brightness difference of the two corresponding image points. If the sign of the brightness difference is different from the brightness difference computed at the previous point, calculate the x locations of the two corresponding image points, which satisfies the two constraints, in $image_{\alpha}$ and $image_{-\alpha}$ with subpixel accuracy and then compute the x location for the $p = 0$ point by

$$x_0 = \frac{x_{\alpha} + x_{-\alpha}}{2\cos\alpha},$$

where x_α and $x_{-\alpha}$ are the x coordinates of the two corresponding image points obtained by searching. Otherwise, pick the next image point in the window in $image_\alpha$ and go to step 2.

In step 3, linear interpolation is used to obtain subpixel accuracy in the locations of the two corresponding image points. In the general case, the two image points which satisfy the above two constraints lie between image pixels. The sign of the brightness difference of the two image points is checked when the search is going on. Let $\ddot{x}_\alpha$ and $\dot{x}_\alpha$ be the x coordinates of two adjacent image points in $image_\alpha$ with $\ddot{x}_\alpha = \dot{x}_\alpha + 1$, and $\ddot{x}_{-\alpha}$ and $\dot{x}_{-\alpha}$ be the x coordinates of two image points in $image_{-\alpha}$ with $\ddot{x}_{-\alpha} = \ddot{x}_\alpha - D$ and $\dot{x}_{-\alpha} = \dot{x}_\alpha - D$. We will drop the y coordinate in the following expressions since all the image points used in the calculation have the same y coordinate. If the sign of $E(\ddot{x}_\alpha) - E(\ddot{x}_{-\alpha})$ is different from the sign of $E(\dot{x}_\alpha) - E(\dot{x}_{-\alpha})$ and the image brightness values between two adjacent image points can be linearly interpolated, the x coordinates, x_α and $x_{-\alpha}$, of the two image points which satisfy the two constraints can be computed with subpixel accuracy by the following calculations. Using linear interpolation, we have

$$E(x_\alpha) = E(\dot{x}_\alpha) + (E(\ddot{x}_\alpha) - E(\dot{x}_\alpha))(x_\alpha - \dot{x}_\alpha) = E(\dot{x}_\alpha) + \Delta E(x_\alpha)(x_\alpha - \dot{x}_\alpha) \tag{5.25}$$

$$E(x_{-\alpha}) = E(\dot{x}_{-\alpha}) + (E(\ddot{x}_{-\alpha}) - E(\dot{x}_{-\alpha}))(x_{-\alpha} - \dot{x}_{-\alpha}) = E(\dot{x}_{-\alpha}) + \Delta E(x_{-\alpha})(x_{-\alpha} - \dot{x}_{-\alpha}), \tag{5.26}$$

where $E(\ddot{x}_\alpha)$, $E(\dot{x}_\alpha)$, $E(\ddot{x}_{-\alpha})$ and $E(\dot{x}_{-\alpha})$ are the image brightness values at the corresponding image points; $\Delta E(x_\alpha) = E(\ddot{x}_\alpha) - E(\dot{x}_\alpha)$ and $\Delta E(x_{-\alpha}) = E(\ddot{x}_{-\alpha}) - E(\dot{x}_{-\alpha})$. Since the two image points must satisfy the equi-brightness constraint,

$$E(\dot{x}_\alpha) + \Delta E(x_\alpha)(x_\alpha - \dot{x}_\alpha) = E(\dot{x}_{-\alpha}) + \Delta E(x_{-\alpha})(x_{-\alpha} - \dot{x}_{-\alpha}). \tag{5.27}$$

Substituting $x_\alpha - D$ for $x_{-\alpha}$ and solving the equation for x_α, we have

$$x_\alpha = \frac{E(\dot{x}_{-\alpha}) - E(\dot{x}_\alpha) + \Delta E(x_\alpha)\dot{x}_\alpha - \Delta E(x_{-\alpha})(D - \dot{x}_{-\alpha})}{\Delta E(x_\alpha) - \Delta E(x_{-\alpha})}. \tag{5.28}$$

Then $x_{-\alpha}$ can be calculated by $x_\alpha - D$.

5.4 Extracting Wire Frame

Since the coordinate reference system is fixed with the viewing direction, a curve of $p = \tan(-\theta)$ on an object surface can be transformed to a curve of $p = 0$ by rotating the object by θ. So the 3D location and surface orientation of a curve of $p = \tan(-\theta)$ can be determined in the same way. As the depth and x location of a curve of $p = \tan(-\theta)$ computed in this way are the depth and x location at image points in $image_\theta$, a curve of $p = tan(-\theta)$ is defined as *a* $p = 0$ *curve in* $image_\theta$. For example, a $p = 0$ curve in $image_{\pi/9}$ means the computed depth and x location of the curve are at image points in $image_{\pi/9}$. Let $n = 2\pi/\theta$ and θ divide 2π, and let the images of an object be taken after the object has been rotated by $k\theta$, $(0 \leq k < n)$. In total, n images are taken during 2π rotation. To extract a $p = 0$ curve in $image_\theta$, three images, $image_{\theta+\pi/2}$, $image_{\theta+k\theta}$ and $image_{\theta-k\theta}$, ($k\theta < \pi/2$) can be used. All the curves extracted from images comprise a wire-frame on the object surface. The depth and surface orientation between two adjacent curves on the wire-frame can be interpolated by using either geometric or photogeometric method.

5.5 An Experiment with a Matte Surface

5.5.1 Image sequence

The imaging condition for our wire-frame technique is the same as that for the photogeometric technique. Thirty six images are taken during 2π rotation of an object with a $\pi/18$ increment for each image. The images are smoothed with a Gaussian filter of $\sigma = 1$ to filter noise and quantization effects. A clay ellipsoid is used for our first experiment. The surface of the ellipsoid can be considered a matte surface. Twelve images from the image sequence of the ellipsoid are shown in Fig. 5.31.

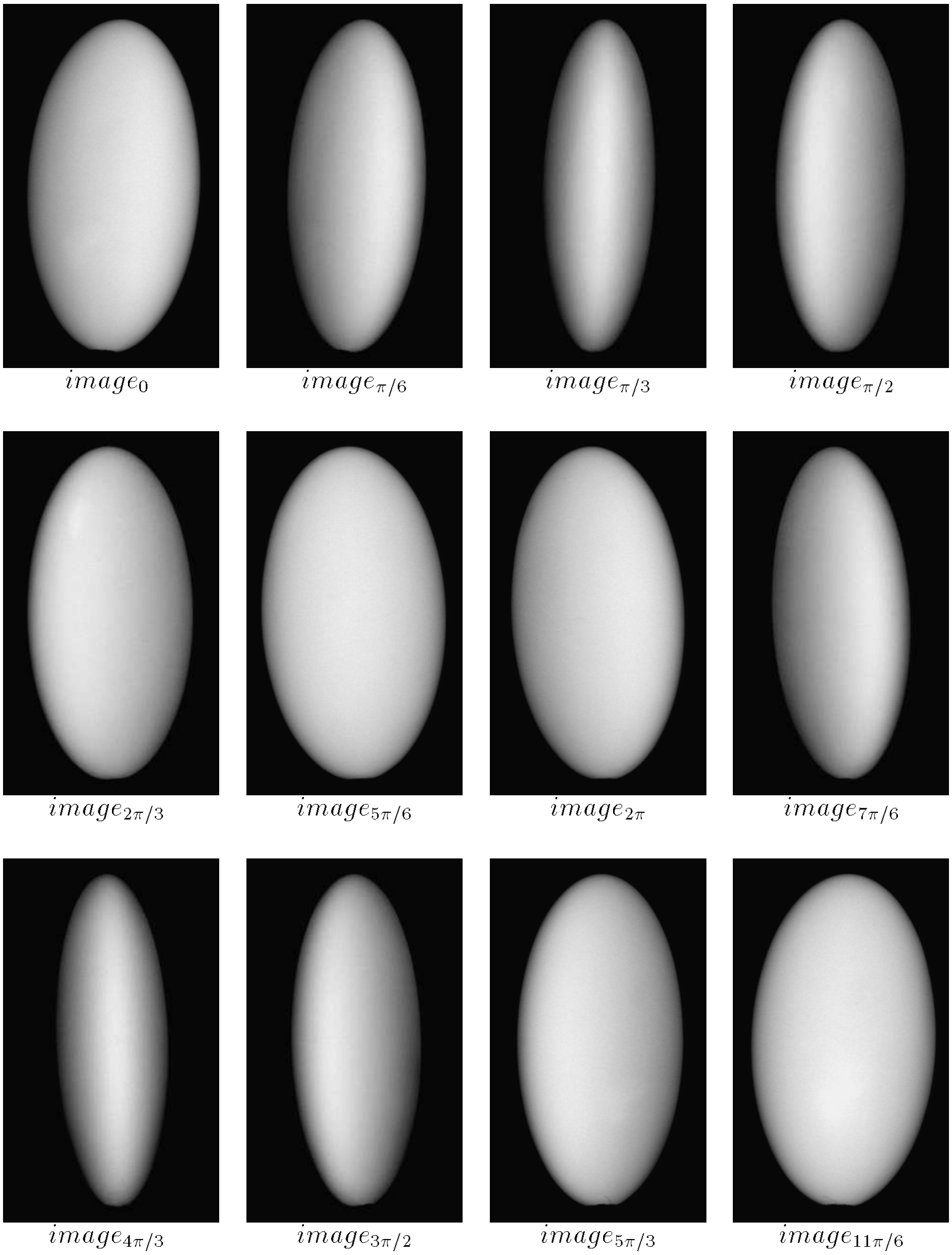


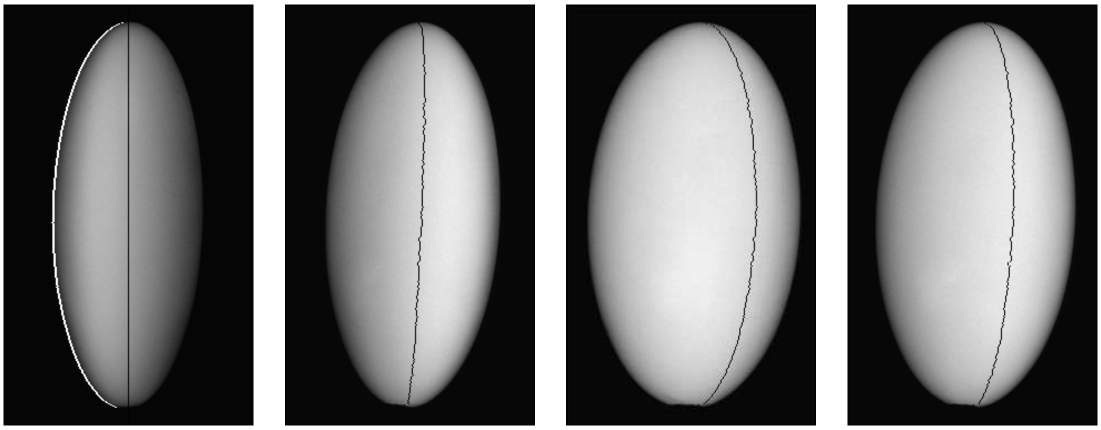
Figure 5.31: Images from an image sequence of a rotating ellipsoid.

5.5.2 Wire frame on an ellipsoid

To extract the $p = 0$ curve in $image_\theta$, three images taken after $(\theta + \pi/2)$ mod 2π, $(\theta - \alpha)$ mod 2π, $(\theta + \alpha)$ mod 2π rotation are used. In our experiment, $\alpha = \pi/9$. For example, to extract the $p = 0$ curve, $image_{\pi/2}$, $image_{\pi/9}$ and $image_{17\pi/9}$ are used. These three images are shown in Fig. 5.32 (a)–(c). The black vertical line in image (a) is the rotation axis. The occluding contour in image (a) is extracted by a simple thresholding method. The horizontal distance from each contour point to the rotation axis gives the depth for the corresponding point on the $p = 0$ curve. For each point on the $p = 0$ curve, the disparity of its two image points in $image_{\pi/9}$ and $image_{17\pi/9}$ is calculated by Equation 5.24 and the x coordinates of its two images in $image_{\pi/9}$ and $image_{17\pi/9}$ are determined by the search method we described in the previous section. The black curve in image (d) is the projection of the $p = 0$ curve on $image_0$. The other curves of constant p are extracted in the same way. All these curves comprise a wire frame on the object surface. The projection of the wire frame on $image_0$ is shown in Fig. 5.33 (a).

5.5.3 Geometric interpolation

Each curve on the wire frame is computed as a $p = 0$ curve from images taken after the object has been rotated by certain angles. The q component on a $p = 0$ curve is computed from the tangent on the corresponding contour. A proper transformation is used to transform the 3D location and surface orientation of a $p = 0$ curve to its initial coordinates. The surface orientation and depth between adjacent curves on the wire frame are geometrically interpolated by using depth and orientation data on the wire frame. The interpolation is implemented with graphics program Optik [1]. First we connect the areas between each two adjacent curves with triangles and get a triangular mesh of the surface(see Fig. 5.33 (b)). Then we input the 3D coordinates and surface orientations at



a. $image_{\pi/2}$ b. $image_{\pi/9}$ c. $image_{17\pi/9}$ d. $image_0$

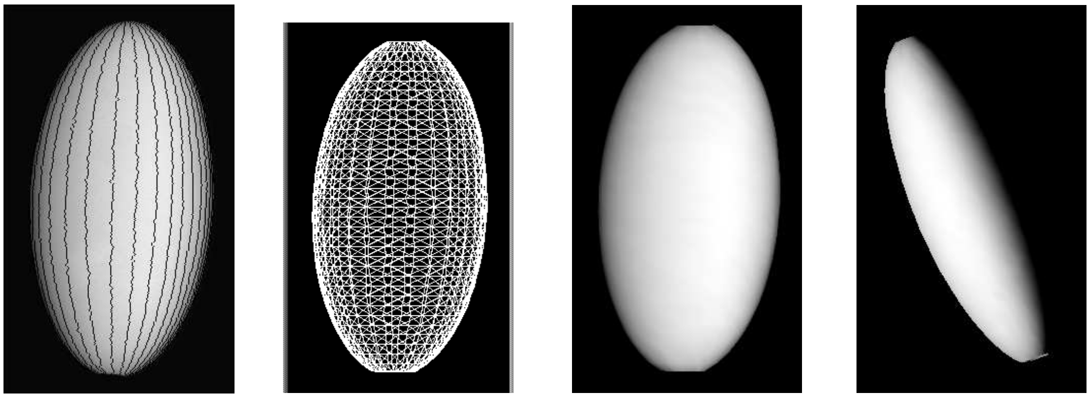
Figure 5.32: Extracting a $p = 0$ curve from images (a)-(c). The black line in (a) is the rotation axis and the white curve is the occluding contour which gives the depth for the corresponding $p = 0$ curve. The black curves in (b) and (c) are the locations of the $p = 0$ curve obtained by the search method. The black curve in (d) is the projection of the computed $p = 0$ curve on $image_0$.

the vertices of the mesh to the program Optik. Optik does surface orientation interpolation and gives a smooth shaded image of the recovered surface. In Fig. 5.33, (c) and (d) are the synthetic images of the recovered surface. The pose of the ellipsoid in image (c) is the same as that in $image_0$. The viewing direction and illuminant direction used to generate image (c) are collinear so that we can compare the synthetic image with the corresponding real image $image_0$. The viewing and illuminant direction as well as the pose of the ellipsoid for synthetic image (d) are different from those for the real images.

5.6 An Experiment with a Specular Surface

5.6.1 Wire frame on a porcelain cup

We also experimented with a porcelain cup. The surface reflectance of the cup contains a strong specular component but this does not affect the wire frame extraction.



a. Wire frame b. Triangular mesh c. Shaded image 1 d. Shaded image 2

Figure 5.33: (a) the wire frame extracted from images. (b) the triangular mesh built by connecting points on the wire frame. (c) and (d) synthetic images generated under different viewing and illuminant conditions.

An image sequence of thirty six images are taken during 2π rotation with a fixed incremental angle between two successive images. Twelve images from the image sequence of the rotating cup are shown in Fig. 5.34. Since the object is not a simple convex object, there may be more than one $p = 0$ curves in any image. Three images, image (a), (b) and (c) in Fig. 5.35, are used to extract the $p = 0$ curves on the cup. In this experiment, both background-occluding contours and self-occluding contours are used. The occluding contours are detected by the method described in Subsection 3.2.3. Image (a) shows the occluding contours in $image_{\pi/2}$ as the white curves. One of the contours in image (a) is a self-occluding contour and corresponds to the $p = 0$ curve on the handle. The x coordinates of these white curves give depths for the corresponding $p = 0$ curves. Images (b) and (c) in Fig. 5.35 are images taken after $\pi/9$ and $17\pi/9$ rotation of the object, respectively. The projection of the $p = 0$ curves in $image_{\pi/9}$ and $image_{17\pi/9}$ are determined by searching for the image points which satisfy the disparity and equi-brightness constraints. The black curves in (b) and (c) of Fig. 5.35 are projections of the $p = 0$

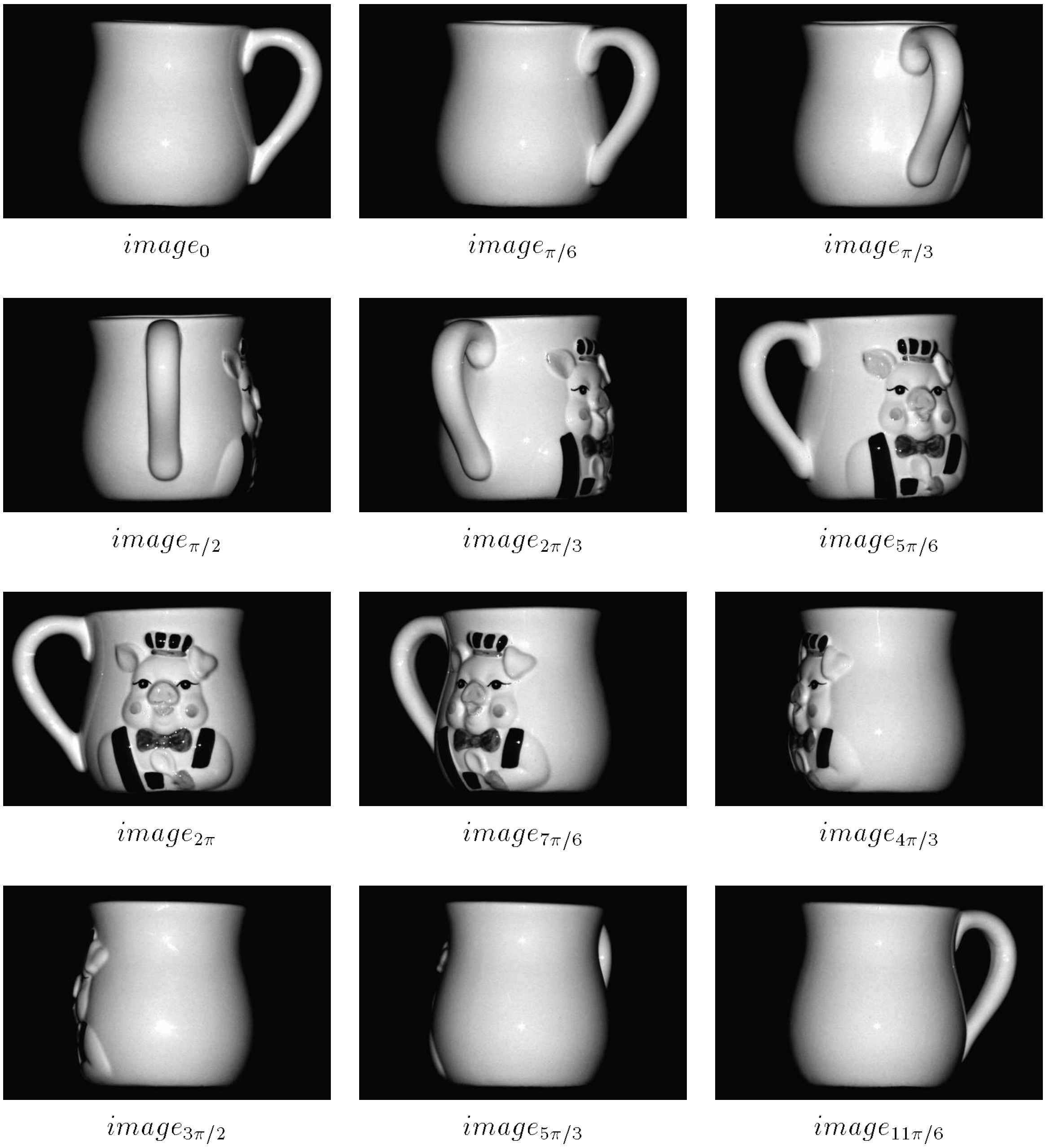
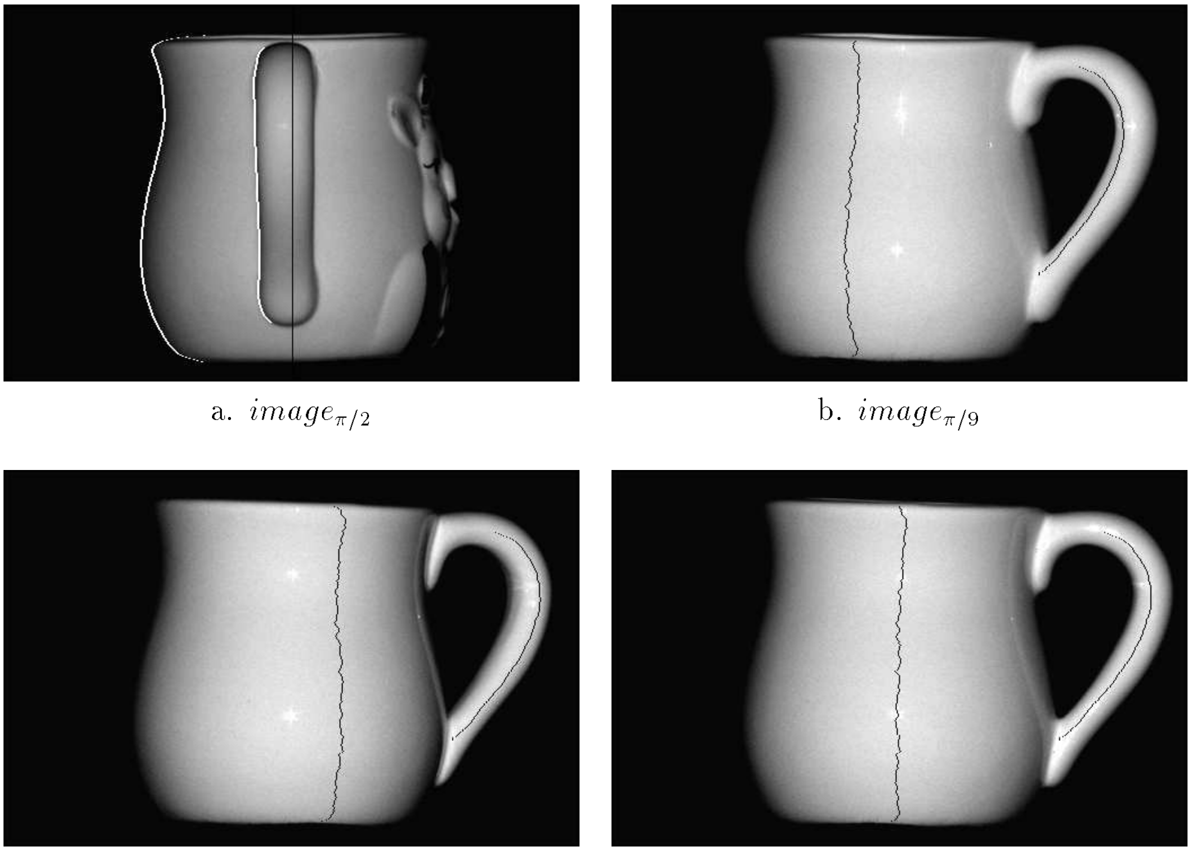


Figure 5.34: Images from an image sequence of a rotating cup.



a. $image_{\pi/2}$ b. $image_{\pi/9}$

c. $image_{17\pi/9}$ d. $image_0$

Figure 5.35: Extracting $p = 0$ curves from images. The black line in (a) is the rotation axis and the white curves are the occluding contours which give the depths for the corresponding p =0 curves. The black curves in (b) and (c) are the locations of the $p = 0$ curves obtained by the search method. The black curves in (d) are the projection of the computed $p = 0$ curves on $image_0$.

curves in $image_{\pi/9}$ and $image_{17\pi/9}$. The black curves in Fig. 5.35(d) are the two $p = 0$ curves in $image_0$. The $p = 0$ curves in other images are extracted in the same way. These extracted curves comprise a wire frame on the object surface. The projections of the wire frame on images are shown in Fig. 5.36. We can see that some curves are not completed and some curves are missing. This is because the corresponding occluding contours are partially occluded or totally occluded by other part of the object. It is notable that a curve on the handle is totally missing. This is because the occluding contour point which corresponds to the singular image point on the curve cannot be detected. There is a pig figure on one side of the cup. The protrusion and the color of the figure violates our uniform and smooth surface assumption so that the curves on the body part of that side cannot be extracted.

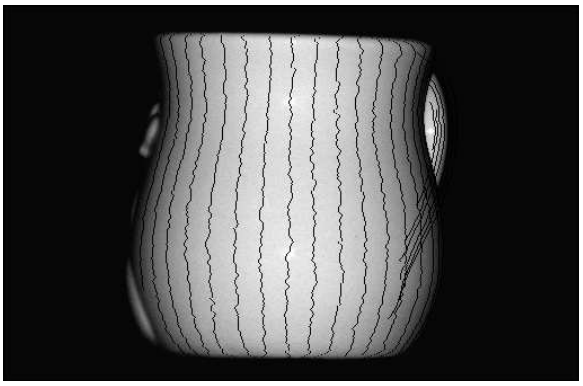
5.6.2 Photogeometric interpolation

Since the wire frame on the cup surface is incomplete, some details of the surface will be lost in recovery if geometric interpolation is used. For example, there is not sufficient geometric information on the wire frame to interpolate the joints between the handle and the body of the cup. Photogeometric interpolation is the application of the photogeometric technique described in Chapter 4. Instead of starting from a few 3D surface points, the surface recovery here starts from a wire frame. To apply the photogeometric technique, the surface reflectance function of the cup must be known. It is extracted by using the same method as we described in Section 4.2. Figure 5.37 shows the reflectance function extracted from the images of the rotating cup.

Photogeometric interpolation uses two images taken at different rotation angles of the object and recovers the depth and surface orientation at every pixel in one of the two images. The interpolation starts from image points on the wire frame. The computation for depth and surface orientation is propagated in the x direction until an image point on

projected on $image_0$

projected on $image_{\pi/6}$



projected on $image_{5\pi/3}$

projected on $image_{11\pi/6}$

Figure 5.36: Projections of the wire frame on the original images.

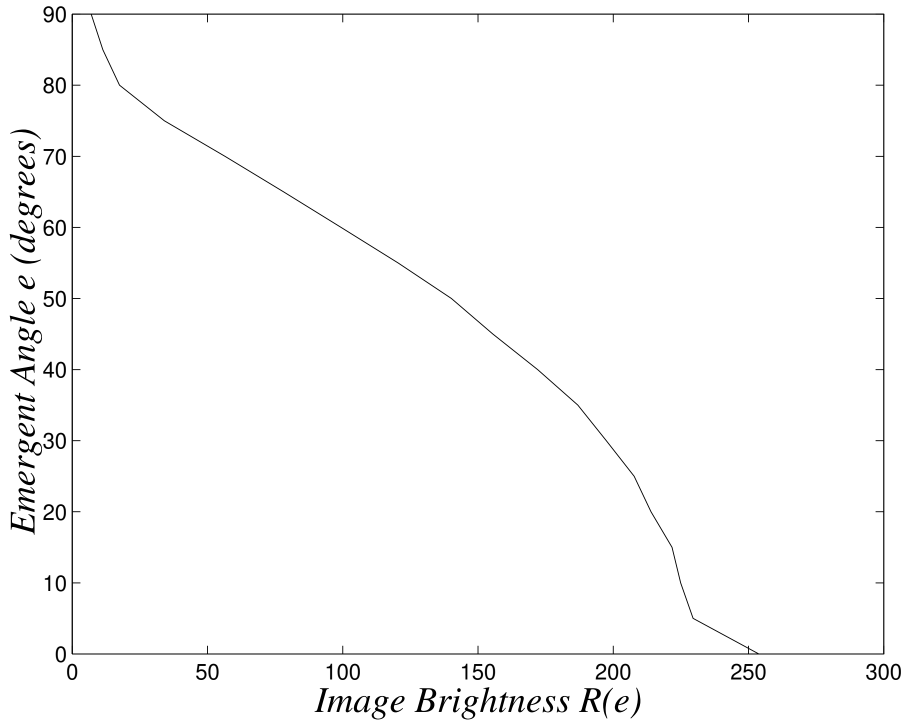


Figure 5.37: The surface reflectance function of the cup.

another curve of the wire frame is reached. In this way, the depth and surface orientation between the curves of the wire frame can be recovered. Because of the accumulated error caused by image noise and other facts, the depth calculated at the end point may not meet the depth on the wire frame. This will cause a discontinuity in the depth. To overcome this problem, the *distance-weighted averaging* method described in Subsection 4.6.2 is used. Figure 5.38 shows the surface depth plot of the the recovered cup. The depth data is directly used for display without any smoothing or regularization. Figure 5.39 is the surface orientation map of the recovered cup.

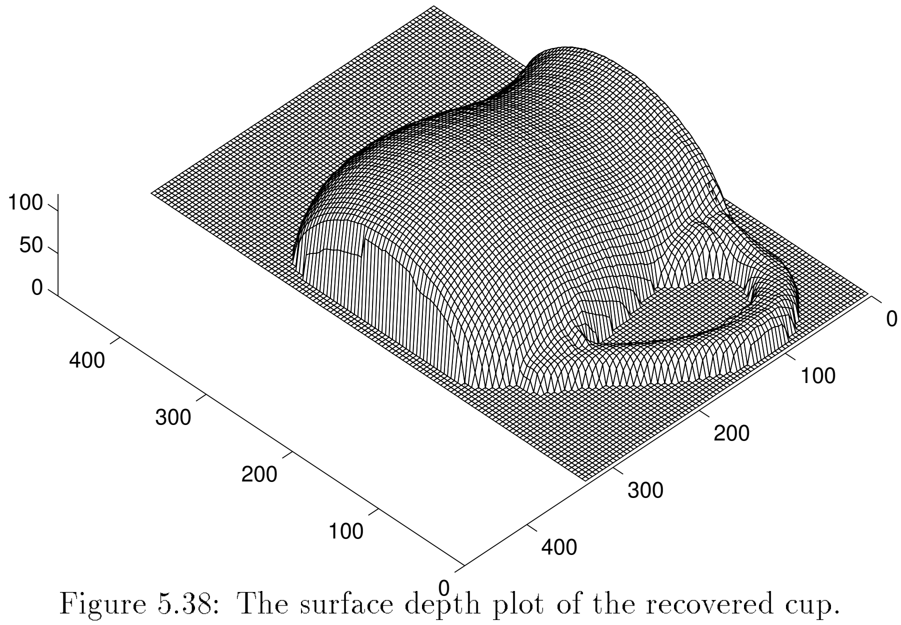


Figure 5.38: The surface depth plot of the recovered cup.

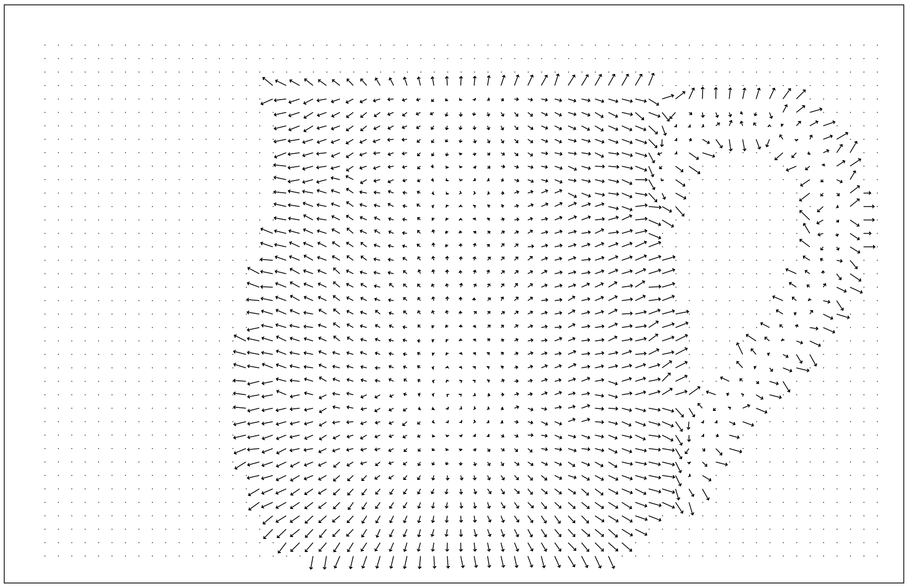


Figure 5.39: The surface orientation plot of the recovered cup.

5.7 Summary

The wire-frame technique has been presented in this chapter. The technique extracts a 3D wire frame on an object surface from images without using the surface reflectance function. The experimental results show the technique is simple and reliable. The computational cost of the technique depends on the image size and the number of curves on the wire frame to be extracted. The maximum cost for detecting singular points and occluding contours in an image is linear with the image size. The maximum cost for determining the 3D coordinate on a $p = 0$ curve by searching for image points which satisfy the given disparity and equi-brightness constraint is linear with the image size. So the maximum cost for extracting a wire frame of k curves of constant p from images of size N is linear with kN. The cost can be reduced by using the results in the current row for the computation and search in the next row.

The most important aspect of the technique is that the surface recovery on the wire frame is local. The computation is local so that there is no accumulated error. The information used is local so that the object surface can be piecewise uniform. As depth and surface orientation are recovered at every point on the wire frame, the brightness values of the points on the wire frame can be used to build the surface reflectance function even when no singular image points are available from images. Then the surface reflectance function extracted can be used in the photogeometric technique for surface recovery. By tracking the surface points on the wire frame the surface reflectance function for the local area can be built and then be applied for surface recovery for the local area as long as the surface can be segmented into different areas of uniform reflectance.

To extract the 3D coordinates on a $p = 0$ curve, the curve must be visible in $image_{\pi/2}$, $image_{\alpha}$, and $image_{\alpha}$. However, the curve may be partially or totally occluded by the other part of the object so that they are partially or totally not visible in these images.

In this case, only a part or none of any part of the $p = 0$ curve can be extracted. For example, for any surface curve within a concave area, it can not be seen as an occluding contour in images. Therefore, a $p = 0$ curve within a concave area cannot be extracted by the wire frame technique.

In Chapter 7, we will show that a collinear light source is not necessary for extracting the wire frame. We will also give detailed error analysis on the 3D location of the extracted wire frame.

The next chapter will discuss some practical issues related to the experimental conditions. The chapter can be skipped without losing the continuity of this thesis.

Chapter 6

Experimental Considerations

This chapter describes some practical problems with the experimental conditions. It can be skipped since it is not necessary for the understanding of the other chapters of this thesis. The first section talks about the imaging facility we used for surface recovery. The second section introduces a simplified camera model. The third section presents a calibration method for overcoming photometric distortion. The fourth section discusses geometric aberration of the camera. The last section describes how to determine the projection of the rotation axis in images.

6.1 Calibrated Imaging Facilities

The imaging facility we used is the calibrated imaging facility (CIF) built in our lab. The CIF has been used for photometric stereo [82, 38], photometric optical flow [81, 58] and, several other projects [45]. The imaging facility is isolated in a dark room surrounded by black curtains. The black curtains block light from outside and absorb the ambient light reflected inside the dark room. It also gives a uniform dark background on the images of an object. The base of the CIF is an optical bench on which apparatus can be mounted at various positions and pointed in different directions. The camera is mounted on a pedestal on the optical bench through a support so that its 3D position and viewing direction can be adjusted. There is a motion stage on the optical bench consisting of a translation mechanism and a rotary table so that object can be translated and rotated on a horizontal plane. The rotary table is controlled by a computer. It can be rotated in any direction by any amount of angle. The rotation angle can be controlled with an accuracy of one thousandth of a degree.

The camera of the CIF is a 3CCD Sony camera with DCX-755 control unit. The zoom lens has a wide range of focal lengths from wide angle to telephoto mode. The camera provides three B&W images which correspond to the three images from red, green and blue channels. Since color information is not needed in our surface recovery, only B&W images from one channel are used. The camera control unit is used to adjust the iris of the camera and to control the gain of the video output from the camera. The unit can also be used to set the gamma correction of the video output on and off.

There are three directional light sources available(red, green and blue). Since we only need B&W images from one color channel, only one of them is used. These light sources are focused by a set of optical lenses so that the outcoming light rays are projected in parallel or almost in parallel. The light sources are designed to make the radiation over the illuminated area as uniform as possible. Ideally, the illuminant direction of the light source and the viewing direction of the camera should be aligned along a line. We considered putting a half-silvered mirror in front of the camera to make the viewing direction and illuminant direction precisely collinear. However, the half-silvered mirror reflects only a small portion of the light to the object and blocks a large portion of the reflection from the object surface to the camera. As the half-silvered mirror does more harm than help, we just directly put the light source on the top of the camera. The distance from the object under recovery to the camera and the light source is about 242cm and the distance between the optical center of the camera and the optical center of the light source is about 8cm. So the angle between the optical axis of the camera and the light source is about 0.033 radians. This angle violates the collinear light source assumption. However the error caused by this small angle is not serious. The effects of this small angle will be analyzed in Section 7.4.2.

An image sequence of an object is taken when the object is rotating on the rotary table. In taking an image sequence of the object, the iris, the focus, the gain control of the camera are fixed and the gamma correction is turned off. Since the object is far away, the camera is set to telephoto mode. The action of rotating an object and taking images of the object are well synchronized by a computer program. The accuracy of the rotation control is within 1 thousandth of a degree. Usually, each image is taken after the object has been rotated by a fixed angle increment. The video output from camera is digitized by a frame grabber of Max-Video200 image processor and from the frame grabber the images can be downloaded and saved as digital image files. A digital image is stored in a file as a 512x480 8 bit array. Therefore the maximum brightness value in an image is 255.

6.2 Simplified Camera Model

The light source we used has ±10% of spatial variation in radiation uniformity according the specification. This violates the assumption on the uniformity of the light source. In addition, the vignetting effect of the camera [30] makes brightness fade away from the center to the boundary of an image. To overcome these problems, a calibration process is used. The calibration process is based on a simplified linear camera model.

The digital image used in computer vision is a two dimensional function $E(x, y)$ where both the domain and range of E are digitized. A CCD camera has a two dimensional array of photo sensors. When light (photo flux) hits a photo sensor, electrons are generated. If the camera shutter speed is fixed and the sensor is exposed to the light in a unit time defined by the fixed shutter speed, the number of electrons generated is kL, where the factor k is the quantum efficiency (electrons generated/incident photon flux) and L is the irradiance of the incident light over the photo sensor. The image brightness value at an image point is determined by the electrons collected at the corresponding photo sensor. At each photo sensor, there are also some free electrons produced by thermal energy. These free electrons are called dark current and they are highly temperature dependent. If we ignore the shot noise (noise generated by quantum nature of light), the number of electrons collected at each sensor is $kL + D$, with D as the dark current. For a CCD sensor array, the dark current D is a constant at certain temperature. After electrons are collected at each photo sensor, they are transformed into a voltage signal, amplified by the camera output circuit, and digitized by a frame grabber. In the frame grabber, the brightness value of an image point is expressed as $E = g(kL + D)$ with g as the gain of the output amplifier. The quantity gD can be determined from the brightness values in the images which are taken by blocking light to the camera with a lens cap.

In a camera with a compound lens, some light rays that pass through a lens may be blocked by the lenses behind. The entering ray with larger inclination is more likely to be blocked by some lenses. This effect is called vignetting [30]. As a result, even when the irradiance is uniform over the front lens of the camera, the sensors away from the optical axis of the camera gather less light than the sensors close to the optical axis. The vignetting of the camera is related to the configuration of the camera lens. When the iris and the focal length of a camera are smaller, the vignetting effect is more severe. If the iris and focal length are fixed, the vignetting effects purely depend on the physical location of each photo sensor. We model the vignetting effect by using a vignetting factor for each photo sensor. Assuming the irradiance is uniform over the front lens of a camera, the ratio of the irradiance over the photo sensor to the irradiance over the front lens is defined as the vignetting factor h.

When an object is illuminated by a light source, the surface reflection in the direction of the camera at the surface point whose image is image point (x, y) can be written as

$$R(x,y) = I(x,y)Q(x,y).$$

$I(x, y)$ is the irradiance of the light source at the corresponding surface point and $Q(x, y)$ is the surface reflectance function under a unit irradiance. For a uniform surface, Q is a function of the incident angle i, emergent angle e and phase angle g at a surface point. Assuming a photo sensor in the CCD array corresponds an image point. then the brightness value $E(x, y)$ of image point (x, y) can be expressed as

$$E(x,y) = g[k(x,y)h(x,y)R(x,y) + D] = g[k(x,y)h(x,y)I(x,y)Q(x,y) + D]. \quad (6.29)$$

Quantities with properties that depend on location (x, y) are explicitly indicated. In the above formula, $k(x, y)$ is the quantum efficiency at the corresponding photo sensor, $h(x, y)$ is the vignetting factor which accounts for vignetting effect of the camera.

The above imaging model does not consider the spectrum of the light source since the model is a simplified model and also the light source we used is filtered with a very narrow band filter. The model also does not consider shot noise, amplifier noise, signal transfer noise and digitization noise either because the effects of these noises are equivalent to a uniform distributed Gaussian noise. The noise can be filtered by a Gaussian filter. In deriving the imaging model, we assume the surface reflectance is uniform. But the model also applies to the object of non-uniform surface.

6.3 Photometric Calibration

From the term $k(x, y)h(x, y)I(x, y)$ in Formula 6.29, we see that because of the different quantum efficiency of the photo sensor, the vignetting effects of the camera, non-uniform illumination of the light source, the image brightness value of a surface point is not a simple function of the surface orientation. A single calibration process can be used to remove all the above effects.

The calibration is based on the fact that when the camera configuration and light source are fixed the value for $k(x,y)h(x,y)I(x,y)$ will be fixed at each image point and it is invariant to the time and the object surface. If the value for $k(x,y)h(x,y)I(x,y)$ at each image point can be calculated, the brightness value can be adjusted to remove these effects. To calculate $k(x,y)h(x,y)I(x,y)$, a uniform flat board is used. The surface reflectance $Q(x,y)$ is a constant for the uniform flat board. The image of the calibration board is called the calibrating image and the brightness value $E_b(x,y)$ of the calibrating image is

$$E_b(x,y) = g[k(x,y)h(x,y)I(x,y)C + D]$$

with C as the surface reflectance of the uniform flat board. Then we have

$$gk(x,y)h(x,y)I(x,y)C = E_b(x,y) - gD.$$

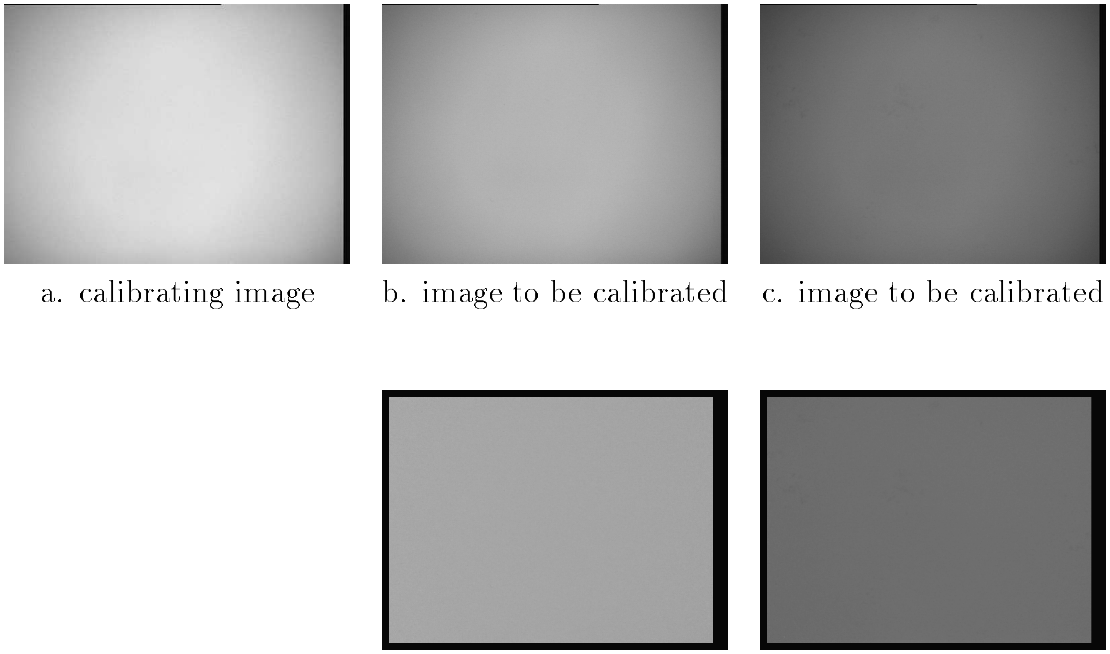
For calibration, the quantity $gk(x,y)h(x,y)I(x,y)C$ is sufficient. Assume we have another generic object of smooth curved surface. Suppose the reflectance of the surface is uniform and the surface reflectance function is $Q_s(x,y)$. If the image of the object is taken with the same camera configuration and the same illuminant condition, the brightness value of the object image is

$$E_s(x,y) = gk(x,y)h(x,y)I(x,y)Q_s(x,y) + gD.$$

Using the calibrating image to calibrate the image of the generic object, the brightness value $\bar{E}_s(x,y)$ of the calibrated image is

$$\bar{E}_s(x,y) = \frac{E_s(x,y) - gD}{E_b(x,y) - gD} + gD = \frac{gk(x,y)h(x,y)I(x,y)Q_s(x,y)}{gk(x,y)h(x,y)I(x,y)C} + gD = \frac{Q_s(x,y)}{C} + gD.$$

Since $Q_s(x,y)$ depends only on surface orientation, so does the brightness value of the calibrated image $\bar{E}_s(x,y)$. This is the desired property which is consistent with our



a. calibrating image　b. image to be calibrated　c. image to be calibrated

b'. calibrated image　c'. calibrated image

Figure 6.40: (a) is an image of white board used for calibrating other images. (b) and (c) are the images of a gray board and a yellow board before calibration. (b') and (c') are the calibrated images of (b) and (c).

assumptions. After calibration, the effects of different quantum efficiency, vignetting and non-uniform illumination are eliminated. The term $E_b(x, y) - gD$ can be scaled to avoid the brightness values of the calibrated image exceeding the maximum possible brightness value. The scale factor can be determined from the maximum brightness value in the images to be calibrated.

The calibration board is made of a piece of thin glass painted on one side. The flatness and uniformity of the board are very important. Otherwise, additional errors will be introduced by the calibration process. We have tried flat surfaces of different materials, such as cardboard, plastic, and metal. None of them is as uniform and flat as painted glass boards. Before taking an image sequence of the object to be recovered, we adjust the iris, focal length and focus of the camera as well as the illuminant intensity of the light source. With the fixed imaging condition, the images of the calibration board and the object are taken. When we take the image of the calibration board, the unpainted side of the calibration board is facing the camera. To avoid specular reflection, the surface normal of the calibration board is tilted a little bit with respect to the viewing direction. This does not affect the calibration very much. But to make the calibration more accurate, another image of the calibration board is taken with the surface normal tilted by the same amount but in the opposition direction. The average of these two images is used as the calibrating image. Before the calibration, a Gaussian filter is used to filter the image noise in the calibrating image. Then the image sequence of an object taken under the same imaging conditions is calibrated by the calibrating image.

The calibration process is evaluated by calibrating other flat surfaces of uniform reflectance. These flat surfaces are also made of thin glass but painted with different colors. In Fig. 6.40, (a) is the calibrating image, (b) and (c) are images of a flat yellow board and a flat gray board, (b') and (c') are the calibrated images of (b) and (c). The standard deviation σ,

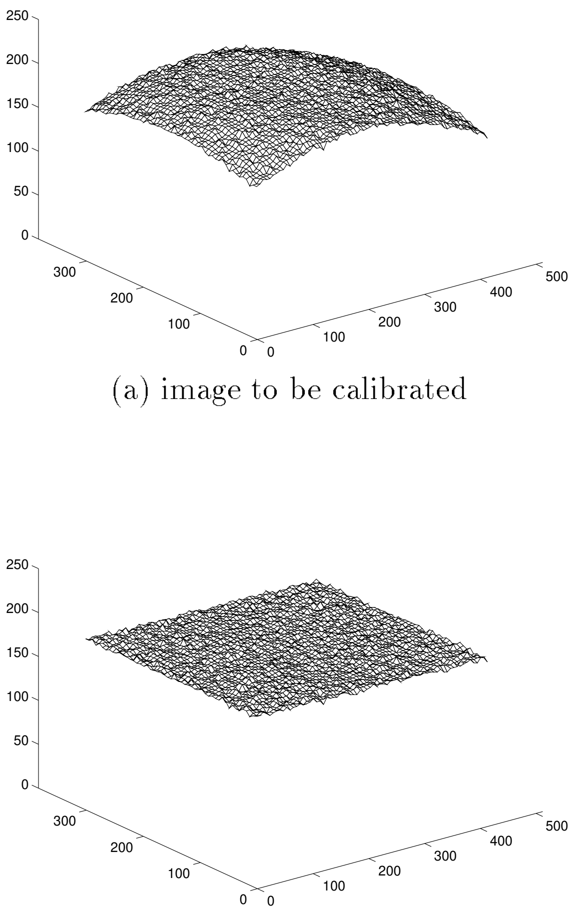
$$\sigma = \left[\frac{\sum (E(x,y) - M)^2}{N - 1} \right]^{\frac{1}{2}}$$

is used to compute the brightness variations of images before and after the calibration. In the above formula, M is the mean of the brightness values and N is the number of pixels in an image. For the yellow board, $\sigma = 14.93$ before the calibration and $\sigma = 2.05$ after the calibration. For the gray board, $\sigma = 10.58$ before the calibration and $\sigma = 1.685$ after the calibration. To give a better visualization of the calibration results, Figure 6.41 gives the brightness plots of the images of the yellow and gray boards before and after the calibration. The brightness values of the plot are extracted from every sixth row and column in an image. Ideally, the calibrated images should have constant brightness value. Because of image noise, slight non-uniformity of the calibration board, and other causes, there are still some small variations on the brightness values. Comparing the standard deviations before and after the calibration, we can see significant differences. The calibration results confirm the simplied camera model and show that the calibration process is necessary and effective.

6.4 Geometric Aberration

In the calibration process, we assume that each point in an image corresponds to a photo sensor of the CCD array in a camera. This is not true in our case. The CCD sensor in the 3CCD Sony camera is a 768x493 array. The video signal from the output of the camera is resampled by the frame grabber of the Max-Video200 image processing hardware. The output image is a 512x480 array with an incorrect aspect ratio. The resampling does not affect the calibration process since the calibrating image and the images calibrated are all resampled in the same way. However it does affect the results of surface recovery because the recovery process uses surface derivatives to compute depth. To get the right aspect ratio, images have to be scaled with different factors in the dimensions x and y. The scaling factors can be easily found from the image of a flat board with a grid of squares.

In the general case, there will be some geometric aberrations, such as barrel distortion and radial distortion, introduced by camera lens. The degree of the distortion depends on the focal length of the camera. Under our imaging conditions, the object is far away and the camera is set to telephoto mode so that the distortion is very small. A flat board with a grid of squares is used to detect the distortion. We found that the distortion is so small that the effects of the geometric aberrations can be ignored.



(a) image to be calibrated

(b) image to be calibrated

(a') calibrated image

(b') calibrated image

Figure 6.41: (a) and (b) are the brightness plots of images of a gray board and yellow board before calibration. (a') and (b') are the brightness plots of the calibrated images of (b) and (c).

6.5 Projection of the Rotation Axis

The images used for surface recovery are taken during the rotation of an object. To recover the object surface from its images, the projection of the rotation axis in the images must be known. Since the rotation axis is used as y axis in the coordinate reference system, the projection of the rotation axis should be a vertical line in images. To make the projection of the rotation axis vertical and to find out the location of the rotation axis in images, a thin and transparent flat card with tilted lines (see Fig. 6.42 (a)) is used. The surface normal of the card is perpendicular to the image plane. Two images of the card are taken before and after the card is rotated by π. The two images are superimposed together as one image (see Fig. 6.42(b)) The line which passes through the symmetric center of the image is the projection of the rotation axis(see dotted line in Fig. 6.42(b)). The location of the rotation axis in the image can be determined from the intersections of the two groups of tilted lines. By adjusting the orientation of the camera, the projection of the rotation axis can be made vertical. With the camera position and the rotation axis fixed, the projection of the rotation axis in the images taken during the object rotation will be also fixed. The projection of the rotation axis can be determined with subpixel accuracy. From analysis in the next chapter, we will see that this accuracy is sufficient for surface recovery.

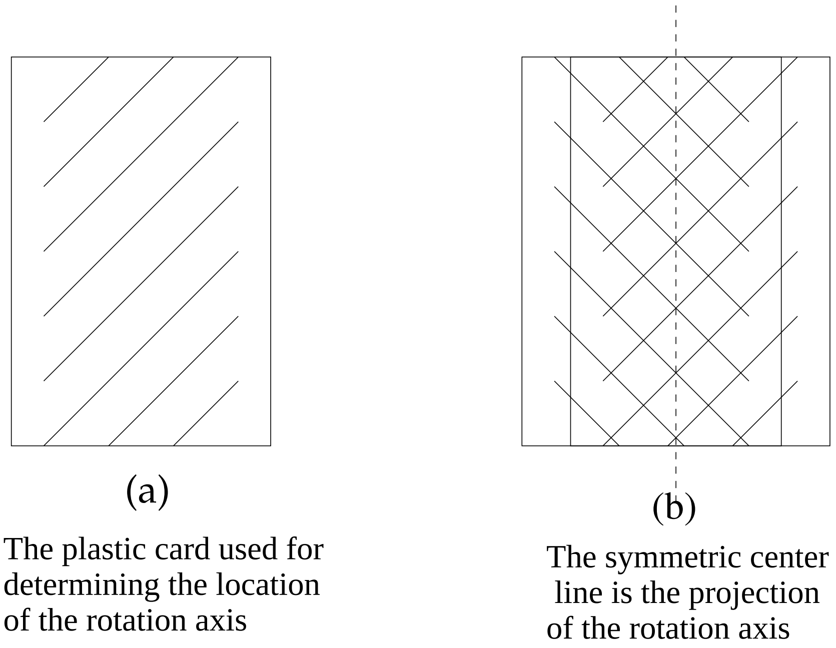


Figure 6.42: Determining the projection of the rotation axis. (a) is a flat transparent plastic card with tilted lines. The surface normal of the card is perpendicular to the image plane. (b) is the image obtained by overlaying two images taken before and after π rotation. The symmetric center of the tilted lines gives the rotation axis

Chapter 7

Analysis and Evaluation

In the previous chapters, we have described techniques for surface reflectance extraction and surface recovery and presented some experimental results. We have also discussed some practical problems with experimental conditions and the methods used to overcome the problems. In this chapter, we will give detailed analysis and evaluation of these techniques. An ellipsoid with ground truth is used in the analysis and evaluation. There are many factors which can affect the accuracy of the surface recovery. We will consider some major factors. Section 1 introduces the measurements used in the evaluation and analysis. Section 2 describes the experiments with an ellipsoid with ground truth. Section 3 discusses the surface reflectance function extracted from the images. Section 4 analyzes the surface recovery error caused by image noise, non-uniform surface reflectance, non-collinear light source, error accumulation, and other factors in surface recovery using the photogeometric technique. Section 5 evaluates and analyzes the wire-frame technique. The final section summarizes the evaluation and analysis of the two techniques.

7.1 Measurements in Analysis

Before we perform quantitative analysis, we would like to introduce some error measurements. From now on, we denote the computed or estimated value for variable d as $\tilde{d}$ and the real value for d as itself. For example, $\tilde{z}$ is the computed depth and z is the real depth. Also we denote the difference of the computed value $\tilde{d}$ and the real value d as δd. For instance, $\delta z = \tilde{z} - z$. In the analysis, 3D and 2D locations are measured in image pixels. One of the most widely used error measurements is the root mean square error (r.m.s.) given by:

$$\left(\frac{\sum_{x,y}\delta d(x,y)^2}{N}\right)^{1/2} = \left(\frac{\sum_{x,y}(\tilde{d}(x,y)-d(x,y))^2}{N}\right)^{1/2},$$

where $\tilde{d}(x,y)$ is the computed value and $d(x,y)$ is the real value at image point (x,y) and N is the number of data used.

The calculations for r.m.s. error in the depth, p and q components are straightforward. The angular error gives the angle between the computed surface normal and the real surface normal. The r.m.s. angular error is computed by

$$\left(\frac{\sum_{x,y} angle(x,y)^2}{N}\right)^{1/2}$$

with

$$angle(x,y) = \arccos\left(\frac{\tilde{p}(x,y)p(x,y)+\tilde{q}(x,y)q(x,y)+1}{\sqrt{\tilde{p}(x,y)^2+\tilde{q}(x,y)^2+1}\sqrt{p(x,y)^2+q(x,y)^2+1}}\right).$$

Besides r.m.s. error, the average error, $\frac{\sum_{x,y}\delta d(x,y)}{N}$ is also used to check if the error is biased or not. Since we compute the surface depth and orientation at every pixel in an image, we can measure the error between the computed data and the real data at each pixel. It is natural to think about representing the error in brightness at each pixel. Therefore we define a set of error maps in terms of images.

A depth error map is an image. The brightness $I(x,y)$ of a point in the error map is defined by

$$I(x,y) = \begin{cases} 128 + 127\delta z(x,y)/T_d & \text{if} \quad -T_d < \delta z(x,y) < T_d \\ 0 & \text{if} \quad \delta z(x,y) < -T_d \\ 255 & \text{if} \quad \delta z(x,y) > T_d \end{cases}$$

with T_d as a threshold. If (x,y) is on the background, $I(x,y) = 128$.

A p error map is an image. The brightness $I(x, y)$ is defined by

$$I(x,y) = \begin{cases} 255 - 254|\delta p(x,y)|/T_p & \text{if } |\delta p(x,y)| < T_p \\ 0 & \text{otherwise} \end{cases}$$

with T_p as a threshold. If point (x, y) is on the background, $I(x, y) = 255$.

A q error map is defined in the same way as the p error map except that $\delta p(x, y)$ and T_p are substituted by $\delta q(x, y)$ and T_q, respectively. In both p and q error maps, the darker the brightness in the map, the larger the error in the computed data.

An angular error map is an image. The brightness $I(x, y)$ is defined by

$$I(x,y) = \begin{cases} 255 - 254 angle(x,y)/T_a & \text{if } angle(x,y) < T_a \\ 0 & \text{otherwise} \end{cases}$$

with T_a as a threshold. If point (x, y) is on the background, $I(x, y) = 255$. Thus, the darker the brightness in the map, the larger the angular error in the computed surface normal.

The values for the thresholds can be adjusted so that the brightness in a map can be changed. The values are empirically determined as $T_d = 4$, $T_p = 2$, $T_q = 2$ and $T_a = 20$. By comparing the brightness values between pixels in a map or between two maps, the variations of error can be visualized. We also use profiles to view the depth error. The profiles of the computed depth and real depth are extracted from some specified rows or columns and overlaid so that direct visual comparison can be made.

7.2 Analysis with an Ellipsoid

7.2.1 Surface recovery of an ellipsoid

An ellipsoid object, which is designed by Woodham and used by Ying Li in here Ph.D work [45], is used in the analysis for the photogeometric and wire-frame techniques. The geometric model of the ellipsoid can be expressed as

$$\frac{x^2}{(5S)^2} + \frac{y^2}{(9S)^2} + \frac{z^2}{(3S)^2} = 1 \tag{7.30}$$

where S is a scale factor. A sequence of fifty four images of the ellipsoid is taken during 2π

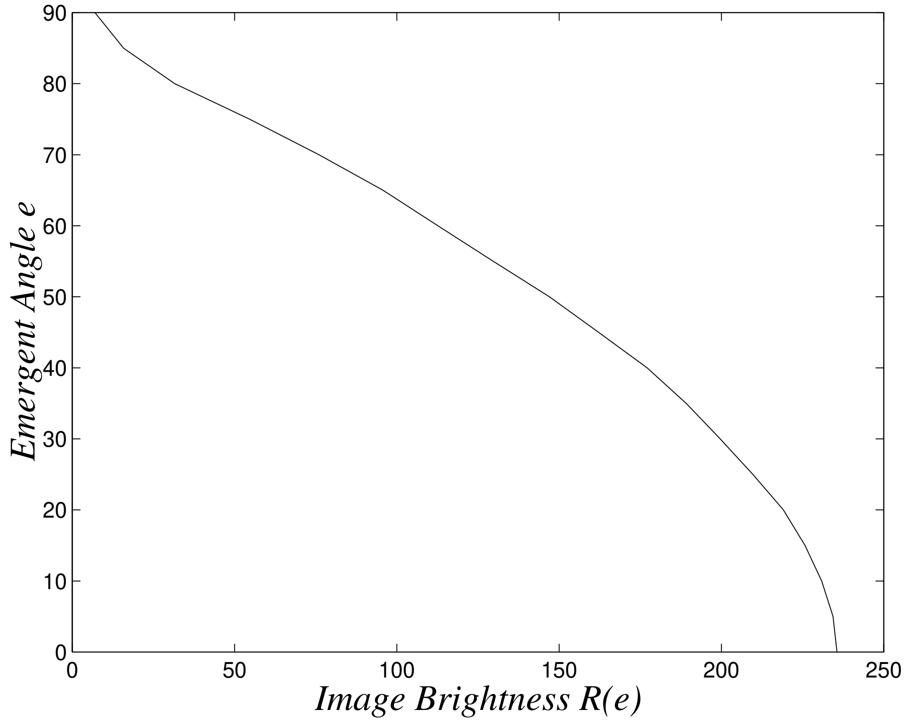
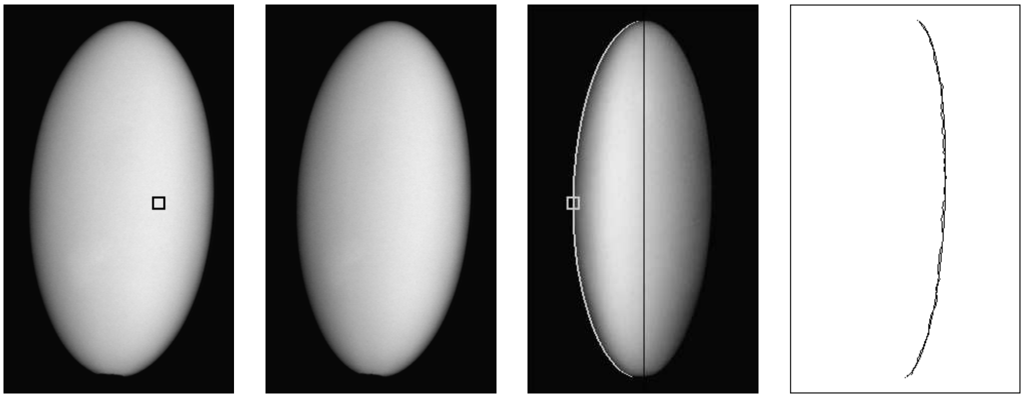


Figure 7.43: Surface reflectance of the ellipsoid extracted by averaging nine sample points

rotation of the object. The rotation angle between two successive images is $\pi/36$ during zero to π rotation and $\pi/18$ during π to 2π rotation. Figure 5.31 shows some images from the sequence. For extracting the surface reflectance function, the images taken during zero to $\pi/2$ rotation are sufficient. To reduce the error in the surface reflectance function, more sample points are used and thus more images are needed. The surface reflectance function of the ellipsoid is extracted by tracking 9 sample points. The sample points have surface orientation $(-\tan(k\pi/18), 0, 1)$, $0 \leq k < 9$. The 3D locations of the sample points are computed from the singular points in $image_{k\pi/18}$ and the corresponding occluding contour points in $image_{k\pi/18+\pi/2}$, $0 \leq k < 9$. The surface reflectance function (see Fig. 7.43) is obtained by averaging the nine reflectance functions extracted from the 9 sample points. To recover the surface of the ellipsoid, $image_0$ and $image_{\pi/18}$ (see



a. $image_0$ b. $image_{\pi/18}$ c. $image_{\pi/2}$ d. $p = 0$ curves

Figure 7.44: (a) and (b) are images used for surface recovery. The white curve in (c) is the occluding contour. Square boxes denote the singular point and its corresponding contour point. The curves in (d) are the real $p = 0$ curve and the computed $p = 0$ curve.

Fig. 7.44 (a) and (b)) are used. The depth and surface orientation on the $p = 0$ curve are recovered first by using the occluding contour in $image_{\pi/2}$ (see white curve in Fig. 7.44 (c)). In Fig. 7.44 (d), the smooth curve is the real $p = 0$ curve, and the other curve is the computed $p = 0$ curve. The depth and surface orientation on the whole surface of the ellipsoid are recovered by extending the computation from the $p = 0$ curve in the x direction. The depth plot of the recovered surface is shown in Fig. 7.45. The surface orientation plot of the recovered surface is shown in Fig. 7.46. The sign for the q component at an image point is determined by the ground truth of the ellipsoid.

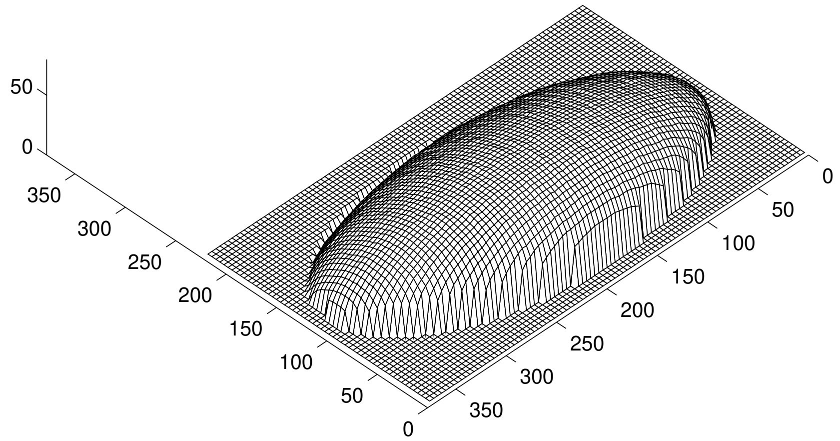


Figure 7.45: The recovered surface depth of the ellipsoid

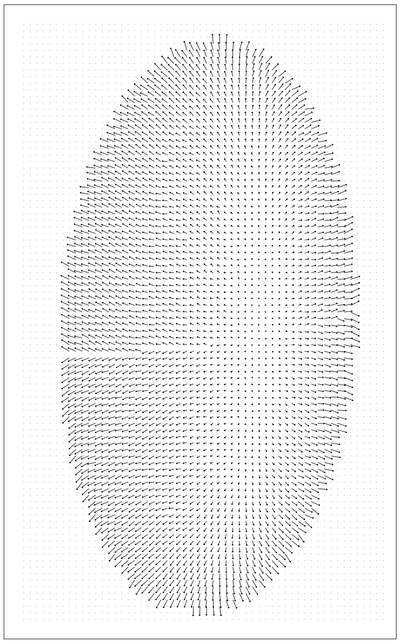


Figure 7.46: The recovered surface orientation of the ellipsoid

7.2.2 Pose determination

Since the handle for supporting the manufactured ellipsoid is not aligned with any axes of the ellipsoid, the pose of the ellipsoid in the images is not the same as that described by Formula 7.30. To analyze error in the recovered surface of the ellipsoid, the pose of the ellipsoid must be found. Six parameters, three for rotation and three for translation, need to be computed. The three parameters for rotation are defined as angles γ, φ, and β, which are clockwise around the z, x and y axes respectively. The three translation parameters are δx, δy and δz. The transformation from a point (x, y, z) on the ellipsoid defined by Formula 7.30 to a point (x_0, y_0, z_0) on the ellipsoid of the initial pose in $image_0$ is

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} = R_\beta R_\varphi R_\gamma \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} \delta x \\ \delta y \\ \delta z \end{bmatrix}$$

where R_β, R_φ, R_γ are the rotational transformations about y, x, z axes respectively.

The images of the ellipsoid are taken during rotation of the ellipsoid around the y axis. After the object has been rotated by α, a point (x_0, y_0, z_0) on the ellipsoid will be transformed to a point $(x_\alpha, y_\alpha, z_\alpha)$. Let the rotation transformation be R_α; we have

$$\begin{bmatrix} x_\alpha \\ x_\alpha \\ x_\alpha \end{bmatrix} = R_\alpha \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} = R_\alpha \left(R_\beta R_\varphi R_\gamma \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} \delta x \\ \delta y \\ \delta z \end{bmatrix} \right) \tag{7.31}$$

To determine δx, δy and δz, the image centroids of the ellipsoid in $image_{k\pi/18}$, $(0 \leq k < 36)$, are computed. Then δy is determined by averaging the y coordinates of the centroids; δx and δz are calculated by minimizing $\sum(\delta x \sin\alpha + \delta z \cos\alpha - Xc_\alpha)^2$ with Xc_α as the x coordinates of the centroids. The scale factor S, rotation angle β, φ and γ are determined by optimization. The depths on the $p = 0$ curves in $image_{2k\pi/18}$, $(0 \leq$

$k < 18$), of the ellipsoid are calculated from the occluding contours in $image_{2k\pi/18+\pi/2}$. The depths on the corresponding $p = 0$ curves of the ellipsoid defined by Formula 7.31 are computed. The values of S, β, φ and γ which minimize the least-squares difference of the two sets of depths are found. Finally, we have all the parameters determined with ± 0.01 accuracy.

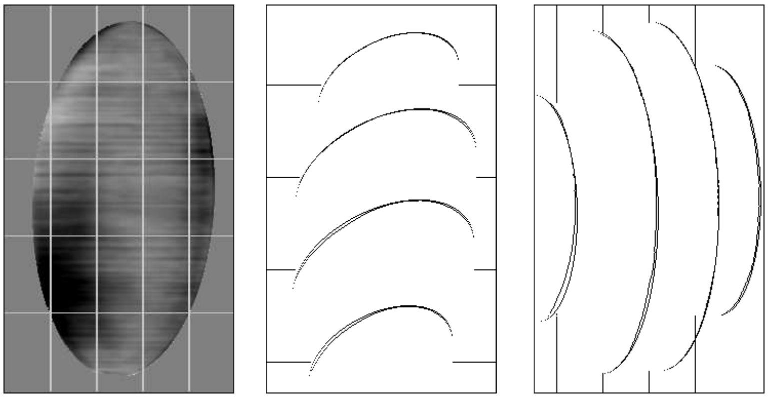
S	β	φ	γ	δx	δy	δz
19.42	$-26.39°$	$1.78°$	$2.11°$	1.77	-0.24	1.06

Table 7.1: The scale factor and the initial pose of the ellipsoid.

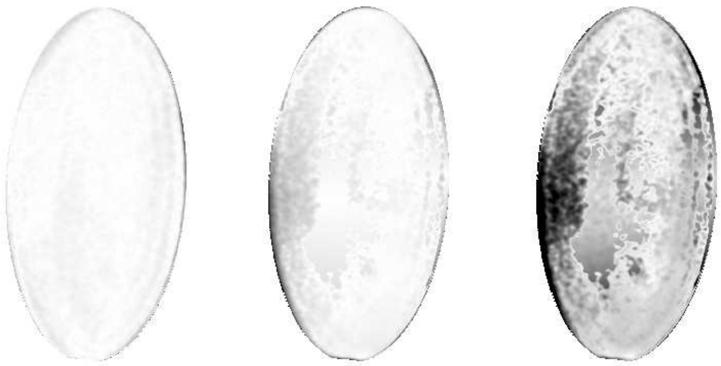
7.2.3 Surface recovery error

After the pose of the ellipsoid has been determined, the ground truth on the depth and surface orientation at each pixel in $image_0$ can be found. The number of image pixels corresponding to the recoverable area in $image_0$ is 48982. The depth and surface orientation are actually recovered at 47781 image points, about 98% of the recoverable area in $image_0$. The recovered depth and surface orientation are compared with the ground truth. The depth error map in Fig. 7.47 (a) shows the depth error of the recovered surface. When the brightness at a pixel in the map is darker than the pixels corresponding to the background, the recovered depth at that pixel is smaller than the true depth. Otherwise, the recovered depth at that pixel is larger than the true depth. To give a better illustration of the depth error, the depth profiles of both recovered surface and true surface are extracted from several rows and columns, which are labeled as the white lines in $image_0$, and shown in Fig. 7.47 (b) and (c).

The maps of p error, q error and angular error are shown in Fig. 7.47(d), (e) and (f). In these error maps, the darker the brightness, the larger the error. From the fact



a. depth error b. row profiles c. column profiles



d. p error e. q error f. angular error

Figure 7.47: Surface recovery error of the ellipsoid. (a) shows the depth error map. When $I(x,y) = 128$, $\delta z(x,y) = 0$; when $I(x,y) = 0$, $\delta z(x,y) < -4$; when $I(x,y) = 255$, $\delta z(x,y) > 4$. (b) and (c) are the depth profiles of the true depth and computed depth from some rows and columns labeled in (a) with white lines. (d) and (e) are the error maps of p, q. When $I(x,y) = 0$, $|\delta p(x,y)| > 2$ or $|\delta q(x,y)| > 2$; when $I(x,y) = 255$, $|\delta p(x,y)| = 0$ or $|\delta q(x,y)| = 0$. (f) is the angular error map. When $I(x,y) = 0$, $angle(x,y) > 20$; when $I(x,y) = 255$, $angle(x,y) = 0$.

that the points near the boundary of the object are darker than the points inside the object, we know the error is larger at the points near the boundary. Comparing the brightness in the p error map to that in the q error map, we can see that the error in the q component is larger than the error in the p component at most pixels. Table 7.2 gives the r.m.s. errors in depth and surface orientation of the recovered surface. Considering the maximum depth of the ellipsoid is 69, we can have a rough idea of how much the r.m.s. depth error is relative to the size of the ellipsoid. The table verifies that the error in the q component is larger than the error in the p component. The error listed in

	depth	p component	q component	Angular
r.m.s. error	1.622	0.130	0.251	7.336°

Table 7.2: The surface recovery error of the ellipsoid

this table may contain the error caused by the error in the pose of the ellipsoid. The maximum depth and orientation errors caused by the error in the pose can be calculated since the maximum error in the parameters for the pose are within 0.01. By applying 0.01 perturbation on all the parameters and comparing the depth and orientation of the ellipsoid before and after the perturbation, the maximum depth and orientation errors caused by the error in the pose are calculated and shown in Table 7.3. The errors in the table, especially the angular error, are very small compared to the surface recovery errors of the ellipsoid.

	depth	p component	q component	Angular
r.m.s. error	0.076	0.033	0.075	0.09°

Table 7.3: Bounds on the depth and orientation error contributed by the error in pose determination of the ellipsoid.

7.3 Surface Reflectance Function

7.3.1 Accuracy of surface reflectance function

The surface reflectance function used in the photogeometric technique is extracted by tracking sample points during object rotation. The calculation of the 3D locations of sample points has been described in Section 3.3.2. Because of image noise, quantization, and the local surface shape, the calculated 3D location of a sample point may contain error. The error in the location will cause an error in the surface reflectance function extracted from these points.

One way deal with the problem is to use multiple sample points on the surface. For a generic smooth object, many surface points will give rise to singular points in images and will also be contour points during rotation. All of these surface points can be used as sample points for extracting the surface reflectance function. The surface reflectance function obtained by averaging the brightness values of the sample points reduces the error caused by image noise, quantization, local surface shape and other random effects. Figure 4.17 shows the surface reflectance functions, R_1 (dotted line), R_2 (dash-dot line), and R_3 (dashed line) extracted, respectively, from the three sample points of the vase (see Fig. 4.16). The differences between the three reflectance functions are evident. The solid line in Fig. 4.17 is the surface reflectance function obtained by averaging the three reflectance functions.

	function R_1	function R_2	function R_3
r.m.s. difference	4.426	2.129	4.055
average difference	-3.83	0.248	3.58

Table 7.4: The difference between the averaged surface reflectance function and the surface reflectance function extracted from each sample point on the vase.

Table 7.4 shows the r.m.s. and average differences between each surface reflectance function and the averaged surface reflectance function of the vase. The r.m.s. difference is computed by $\left(\frac{\sum[R_i(e)-R(e)]^2}{k}\right)^{1/2}$ and the average difference is computed by $\frac{\sum[R_i(e)-R(e)]}{k}$ with $k = 19$ as the number of the image brightness values used for building each surface reflectance function. $R_i(e)$ and $R(e)$ are, respectively, the reflectance functions extracted from each sample point and the averaged reflectance function.

	R_0	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8
r.m.s. difference	1.57	1.35	3.92	1.92	1.41	5.20	3.64	4.89	5.69
average difference	0.19	−0.36	3.12	1.48	1.19	−3.57	−2.84	−3.34	4.12

Table 7.5: The r.m.s. difference and average difference between the averaged reflectance function and the reflectance function extracted from each sample point on the ellipsoid.

The surface reflectance function of the ellipsoid (see Fig. 7.43) is built by averaging the brightness values of nine sample points. Table 7.5 shows the r.m.s. and average differences between each surface reflectance function $R_i(e)$ $(0 \leq i < 9)$ and the averaged surface reflectance function. From the two tables, we can see that the differences are large among the reflectance functions extracted from sample points. The large differences are caused by either the inaccurate calculation of the 3D locations of the sample points or the different reflectance factors of the sample points. The second case implies that the surface reflectance of the ellipsoid is not uniform.

7.3.2 Pseudo surface reflectance function

The accuracy of the surface reflectance function will affect the accuracy of the recovered surface. However, if only surface depth and the p component of surface orientation are concerned, a pseudo surface reflectance function is sufficient.

Let reflectance function $E = Q(\cos e)$ be the real reflectance function of a surface.

Let $E = Q_s(\cos e)$ be another reflectance function which satisfies the following condition:

$$\frac{Q_s^{-1}(E_0)}{Q_s^{-1}(E_\alpha)} = \frac{Q^{-1}(E_0)}{Q^{-1}(E_\alpha)} \tag{7.32}$$

for any image brightness values of E_0 and E_α. The above condition is equivalent to the condition

$$Q_s^{-1}(E) = KQ^{-1}(E)$$

for any brightness value E. Here K is a constant. The reflectance function Q_s is called *a pseudo reflectance function of* Q. In surface recovery, we developed the formula

$$p = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{Q^{-1}(E_\alpha)}{Q^{-1}(E_0)}$$

to compute the p component (see Equation 3.21). Since $\frac{Q_s^{-1}(E_\alpha)}{Q_s^{-1}(E_0)} = \frac{Q^{-1}(E_\alpha)}{Q^{-1}(E_0)}$, calculating p using Q_s^{-1} gives the same results as calculating p using Q^{-1}. If surface depth is integrated only by using the p component, the depth recovered by using the two reflectance functions will also be the same.

A pseudo reflectance function can be extracted from $p = 0$ surface points. Let d be a point of surface orientation $(-p_0(d), -q_0(d), 1)$ with $p_0(d) = 0$. Assume the 3D location of d is known so that we can calculate the projection and extract the image brightness values of d from the image sequence taken during object rotation. Let $e_0(d)$ be the emergent angle of d before the rotation of the object; then $\cos e_0(d) = \frac{1}{\sqrt{q_0^2(d)+1}}$. After rotating the object by α, from Equation 3.19, we have

$$\cos e_\alpha(d) = \cos e_0(d)\cos\alpha \quad \& \quad \cos\alpha = \frac{\cos e_\alpha(d)}{\cos e_0(d)}.$$

If $q_0(d)$ is known, the emergent angle of d during object rotation can be calculated and the surface reflectance function can be built from the correspondence between the emergent angle and the brightness value of d. If $q_0(d)$ is unknown, another surface reflectance function can be built by using the rotation angle α of the object as the estimated value for the emergent angle of d. Then the reflectance function built is

$$E_\alpha = f(\cos\alpha) = f(\frac{\cos e_\alpha(d)}{\cos e_0(d)})$$

and the inverse function is

$$f^{-1}(E_\alpha) = \frac{\cos e_\alpha}{\cos e_0(d)} = \frac{1}{\cos e_0(d)} Q^{-1}(E_\alpha).$$

From the definition, function f is a pseudo reflectance function of Q.

In extracting the surface reflectance function, the sample points of initial surface orientation $(0,0,1)$ are used. In the presence of error, the q component of the surface orientation of the sample points may not be zero. Thus the surface reflectance function extracted may not be the real surface reflectance function but a pseudo reflectance function. When no singular image points are available from images, the pseudo reflectance functions can be extracted and used for surface recovery.

The pseudo surface reflectance function can be used to compute the p component, and the surface depth can be computed by integrating p along the x direction. After the depth has been obtained, the q component at each point can be computed from the derivatives of the depth along the y direction. After surface orientation has been found at some image points, the real surface reflectance function can be estimated.

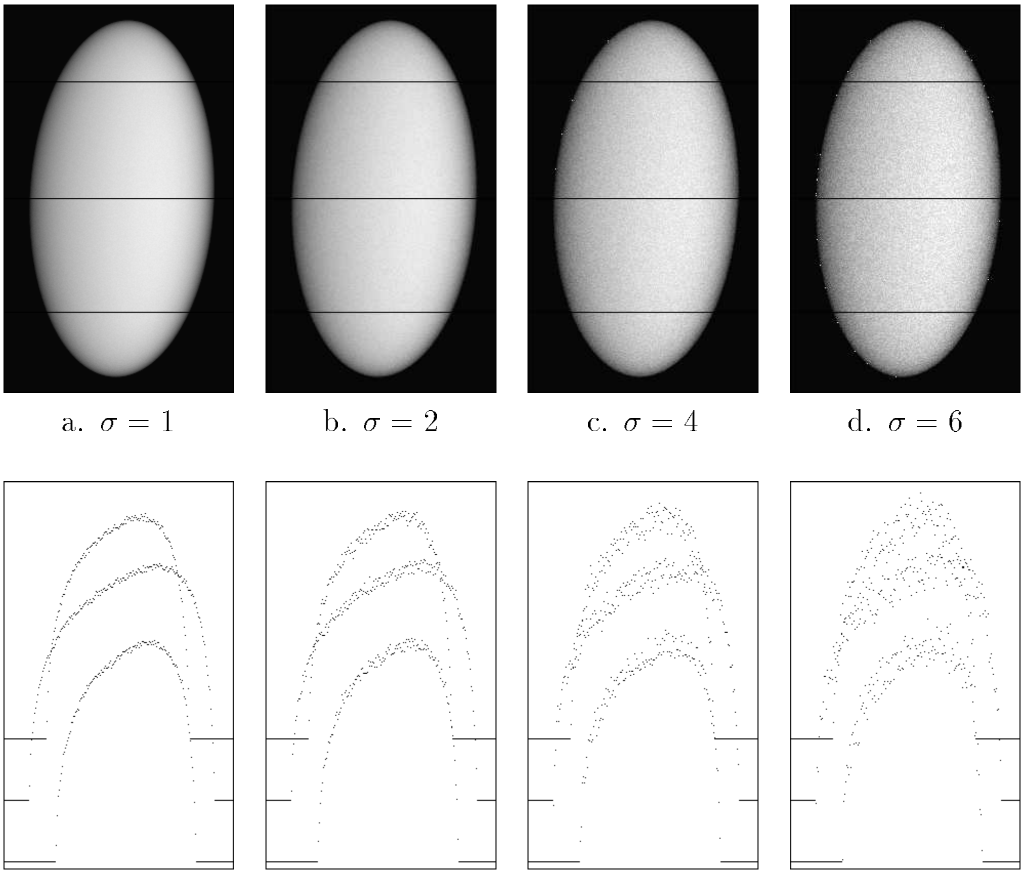
7.4 Analysis of the Photogeometric Technique

7.4.1 Effects of image noise

There are several error sources in surface recovery. One of them is image noise, which may come from the CCD sensor, output amplifier, signal transfer cable, and somewhere else. Image noise is usually considered as additive Gaussian noise with zero mean. To analyze the effects of image noise, a set of synthetic images of the ellipsoid with image noise at different levels is generated and the surface recovery technique is applied to the synthetic images. Then the recovered surfaces are compared with the ground truth. To make the analysis applicable to the surface recovered from the real images of the ellipsoid, the synthetic images are generated under the same illuminant and viewing conditions as those under which the real images are taken. The pose of the ellipsoid in each synthetic image is the same as that in the corresponding real image. The synthetic images are rendered with the surface reflectance function extracted from the real images. The synthetic images are also labeled as $image_\alpha$ with α as the rotation angle of the ellipsoid. The idea is to generate synthetic images which are the same as the corresponding real images except that the image noise level in the synthetic images can be controlled. A random process

$$N(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x}{\sigma})^2}$$

is used to add Gaussian noise to synthetic images. The parameter σ determines the noise level. The image noise levels we used are 1, 2, 4 and 6. At each level, two synthetic images, $image_0$ and $image_{\pi/18}$, are generated. Figure. 7.48(a)-(d) shows $image_0$ of the synthetic images at different noise levels and the diagram below each image shows the brightness profiles extracted from the three rows labeled with black lines in the image. At each of the four noise levels, the surface is recovered by using the two synthetic images, $image_0$ and $image_{\pi/18}$. Since we do not consider the noise effects on the surface reflectance function extraction, we do not extract the surface reflectance function from the synthetic images. The surface reflectance function used in surface recovery from the synthetic images is the same as that used to render the synthetic images. To filter image noise, Gaussian filters of different sizes are used(see Table 7.6). The best surface recovery results are obtained with the Gaussian filter of the given sizes. In order to compare the surface recovered from the real images with those recovered from the synthetic images,



a. $\sigma = 1$ b. $\sigma = 2$ c. $\sigma = 4$ d. $\sigma = 6$

e. f. g. h.

Figure 7.48: (a)-(d) are the synthetic images of the $image_0$ of the ellipsoid at different image noise levels. (e)-(h) are the brightness profiles extracted from the corresponding images in the same column.

the $p = 0$ curve recovered from the real images is used for the surface recovery from the synthetic images. The surface recovery starts from the $p = 0$ curve and extends in the x direction.

noise level	$\sigma = 1$	$\sigma = 2$	$\sigma = 4$	$\sigma = 6$	real images
filter size(in pixel)	9	13	19	21	11

Table 7.6: The size of the Gaussian filter used for the images at different noise levels

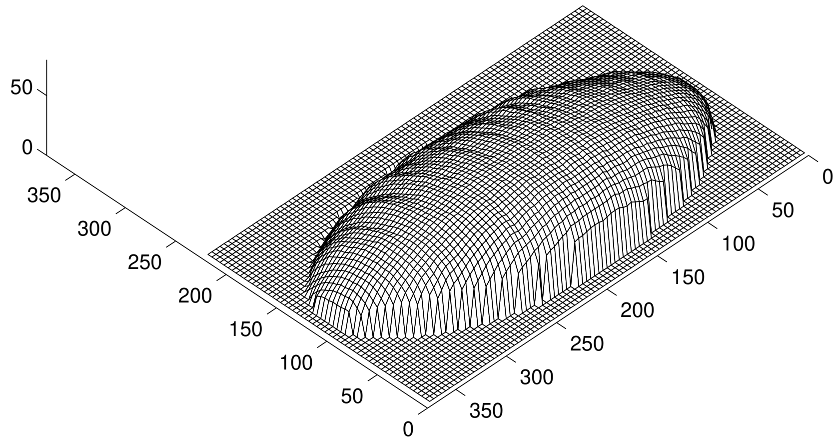


Figure 7.49: The surface depth recovered from synthetic images of noise level 6

The depth and surface orientation of the surface recovered from the synthetic images of noise level 6 are shown in Fig. 7.49 and Fig. 7.50. The sign of the q component is determined from Formula 7.31 which is used to generate the synthetic images. The depth errors in the recovered surfaces at different noise levels are shown in Fig. 7.51. The diagrams in the same column correspond to the same image noise level. The first row in Fig. 7.51 is the depth error maps. The second and third rows are the depth profiles. The rows and columns from which the profiles are extracted are shown in Fig. 7.51(a)

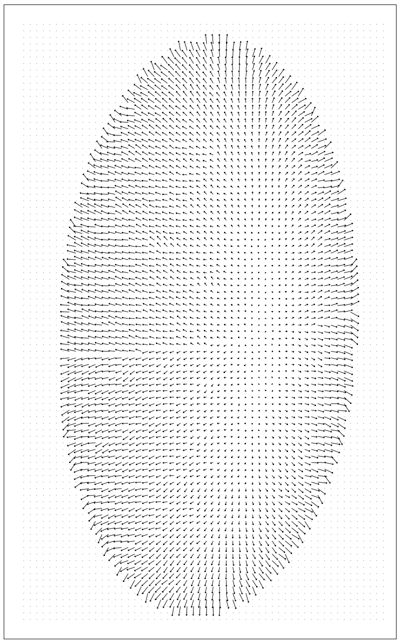


Figure 7.50: The surface orientation recovered from synthetic images of noise level 6

	$\sigma = 1$	$\sigma = 2$	$\sigma = 4$	$\sigma = 6$	real image
depth error	0.311	0.417	0.621	0.824	1.616
p error	0.070	0.080	0.102	0.122	0.103
q error	0.133	0.149	0.189	0.220	0.246
angular error	3.685°	4.602°	5.676°	6.568°	7.279°

Table 7.7: The r.m.s. error of surface recovery from synthetic images at different noise levels

as white lines. For comparison, the true depth profiles are overlaid with the computed depth profiles in each diagram. The p, q, and angular error maps corresponding to each noise level are shown in Fig. 7.52. The diagrams in the same column are the error maps of the surfaces recovered from the same synthetic images. Comparing the p error map with the q error map at the same noise level, the q error map is darker than the p error map. This means that, in surface recovery, the error in q component is larger than the error in p component. The r.m.s. errors of depth, p, q, and angular listed in Table 7.7 verify this observation. To make the r.m.s. errors in the surfaces recovered

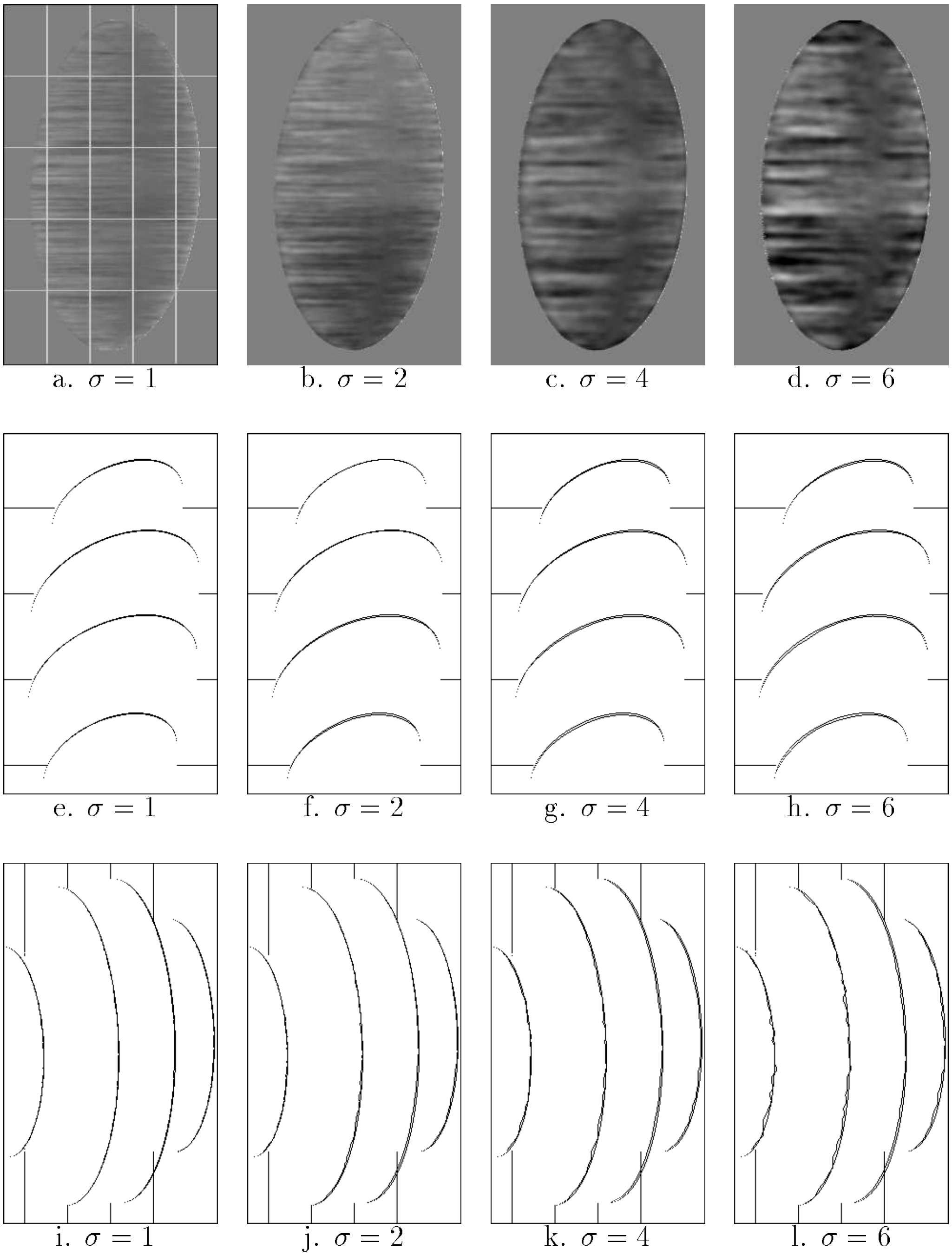


Figure 7.51: (a)-(d) are the depth error maps of the surfaces recovered from the synthetic images of the different noise levels. (e)-(h) are the row depth profiles extracted from the recovered surfaces. (i)-(h) are the column depth profiles extracted from the recovered surfaces. The rows and columns are labeled in (a) with white lines

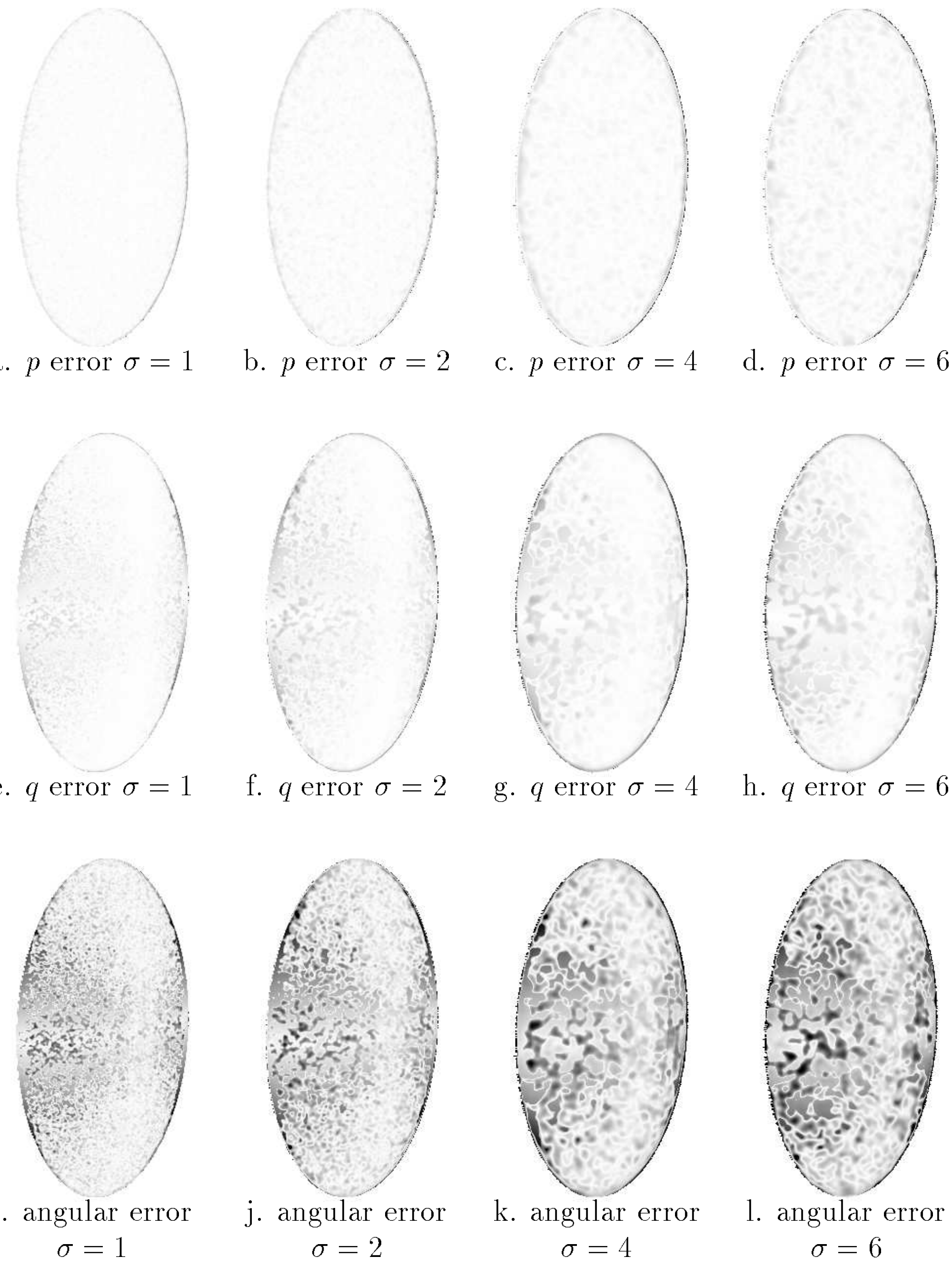


Figure 7.52: (a)-(e) are p error maps of the surfaces recovered from synthetic images of the different noise levels. (f)-(h) are q error maps of the recovered surfaces. (i)-(l) are angular error maps of the recovered surfaces.

from the synthetic images comparable with those recovered from the real images, the r.m.s. errors are computed from the image points in the area on which the depth and surface orientation are recovered from both synthetic images and real images. The area contains 47615 image points, which are about 97% of the image points in the recoverable area. The r.m.s. errors in the surface recovered from the real images are recomputed by using the data in the joined area. From the table and the error maps, we can see that the surface recovery error increase as the noise level increases. In Fig. 7.53, (a) is $image_0$ of the real images, (c) is $image_0$ of the synthetic images with noise level 1. (b) and (d) are brightness profiles extracted from the two images from the rows labeled with the black lines. Visually comparing the two profiles, we see that the noise level in the real images is about 1. Therefore the errors caused by image noise in the surface recovered from the real images should be close to that in the surface recovered from the synthetic images at noise level 1. Because of the other factors, such as non-uniform reflectance, inaccurate surface reflectance functions, non-collinear illumination etc., the error in the surface recovered from the real images of noise level 1 is much larger.

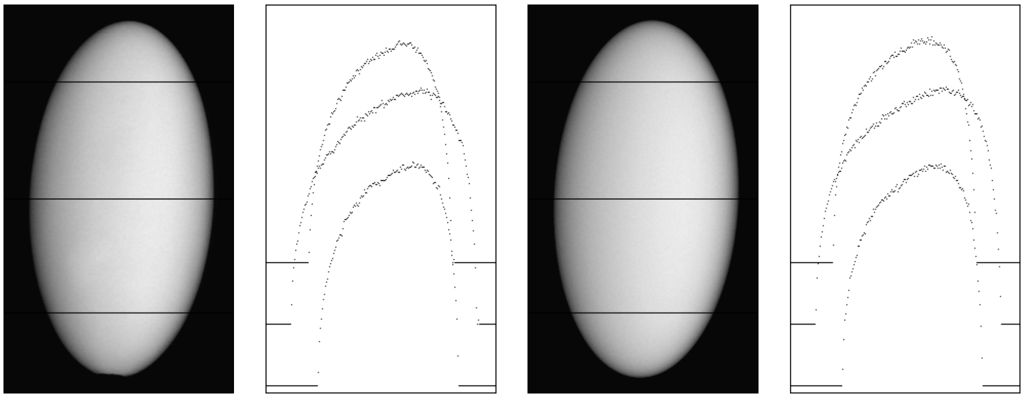
The following is a mathematical analysis of the effects of image noise on surface recovery. In surface recovery, the formula

$$z = z_0 + \int_{x_0}^{x} p dx$$

is used to compute the depth at each pixel in $image_0$ except at the $p = 0$ curve. In the formula, x_0 and z_0 are the coordinates of a starting point and p is computed by

$$p = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{\cos e_\alpha}{\cos e_0} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{Q^{-1}(E_\alpha)}{Q^{-1}(E_0)}.$$

Let the object surface be a uniform Lambertian surface. The surface reflectance function can be expressed as $E = Q(\cos e) = C\cos e$ with C as a constant. The inverse reflectance function is $\cos(e) = Q^{-1}(E) = E/C$. In the presence of random image noise n, $\tilde{E} = E + n$.



a. $image_0$ of the real images | b. brightness profiles from a. | c. $image_0$ of the synthetic images | d. brightness profiles from c.

Figure 7.53: (a) is the real image used for surface recovery. (c) is the synthetic image, with Gaussian noise of $\sigma = 1$, used for surface recovery. (b) and (c) show the brightness profiles of the real images and the synthetic images, respectively.

Applying the inverse function, we have $\cos \tilde{e} = Q^{-1}(\tilde{E}) = (E + n)/C$, where $\cos \tilde{e}$ is the estimated value for $\cos e$. Let n_0 and n_α be two independent Gaussian noise processes in $image_0$ and $image_\alpha$ of the object, respectively. Then

$$\tilde{E}_0 = C\cos(e_0) + n_0 = E_0 + n_0 \quad \& \quad \cos \tilde{e}_0 = \frac{E_0 + n_0}{C}$$

and

$$\tilde{E}_\alpha = C\cos(e_\alpha) + n_\alpha = E_\alpha + n_\alpha \quad \& \quad \cos \tilde{e}_\alpha = \frac{E_\alpha + n_\alpha}{C}.$$

The computed p component is

$$\tilde{p} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{\cos\tilde{e}_\alpha}{\cos\tilde{e}_0} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{E_\alpha + n_\alpha}{E_0 + n_0} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{E_\alpha}{E_0}\left(\frac{1 + \frac{n_\alpha}{E_\alpha}}{1 + \frac{n_0}{E_0}}\right).$$

Using the binomial series expansion and assuming that $E_0 \gg n_0$, we have

$$\frac{1}{1 + \frac{n_0}{E_0}} \approx 1 - \frac{n_0}{E_0}.$$

The depth $\tilde{z}$ computed by using $\tilde{p}$ is

$$\tilde{z} = z_0 + \int_{x_0}^{x} \tilde{p} dx = z_0 + \frac{x - x_0}{\tan\alpha} - \frac{1}{\sin\alpha}\left(\int_{x_0}^{x} \frac{E_\alpha}{E_0} dx + \int_{x_0}^{x} (\frac{n_\alpha}{E_0} - \frac{E_\alpha n_0}{E_0^2} - \frac{n_0 n_\alpha}{E_0^2}) dx\right).$$

As $E_0 = C\cos e_0$ and $E_\alpha = C\cos e_\alpha$,

$$\tilde{z} = z_0 + \frac{x - x_0}{\tan\alpha} - \frac{1}{\sin\alpha}\int_{x_0}^{x} \frac{\cos e_\alpha}{\cos e_0} dx + \frac{1}{\sin\alpha}\int_{x_0}^{x} (\frac{n_\alpha}{E_0} - \frac{E_\alpha n_0}{E_0^2} - \frac{n_0 n_\alpha}{E_0^2}) dx.$$

The second integral represents the effects of image noise. Since n_0 and n_α are independent, the mean and deviation of the integration of $\frac{n_0 n_\alpha}{E_0^2}$ will become zero.

From the above equation, we conclude that

1. The image noises, n_0 and n_α, are suppressed by image brightness values.
2. The error caused by image noise is accumulated over the integration path. The mean of the accumulated error is zero and the deviation increases as the integration is computed along the path.

A surface which gives only specular reflection is called a *specular surface*. A surface which gives both diffuse and specular reflection is called a *hybrid surface*. The above conclusions can also be made on specular and hybrid surfaces(see Appendix A).

In signal processing, noise signals are usually characterized as signals of high frequency and low magnitude. The Fourier transform of integration is

$$F(\int f(x) dx) = \frac{1}{-i\omega} F(f(x)).$$

where ω is the signal frequency. The term $\frac{1}{\omega}$ suppresses high frequency signals. The Fourier transform of integration gives another explanation on why the photogeometric technique is resistant to image noise.

7.4.2 Effects of non-collinear light source

In surface recovery, the images are assumed to be taken under a collinear light source. But in practice, exactly aligning the illuminant direction and the viewing direction along a straight line is impossible. We put the light source at the top of the camera and adjust the light source so that the optical center of the light source is pointed at the object. In this case, there is a very small angle between the illuminant and viewing direction. Under the illumination of a non-collinear light source, the brightness value at a surface point is no longer a function only of the emergent angle, that is

$$E \neq Q(\cos e).$$

However, if the angle between the illuminant direction and viewing direction is sufficiently small, we still can treat a non-collinear light source as a collinear light source and extracting the surface reflectance function in the same way as before. Then the surface reflectance function extracted will be an approximation of the surface reflectance function extracted under a collinear light source.

For a uniform Lambertian surface, its surface reflectance function has the form $E = C\cos i$. The incident angle i $(0 \leq i \leq \pi/2)$ is the angle between the surface normal and the illuminant direction. The constant C is determined by surface albedo and radiance of the light source. Let the illuminant direction of a non-collinear light source be $(0, -q_l, 1)$, and q_l sufficiently small. If we consider the light source as a collinear light source and extract the surface reflectance function with the same method described in Section 4.2, the surface reflectance function to be extracted can be predicted by calculation.

Let surface point S be a sample surface point whose image in $image_0$ is a singular point under the non-collinear light source. Then the surface orientation of S is $(0, -q_l, 1)$. After rotation by α,

$$p_\alpha(S) = \tan\alpha, \quad \& \quad q_\alpha(S) = \frac{q_l}{\cos\alpha},$$

$$\cos i_\alpha(S) = \frac{(-p_\alpha(S), -q_\alpha(S), 1) \cdot (0, -q_l, 1)}{\sqrt{p_\alpha^2(S) + q_\alpha^2(S) + 1}\sqrt{q_l^2 + 1}} = \frac{q_l^2 + \cos\alpha}{q_l^2 + 1},$$

and

$$E_\alpha(S) = C\cos i_\alpha(S) = C\frac{q_l^2 + \cos\alpha}{q_l^2 + 1}.$$

If we approximate the non-collinear light source as a collinear light source and take the rotation angle α of the object as the emergent angle e of the sample point as we did in Section 4.2, the surface reflectance function we extracted and its inverse are

$$E = Q(\cos\tilde{e}) = C\frac{q_l^2 + \cos\tilde{e}}{q_l^2 + 1} \quad \& \quad \cos\tilde{e} = Q^{-1}(E) = \frac{E}{C}(q_l^2 + 1) - q_l^2,$$

where $\cos\tilde{e}$ is the approximated value for $\cos e$.

Now let's look at surface recovery error caused by using this surface reflectance function. Let a 3D surface point have surface orientation $(-p_0, -q_0, 1)$. The image brightness value of this surface point in $image_0$ is

$$E_0 = C\cos i_0 = C\frac{(q_0 q_l + 1)}{\sqrt{q_l^2 + 1}\sqrt{p_0^2 + q_0^2 + 1}}. \tag{7.33}$$

After rotation by α, the brightness value of this surface point in $image_\alpha$ is

$$E_\alpha = C\cos i_\alpha = C\frac{(q_0 q_l + \cos\alpha - p_0\sin\alpha)}{\sqrt{q_l^2 + 1}\sqrt{p_0^2 + q_0^2 + 1}}. \tag{7.34}$$

The computed p component at the point is

$$\tilde{p} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{Q^{-1}(E_\alpha)}{Q^{-1}(E_0)} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{\cos\tilde{e}_\alpha}{\cos\tilde{e}_0} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{E_\alpha/C(q_l^2 + 1) - q_l^2}{E_0/C(q_l^2 + 1) - q_l^2}$$

Substituting E_0 and E_α with Equation 7.33 and 7.34, we obtain

$$\tilde{p} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{(q_0 q_l + \cos\alpha - p_0\sin\alpha)\sqrt{q_l^2 + 1} - q_l^2\sqrt{p_0^2 + q_0^2 + 1}}{(q_0 q_l + 1)\sqrt{q_l^2 + 1} - q_l^2\sqrt{p_0^2 + q_0^2 + 1}}.$$

Similarly, the computed q component at the point is

$$\tilde{q} = \pm\sqrt{\left(\frac{\sqrt{p_0^2 + q_0^2 + 1}}{(q_0 q_l + 1)\sqrt{q_l^2 + 1} - q_l^2\sqrt{p_0^2 + q_0^2 + 1}}\right)^2 - \tilde{p}^2 - 1}.$$

Both $\tilde{p}$ and $\tilde{q}$ can be considered as functions of q_l. When q_l is small, the error in the computed surface orientation can be calculated by

$$\delta p = \tilde{p} - p_0 \approx \frac{d\tilde{p}}{dq_l}(0)q_l = q_l q_0 \frac{\cos\alpha - p_0 \sin\alpha - 1}{\sin\alpha},$$

and

$$\delta q = \tilde{q} - q_0 \approx \frac{d\tilde{q}}{dq_l}(0)q_l = \pm q_l[\frac{p_0(1-\cos\alpha)}{\sin\alpha} - (1+q_0^2)].$$

The above two equations give the errors in computing p and q for Lambertian surfaces illuminated under a non-collinear light source. For a uniform specular surface, if the illuminant direction is $(0, -q_l, 1)$ and the surface orientation at a point is $(-p_0, -q_0, 1)$, the error in the computed p and q components can be derived in the same way as that for a Lambertian surface(see Appendix B). For a specular surface,

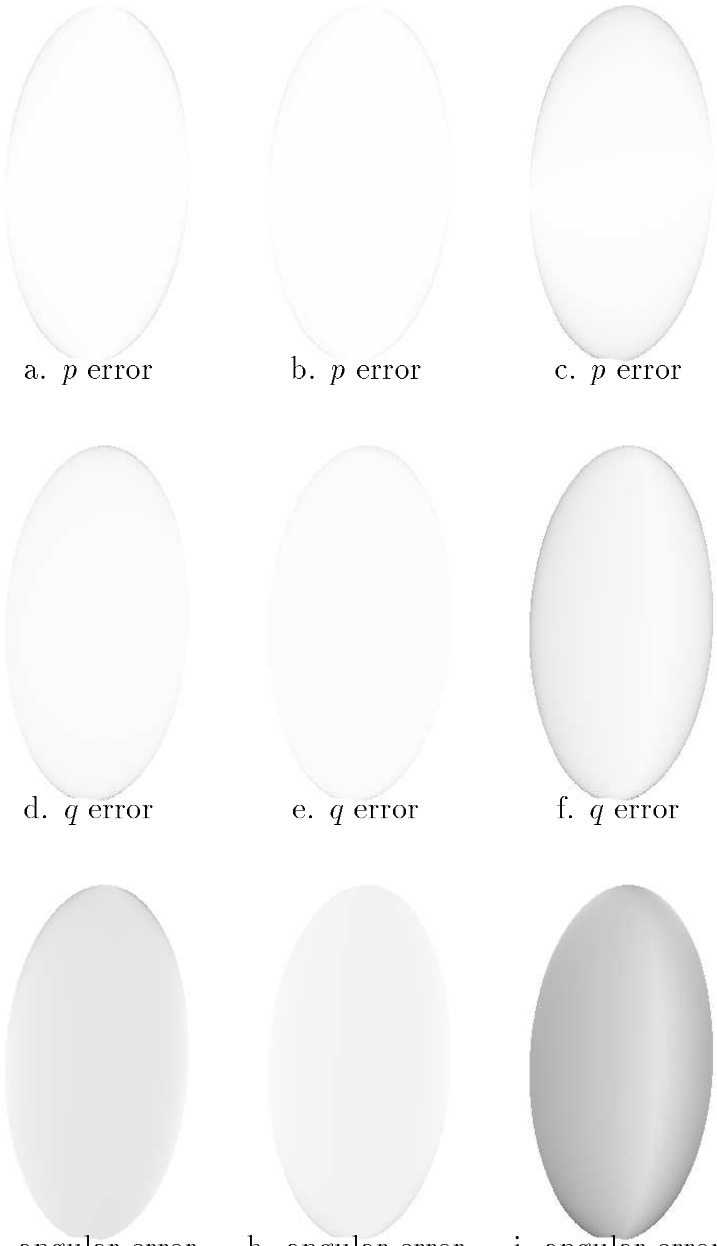
$$\delta p = \frac{q_l}{2} q_0 \frac{\cos\alpha - p_0 \sin\alpha - 1}{\sin\alpha},$$

$$\delta q = \pm\frac{q_l}{2}[\frac{p_0(1-\cos\alpha)}{\sin\alpha} - (1+q_0^2)].$$

Note that under a non-collinear light source, if two points have the same surface orientation, the error in the computed surface orientation for the point on a specular surface is half as much as that for the point on a Lambertian surface.

A lookup table can be used to compensate the surface orientation error. The above equations calculate the correspondence between the computed and actual surface orientation. The computed surface orientation determines the table location. The item in the corresponding table location gives the actual surface orientation.

For a uniform hybrid surface, if the illuminant direction is in $(0, -q_l, 1)$ and the surface orientation at a point is $(-p_0, -q_0, 1)$, the error in the computed surface orientation cannot be calculated exactly. But the error range for the p and q components can be determined (see Appendix C).



a. p error b. p error c. p error

d. q error e. q error f. q error

g. angular error h. angular error i. angular error

Figure 7.54: Orientation error caused by a non-collinear light source in the direction $(0, 0.033, 1)$. The first column is the error maps for a Lambertian surface. The second column is the error maps for a specular surface. The last column is the error maps of the maximum error for a hybrid surface.

In our imaging condition, the light source is mounted at the top of the camera. The distance from the object to the camera is 242cm and the distance between the optical center of the camera and the optical center of the light source is about 8cm. The angle between the illuminant direction and viewing direction is $\frac{8}{242} = 0.033$ radians so the illuminant direction is $(0, 0.033, 1)$. In surface recovery $image_0$ and $image_{\pi/18}$ are used so $\alpha = \pi/18$. Since the surface orientation at every point in $image_0$ of the ellipsoid is known as ground truth, the surface orientation error at every point caused by the non-collinear light source can be calculated. Three groups of error maps (see Fig. 7.54) are generated under the assumption that the ellipsoid has Lambertian, specular, or hybrid reflectance, respectively. Table 7.8 lists the corresponding p, q and angular r.m.s. errors under the assumption of the different reflectance properties. To make the results comparable to those computed from the real images and the synthetic images with noise, the data at the image points in the area, on which the r.m.s. errors caused by image noise are computed, is used for computing the r.m.s. error caused by the non-collinear light source. Under the hybrid surface assumption, the r.m.s. errors in p, q, and surface normal are computed by using the maximum of $|\delta p|$ and the maximum of $|\delta q|$ at every point.

Note that the error given under the assumption of the hybrid surface reflectance is the maximum error. The maximum error happens when a surface point gives Lambertian reflection in one image but specular reflection in the other image. This is not true for surface points on a real object surface. Therefore the error computed under the hybrid surface assumption should be smaller than the maximum error given in the table.

7.4.3 Effects of non-uniform reflectance

In our surface recovery process, we assume that the surface reflectance is uniform so the surface reflectance function extracted from the sample points can be applied to the whole

	Lambertian	specular	hybrid	real
p error	0.019	0.009	0.052	1.616
q error	0.045	0.022	0.0112	0.103
angular error	1.662°	0.838°	2.368°	7.279°

Table 7.8: The orientation error of an ellipsoid illuminated under a non-collinear light source in direction $(0, 0.033, 1)$.

surface. In practice, most surfaces are not uniform although they look uniform. The non-uniform reflectance will cause surface recovery error in the general case. However, for a Lambertian surface, the non-uniform reflectance has no effects in computing the p component.

For a Lambertian surface, the non-uniform reflectance is caused by changing albedo on the surface. The reflectance function of a Lambertian surface can be expressed as $E = Q(\cos e) = \lambda C \cos e$. The albedo λ, $(0 \leq \lambda \leq 1)$, specifies the fraction of the incident light reflected by the surface. It has certain value at each surface point. Let $\lambda(S)$ be the albedo at a sample point S and the surface reflectance function extracted from the sample point be $E = Q(\cos e) = \lambda(S) C \cos e$; then the inverse reflectance function is

$$\cos e = Q^{-1}(E) = \frac{E}{\lambda(S)C}. \tag{7.35}$$

For a surface point d, let $\lambda(d)$ be its albedo, E_0 be its brightness value in $image_0$ and E_α be its brightness value in $image_\alpha$; then $E_0 = \lambda(d) C \cos e_0$ and $E_\alpha = \lambda(d) C \cos e_\alpha$. Applying the inverse function 7.35 to the brightness values E_0 and E_α, we have

$$\frac{Q^{-1}(E_\alpha)}{Q^{-1}(E_0)} = \frac{\frac{E_\alpha}{\lambda(S)C}}{\frac{E_0}{\lambda(S)C}} = \frac{\frac{\lambda(d)C}{\lambda(S)C}\cos e_\alpha}{\frac{\lambda(d)C}{\lambda(S)C}\cos e_0} = \frac{\cos e_\alpha}{\cos e_0}.$$

The albedo change causes changes in both $Q^{-1}(E_0)$ and $Q^{-1}(E_\alpha)$. In computing $\frac{Q^{-1}(E_\alpha)}{Q^{-1}(E_0)}$, the change in $Q^{-1}(E_\alpha)$ is compensated by the change in $Q^{-1}(E_0)$. Therefore, the term

$\frac{Q^{-1}(E_\alpha)}{Q^{-1}(E_0)}$ is not affected by the albedo change so neither is the p component which is computed by $\tilde{p} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{Q^{-1}(E_\alpha)}{Q^{-1}(E_0)}$. If only the p component is used to compute surface depth, then the surface depth will not be affected by the non-uniform reflectance.

To compute the q component for point d, we have

$$\tilde{q} = \pm\sqrt{\frac{1}{(Q^{-1}(E_0))^2} - \tilde{p}^2 - 1} = \pm\sqrt{\frac{1}{(\frac{E_0}{\lambda(S)C})^2} - \tilde{p}^2 - 1} = \pm\sqrt{\frac{1}{(\frac{\lambda(d)}{\lambda(S)}\cos e_0)^2} - \tilde{p}^2 - 1}.$$

The term corresponding to the albedo change in the above expression is $\frac{\lambda(d)}{\lambda(S)}$. If $\lambda(d) \neq \lambda(S)$, the computed q component will be inaccurate.

For a specular or hybrid surface, the reflectance function under a collinear light source can be written as $E = Q(\cos e) = \lambda g(\cos e)$, where the variable λ $(0 \leq \lambda \leq 1)$, represents the reflectance factor at a surface point and it plays the same role as albedo in Lambertian reflectance function. The reflectance factor is a variable over a non-uniform surface. Let $\lambda(S)$ be the reflectance factor at a sample point S; then the surface reflectance function extracted from the sample point is $E = Q(\cos e) = \lambda(S)g(\cos e)$ and the inverse reflectance function is

$$\cos e = Q^{-1}(E) = g^{-1}(\frac{E}{\lambda(S)}). \tag{7.36}$$

For a surface point d, let $\lambda(d)$ be its reflectance factor, E_0 be its brightness value in $image_0$ and E_α be its brightness value in $image_\alpha$. Then $E_0 = \lambda(d)g(\cos e_0)$ and $E_\alpha = \lambda(d)g(\cos e_\alpha)$. Applying the inverse function 7.36 to the brightness values E_0 and E_α,

$$\tilde{p} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{g^{-1}(\frac{E_\alpha}{\lambda(S)})}{g^{-1}(\frac{E_0}{\lambda(S)})} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{g^{-1}(\frac{\lambda(d)}{\lambda(S)}g(\cos e_\alpha))}{g^{-1}(\frac{\lambda(d)}{\lambda(S)}g(\cos e_0))}. \tag{7.37}$$

Let $\delta\lambda = \lambda(S) - \lambda(d)$; then

$$g^{-1}(\frac{\lambda(d)}{\lambda(S)}g(\cos e_\alpha)) = g^{-1}(g(\cos e_\alpha) + \frac{\delta\lambda}{\lambda(S)}g(\cos e_\alpha)).$$

Since function g and g^{-1} are continuous so linear approximation can be used. Assuming $\delta\lambda$ sufficiently small, then

$$\begin{aligned} g^{-1}(g(\cos e_\alpha) + \tfrac{\delta\lambda}{\lambda(S)} g(\cos e_\alpha)) &\approx \cos e_\alpha + \tfrac{\delta\lambda}{\lambda(S)} \tfrac{g(\cos e_\alpha)}{g'(\cos e_\alpha)} \\ &= \cos e_\alpha (1 + \tfrac{\delta\lambda}{\lambda(S)} \tfrac{g(\cos e_\alpha)}{g'(\cos e_\alpha)\cos e_\alpha}). \end{aligned}$$

In the same way, we have

$$g^{-1}(\frac{\lambda(d)}{\lambda(S)} g(\cos e_0)) \approx \cos e_0 (1 + \frac{\delta\lambda}{\lambda(S)} \frac{g(\cos e_0)}{g'(\cos e_0)\cos e_0}).$$

The p component can be approximated by

$$\tilde{p} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha} \frac{\cos e_\alpha (1 + \frac{\delta\lambda}{\lambda(S)} \frac{g(\cos e_\alpha)}{g'(\cos e_\alpha)\cos e_\alpha})}{\cos e_0 (1 + \frac{\delta\lambda}{\lambda(S)} \frac{g(\cos e_0)}{g'(\cos e_0)\cos e_0})}.$$

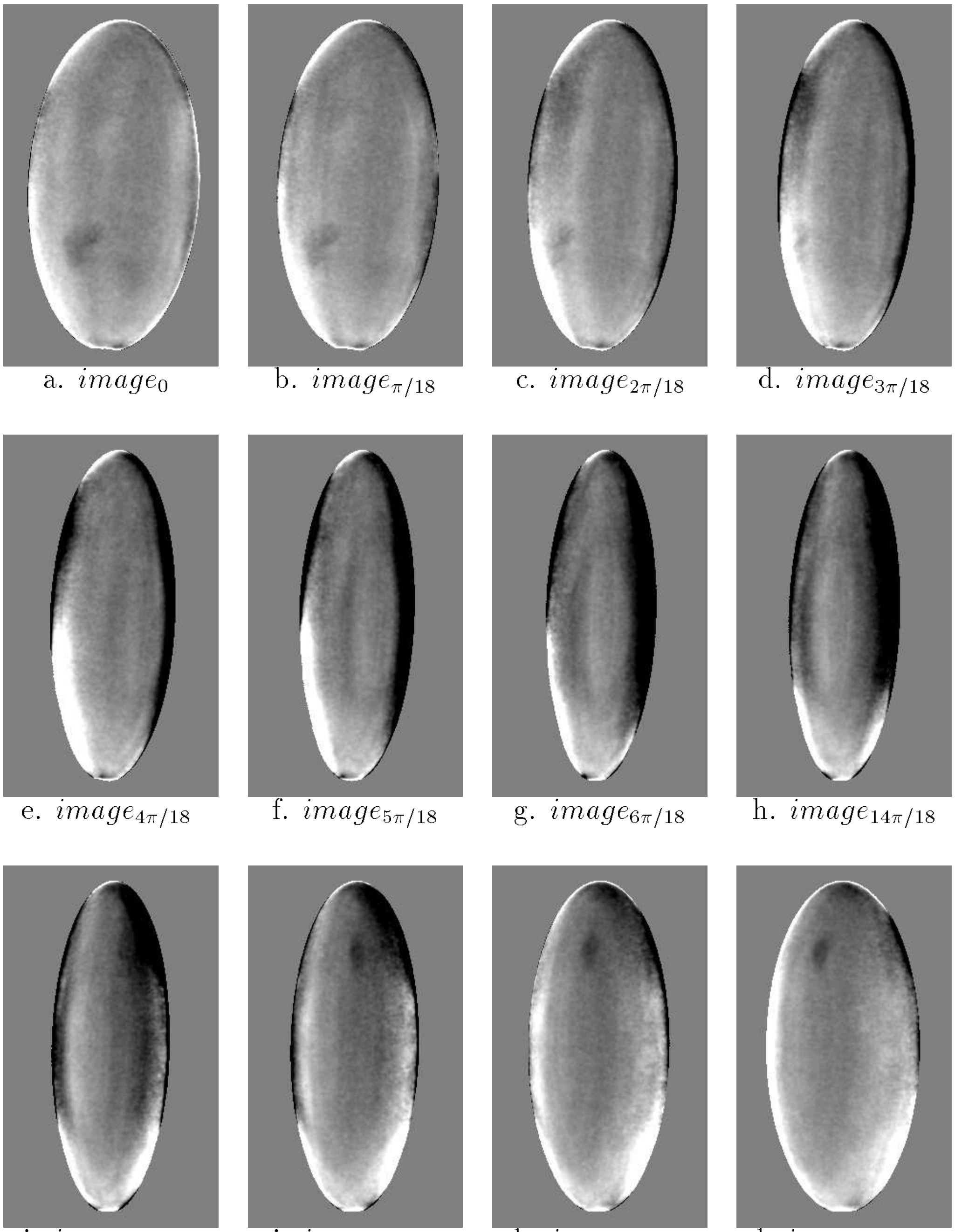
If the rotation angle α is small ($\alpha \le \pi/18$), the emergent angle of a surface point will not change very much after rotation. So $\cos e_0$ is close to $\cos e_\alpha$, $g(\cos e_0)$ is close to $g(\cos e_\alpha)$ and $g'(\cos e_0)$ is close to $g'(\cos e_\alpha)$. Therefore,

$$\frac{1 + \frac{\delta\lambda}{\lambda(S)} \frac{g(\cos e_\alpha)}{g'(\cos e_\alpha)\cos e_\alpha}}{1 + \frac{\delta\lambda}{\lambda(S)} \frac{g(\cos e_0)}{g'(\cos e_0)\cos e_0}} \approx 1.$$

When $\delta\lambda \ll \lambda(S)$, the above approximation will be more close to equality.

For a specular or hybrid surface, although the effects of the non-uniform reflectance in computing p can not be totally compensated, it remains fairly small. The error in the surface depth computed along the horizontal direction will not be as large as we first thought it might. However, the effect of the non-uniform reflectance in computing q is larger and the surface depth computed by using q component is less accurate.

To check if the ellipsoid used in analysis is uniform or not, the brightness values of its real images are compared with the brightness values of its synthetic images. Let the surface reflectance function extracted from the real images be $\lambda(S)g(\cos e)$ with $\lambda(S)$



a. $image_0$ b. $image_{\pi/18}$ c. $image_{2\pi/18}$ d. $image_{3\pi/18}$

e. $image_{4\pi/18}$ f. $image_{5\pi/18}$ g. $image_{6\pi/18}$ h. $image_{14\pi/18}$

i. $image_{8\pi/18}$ j. $image_{9\pi/18}$ k. $image_{10\pi/9}$ l. $image_{11\pi/18}$

Figure 7.55: The brightness variation in a U-map gives the uniformity variation on the object surface. Mid-gray(brightness=128) is zero variation, black(brightness=0) is -20% variation and white (brightness=255) is 20% variation.

as the reflectance factor of the sample point. In generating the synthetic images, the ellipsoid surface is assumed to be uniform and rendered with the surface reflectance function $\lambda(S)g(\cos e)$. The location and surface orientation of the ellipsoid in a synthetic image is the same as that in the corresponding real image. A U–map is defined as an image and the brightness value $I(x,y)$ at each pixel (x,y) in the map is determined by:

$$I(x,y) = \begin{cases} 128 + 127\frac{B_s(x,y)-B_r(x,y)}{0.2*B_s(x,y)} & \text{if} \quad -0.2 < \frac{B_s(x,y)-B_r(x,y)}{B_s(x,y)} < 0.2 \\ 0 & \text{if} \quad \frac{B_s(x,y)-B_r(x,y)}{B_s(x,y)} < -0.2 \\ 255 & \text{if} \quad \frac{B_s(x,y)-B_r(x,y)}{B_s(x,y)} > 0.2 \end{cases}$$

where $B_s(x,y) = \lambda(S)g(\cos e(x,y))$ is the brightness value in a synthetic image and $B_r(x,y) = \lambda(x,y)g(\cos e(x,y))$ is the brightness value in the corresponding real image. The threshold 0.2 is empirically determined. The quantity $(B_s(x,y) - B_r(x,y))/B_s(x,y) = \frac{\lambda(S)-\lambda(x,y)}{\lambda(S)}$ used in generating U–maps is the normalized variation of the reflectance factor at each corresponding surface point. Although the surface reflectance function of the ellipsoid may not be as simple as $\lambda(S)g(\cos e)$, the U–maps give a rough view on the variation of the reflectance factor.

Figure 7.55 shows a set of U–maps for $image_{k\pi/18}$ $(0 \le k \le 11)$ of the ellipsoid. From these maps we can observe some patterns moving during the object rotation. These patterns can be caused by either the non-uniformity of the surface reflectance or the local surface orientation changes. Either case will increase the difference between the recovered surface and the ground truth of the ellipsoid.

It is notable that in the U–maps the upper part of the ellipsoid is darker than the lower part. This phenomenon is caused by the non-collinear illumination. In generating the synthetic images, the illuminant direction is collinear. However, in taking real images, the illuminant direction is tilted on the plane of the viewing direction and rotation axis by a small angle. The non-collinear illumination causes the upper part of the ellipsoid to be brighter and the lower part to be darker. In generating the U–maps, when the brightness value at a pixel in the real image is brighter, the brightness value at the same pixel in the U–map will be darker and vice-versa.

7.4.4 Accumulated error

In our surface recovery, at an image point in $image_0$, its depth is used to compute the x coordinate of its correspondent in $image_\alpha$, and its brightness value and its correspondent's brightness value are used to compute its surface orientation, then its surface orientation is used to compute the depth for its neighboring points. An error in the depth will cause an error in the x coordinate of the correspondent and therefore an error in the brightness value of its correspondent. This in turn will cause an error in the surface orientation computed at this image point and then an error in the depth computed for the neighboring points. It seems that the depth error at one image point will cause a depth error at every following image point and the error will accumulate along the integration path. However this is not always the case. Even if the depth error accumulates in one direction along the path, we can change the integration direction to avoid accumulating error.

When the depth and surface orientation are computed in the positive x direction, $\Delta x = 1$ and the depth z at the next image point N is

$$z(N) = z_0 + p_0 \quad with \quad p_0 = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{Q^{-1}(E_\alpha)}{Q^{-1}(E_0)},$$

where z_0 and p_0 are the depth and the p component at the current image point (x_0, y_0) in $image_0$, E_0 is the brightness value of (x_0, y_0) and E_α is the brightness value of the image correspondent in $image_\alpha$. The x coordinate of the correspondent in $image_\alpha$ is computed by $x_\alpha = x_0 \cos\alpha - z_0 \sin\alpha$. If there is an error δz_0 in the depth z_0, the computed x coordinate of the correspondent in $image_\alpha$ will become $\tilde{x}_\alpha = x_\alpha + \delta x_\alpha$ with $\delta x_\alpha = -\delta z_0 \sin\alpha$. And the brightness value of the correspondent will become

$\tilde{E}_\alpha = E_\alpha + \delta E_\alpha$, where δE_α is caused by δx_α. Then the computed p component at image point (x_0, y_0) is

$$\tilde{p}_0 = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{Q^{-1}(\tilde{E}_\alpha)}{Q^{-1}(E_0)} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{Q^{-1}(E_\alpha + \delta E_\alpha)}{Q^{-1}(E_0)}.$$

Assuming δE_α is small and applying linear approximation to $Q^{-1}(E_\alpha + \delta E_\alpha)$, we have

$$Q^{-1}(E_\alpha + \delta E_\alpha) = Q^{-1}(E_\alpha) + [Q^{-1}(E_\alpha)]'\delta E_\alpha = Q^{-1}(E_\alpha) + \frac{\delta E_\alpha}{Q'(\cos e_\alpha)} \;\&$$

$$\tilde{p} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\left(\frac{Q^{-1}(E_\alpha)}{Q^{-1}(E_0)} + \frac{\delta E_\alpha}{Q'(\cos e_\alpha)Q^{-1}(E_0)}\right) = p_0 - \frac{1}{\sin\alpha}\frac{\delta E_\alpha}{Q'(\cos e_\alpha)\cos e_0} \tag{7.38}$$

where $Q'(\cos e_\alpha)$ is the derivative of Q at $\cos e_\alpha$. The term containing δE_α represents the error in $\tilde{p}_0$. Let the surface orientation at the correspondent (x_α, y_α) be $(-p_\alpha, -q_\alpha, 1)$; then $\cos e_\alpha = \frac{1}{\sqrt{p_\alpha^2 + q_\alpha^2 + 1}}$ and

$$\delta E_\alpha = \frac{dQ(\cos e_\alpha)}{d(\cos e_\alpha)}\frac{\partial \cos e_\alpha}{\partial x_\alpha}\delta x_\alpha = \cos^3 e_\alpha Q'(\cos e_\alpha)(r_\alpha p_\alpha + t_\alpha q_\alpha)\delta z_0 \sin\alpha \tag{7.39}$$

where $r_\alpha = \frac{\partial p_\alpha}{\partial x_\alpha}$ and $t_\alpha = \frac{\partial q_\alpha}{\partial x_\alpha}$. Substituting δE_α in Equation 7.38 with 7.39, we have

$$\tilde{p}_0 = p_0 - \frac{\cos^3 e_\alpha (r_\alpha p_\alpha + t_\alpha q_\alpha)}{\cos e_0}\delta z_0.$$

Then the depth for the next image point is computed by

$$\begin{aligned} \tilde{z}(N) &= (z_0 + \delta z_0) + \tilde{p}_0 = z_0 + \delta z_0 + p_0 - \tfrac{\cos^3 e_\alpha (r_\alpha p_\alpha + t_\alpha q_\alpha)}{\cos e_0}\delta z_0 \\ &= (z_0 + p_0) + \left(1 - \tfrac{\cos^3 e_\alpha (r_\alpha p_\alpha + t_\alpha q_\alpha)}{\cos e_0}\right)\delta z_0. \end{aligned}$$

The term which contains δz_0 in the above expression represents the depth error at the next image point. Since $\cos e_\alpha \geq 0$ and $\cos e_0 \geq 0$, depending on the sign of $(r_\alpha p_\alpha + t_\alpha q_\alpha)$, the factor before δz_0 can be larger or smaller than 1. When $(r_\alpha p_\alpha + t_\alpha q_\alpha) < 0$, the factor is larger than 1, and the depth error will increase at the next image point. Otherwise, the factor is smaller than one and the depth error will be partially compensated at the next image point. The magnitude of the factor is also related to the surface curvature in the x direction. The higher the curvature, the larger the error increase or compensation and vice-versa.

If the depth is computed in the negative x direction, then the depth for the next image point is

$$\tilde{z}(N) = z_0 - p_0 + \left(1 + \frac{\cos^3 e_\alpha (r_\alpha p_\alpha + t_\alpha q_\alpha)}{\cos e_0}\right) \delta z_0.$$

Just opposite to the depth computed in the positive x direction, when $(r_\alpha p_\alpha + t_\alpha q_\alpha) > 0$, the depth error will increase at the next image point. Otherwise, the depth error will be partially compensated at the next image point. By computing the depth along the right direction, the depth error accumulation can be avoided. Assuming that $(r_\alpha p_\alpha + t_\alpha q_\alpha)$ does not change its sign within a small area, by examining the change of the computed depth at the next image point with a small perturbation on the depth at the current image point, we can determine which direction is the right direction for reducing the depth error.

Figure 7.56 (a) shows a cross section of a piece of circular cylinder after it has been rotated by α. We can choose any of the three points of known depths, a, b or c, as the starting point for depth recovery. Since $t = 0$ for a cylindrical surface, the sign of $(rp + tp)$ is determined by rp. Between a and b, $rp < 0$ and between b and c, $rp > 0$. If we choose a as the starting point and move along from a to b, the depth error will accumulate. If we choose b as the starting point and move along from b to a, the depth error will be reduced. It also can be shown that depth recovery from b to c will be more accurate than from c to b. The same analysis can be applied to a piece of cylinder with the other side towards the viewing direction as shown in Fig. 7.56 (b). We can verify that the directions for reducing the depth error are the same.

In surface recovery from the real images of the ellipsoid, we applied some perturbations on the depth at the starting point at which both p and q components are zero, then

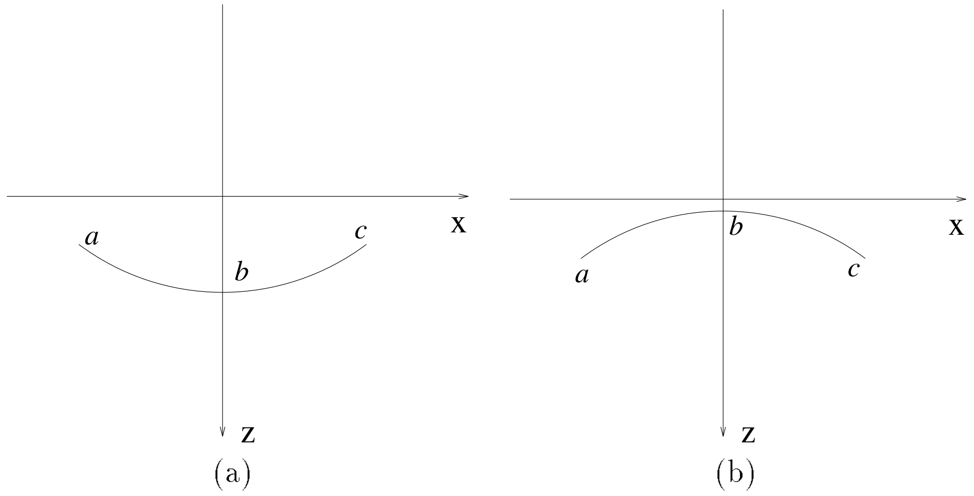


Figure 7.56: The cross section of a piece of circular cylinder with the different side towards the viewing direction.

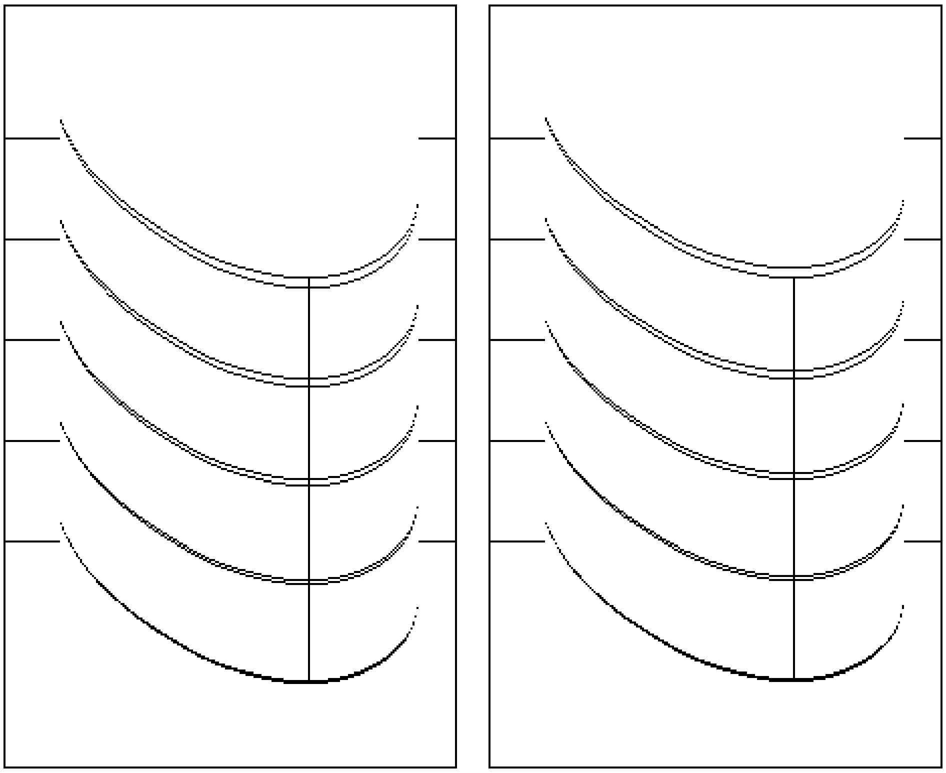
compared the recovered depths with the real depths. The depth perturbation at the starting point is from -5 to 5 pixels. Figure 7.57 (a) shows the profiles of the real depth and the depth recovered with perturbation 5 to 1. Figure 7.57 (b) shows the profiles of the real depth and the depth recovered with perturbation -5 to -1. The vertical line indicates the x location of the starting point. From Fig. 7.57, we can see that the depth error becomes smaller and smaller away from the staring point. Table 7.9 gives r.m.s. and average errors corresponding to the different amounts of perturbation. The table shows that both r.m.s. error and average error of the recovered depth are smaller than the perturbation at the starting point. This means that the depth errors are reduced during the computation.

7.4.5 Orthographic projection

In our surface recovery, orthographic projection is used as an approximation for the image projection. The error caused by the approximation can be analyzed using a pin

perturbation	5	4	3	2	1	-1	-2	-3	-4	-5
r.m.s.	3.69	2.94	2.21	1.48	0.74	0.74	1.48	2.21	2.95	3.69
average	3.17	2.54	1.91	1.27	0.64	−0.63	−1.27	−1.90	−2.54	−3.17

Table 7.9: The depth error caused by the depth perturbation at the starting point.



(a) (b)

Figure 7.57: The vertical line indicates the x coordinate of the starting point. (a) shows the depth profiles recovered with perturbation 5 to 1 (5 at top). (b) shows the depth profiles recovered with perturbation -5 to -1 (-5 at top). The real depth profile is overlaid with the computed profile.

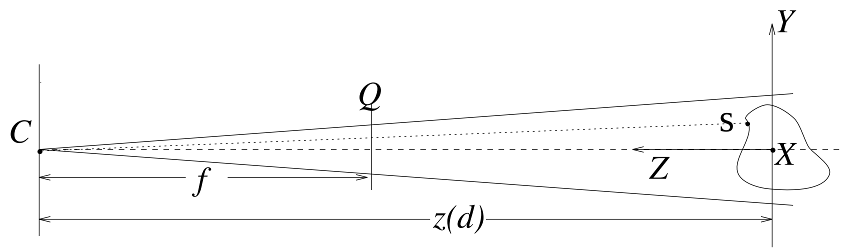


Figure 7.58: A pin hole camera model. The image plane Q is parallel to the xy plane. The focal length f is the distance from the viewing point C to the image plane Q and $z(d)$ is the distance from C to the xy plane.

hole camera model shown in Fig. 7.58. In the figure, the y axis is the rotation axis of the object, the z axis lies along the optical axis of the camera, the focal length f is the distance from the viewing point C to the image plane Q which is parallel with the XZ plane, $z(d)$ is the distance from C to the XY plane. In taking images, the focal length is fixed. In surface recovery, the distance between image pixels is used as the unit for the 3D coordinates of surface points. If all the surface points are on a plane $z = z(r)$, the transformation from the world coordinate (x, y, z) to the pixel measured coordinate $(\bar{x}, \bar{y}, \bar{z})$ will be

$$\bar{x} = kx; \quad \bar{y} = ky; \quad \bar{z} = kz.$$

with $k = \frac{f}{z(d)-z(r)}$ as a constant. This transformation is called scaled orthographic projection and the plane $z = z(r)$ is called the reference plane. However, a generic surface changes its depth from point to point. Let the depth of a surface point s be $z(s)$ and $\delta z(s) = z(s) - z(r)$ be the depth change with respect to the reference plane, the change of the image coordinates of s caused by the change of the depth $\delta z(s)$ can be calculated by

$$\delta\bar{x}(s) = \frac{d\bar{x}(s)}{dz(s)}\delta z(s) = \bar{x}(s)\frac{\delta z(s)}{z(d)-z(s)}; \qquad \delta\bar{y}(s) = \frac{d\bar{y}(s)}{dz(s)}\delta z(s) = \bar{y}(s)\frac{\delta z(s)}{z(d)-z(s)}.$$

The relative change of the image coordinates is

$$\frac{\delta \bar{x}(s)}{\bar{x}(s)} = \frac{\delta z(s)}{z(d) - z(r)}; \qquad \frac{\delta \bar{y}(s)}{\bar{y}(s)} = \frac{\delta z(s)}{z(d) - z(r)}$$

In our surface recovery, the object center is close to the rotation axis, and the maximum depth of our object's surface is about 4cm. The distance from the camera to the rotation axis is about 242cm. Assume the depths of all the visible surface points are larger than zero and $z = 2cm$ is the reference plane; then for any surface point of s, the relative error in the image coordinates under the scale orthographic projection is within $\pm\frac{2}{240} = \pm 0.0083$. This error is so small that it will not cause visual effects on the recovered surface.

In the photogeometric technique, for image points in $image_0$, their correspondents in $image_\alpha$ are computed by the rotation transformation. Let the world coordinates of a surface point be (x_0, y_0, z_0) when the $image_0$ is taken and be $(x_\alpha, y_\alpha, z_\alpha)$ when $image_\alpha$ is taken; then

$$\bar{x}_0 = x_0 \frac{f}{z(d) - z_0}; \qquad \bar{y}_0 = y_0 \frac{f}{z(d) - z_0}.$$

and

$$\bar{x}_\alpha = x_\alpha \frac{f}{z(d) - z_\alpha}; \qquad \bar{y}_\alpha = y_\alpha \frac{f}{z(d) - z_\alpha}.$$

If $z_\alpha = z_0$, then $\bar{x}_0 = kx_0$, $\bar{x}_\alpha = kx_\alpha$, $\bar{y}_\alpha = ky_\alpha$, $\bar{y}_0 = ky_0$ with $k = \frac{f}{z(d) - z_0}$. In this case, there is no error in the location of the image correspondent. If $z_0 \neq z_\alpha$, there will be an error in the location of the correspondent. The error is determined by the difference of z_0 and z_α and can be calculated by

$$\delta \bar{x}_\alpha = \bar{x}_\alpha \frac{z_\alpha - z_0}{z(d) - z_0}; \qquad \delta \bar{y}_\alpha = \bar{y}_\alpha \frac{z_\alpha - z_0}{z(d) - z_0}.$$

In our experiment, the ellipsoid is confined within a cylinder of diameter $7.4cm$ around the rotation axis and the rotation angle α is $\pi/18$. The maximum depth change for a surface point before and after the object rotation by α is $7.2 \div 2x \sin\alpha = 0.62cm$. Taking $z(d) - z_0$ as 240cm, the maximum relative errors in the x, y image coordinates of the correspondent are $\delta\bar{x}_\alpha = 0.0026\bar{x}_\alpha$ and $\delta\bar{y}_\alpha = 0.0026\bar{y}_\alpha$. In the images of the ellipsoid used in our analysis, $-93 < \bar{x} < 93$ and $-176 < \bar{y} < 176$. Therefore the maximum errors in the x and y coordinates of the correspondent are 0.25 pixel and 0.45 pixel respectively. When the surface curvature is small, the small change in the image coordinate of the correspondent will cause a small change in the brightness of the correspondent. Therefore there is no substantial effect in surface recovery. However when a surface point is far away from the optical axis and the curvature at the point is large, the non-orthographic projection may cause surface recovery error. After the initial depth recovery, some compensation for non-orthographic projection can be made to get more accurate results. It is also possible to buy orthographic lenses, which ensure orthographic projection, but they are quite expensive.

7.4.6 Other factors

Besides image noise, non-collinear illumination, non-uniform surface reflectance and error accumulation, there are some other factors which cause errors in surface recovery. One important factor is interreflection.

Interreflection is the light reflection among different objects or among the different elements of an object surface. Its effects lead to complicated structures in scene radiance. Accounting for its effects in surface recovery is very difficult in the general case. Formulation and analysis of interreflection have been done by Koenderink and Van Doorn [40] as well as Forsyth and Zisserman [20]. Quantitative treatments of interreflection in shape recovery have been conducted by Nayar *et al.* [49] and Wada *et al.* [72]. The effects of interreflection are ignored in our surface recovery, since interreflection did not cause as serious a problem as we thought it might. Most of the vase is concave, yet surface recovery worked quite well. However, in highly curved concave regions, such as the joint on the vase handle, the effects of interreflection are substantial. Without explicitly modeling interreflection, we still can make some qualitative analysis on its effects. In our surface recovery, the interreflection effects are partially compensated in the calculation of the p component of the surface orientation. This in turn reduces the effects on the surface depth computed by integrating the p component. In computing the p component, the quantity $\frac{Q^{-1}(E_\alpha)}{Q^{-1}(E_0)}$ is used. In the case of interreflection, both E_α and E_0 become larger. Since Q^{-1} is strictly monotonic, both $Q^{-1}(E_\alpha)$ and $Q^{-1}(E_0)$ become larger. Therefore the effects caused by interreflection are partially compensated in computing $\frac{Q^{-1}(E_\alpha)}{Q^{-1}(E_0)}$. Quantitative analysis of the interreflection effects is difficult and will be investigated in future experiments.

The surface curvature of the object also affects the accuracy of the recovered surface. In surface recovery, the surface depth is obtained by integrating the first partial derivatives only. The terms corresponding to the high order partial derivatives are ignored. The error caused by ignoring these terms is directly related to the surface curvature of the object since the magnitudes of the high order partial derivatives are proportional to the surface curvature. When the surface curvature of the object is small, the error will be small. Otherwise it will be large. As derivatives are computed on pixels, the object should be imaged as large as possible.

The x coordinate of the reference system is defined by the rotation axis of the object. The error in the location of the rotation axis will cause errors in computing the locations of surface points and their image projections during rotation. The following calculation will show how much the surface recovery will be affected by the error. Let (x_0, y_0, z_0) be the initial location of a surface point. After the object has been rotated by α, the location of the surface point can be calculated as $(x_\alpha, y_\alpha, z_\alpha)$. Let δx be an error in the location of the rotation axis in images; then after α rotation of the object, the location of the surface point is calculated as $(\tilde{x}_\alpha, \tilde{y}_\alpha, \tilde{z}_\alpha)$ with

$$\begin{aligned}
\tilde{x}_\alpha &= (x_0 - \delta x)\cos\alpha - z_0 \sin\alpha &&= x_\alpha - \delta x \cos\alpha; \\
\tilde{y}_\alpha &= y_0 &&= y_\alpha; \\
\tilde{z}_\alpha &= (x_0 - \delta x)\sin\alpha + z_0 \cos\alpha &&= z_\alpha - \delta x \sin\alpha.
\end{aligned}$$

Since the rotation axis is also shifted δx in $image_\alpha$, the computed x coordinate of the surface point in $image_\alpha$ is $\tilde{x}_\alpha + \delta x = x_\alpha - \delta x \cos\alpha + \delta x$. The errors caused by the shift δx of the rotation axis are

$$\delta x_\alpha = (1 - \cos\alpha)\delta x \quad \& \quad \delta z_\alpha = \delta x \sin\alpha.$$

The above error will happen in two cases in surface recovery. One is in computing the image correspondent in $image_\alpha$ for an image point in $image_0$. In this case, the rotation angle is usually small so that the error is small. For example, if $\alpha = \pi/18$, $\delta x_\alpha = (1 - \cos\alpha)\delta x = 0.0152\delta x$. Thus δx has to be very large to cause a substantial error. The other is in computing the depth for sample image point. In that case, the rotation angle is $\pi/2$ and the error in the depth is δx. In practice, the location of the rotation axis can be determined with subpixel accuracy (see Section 6.5) so the error in the location of the rotation axis can be ignored.

The distance of the object center from the rotation axis of the rotary table affects surface recovery. If the distance is large the dynamic range of the distance from the surface to the light source will be large since the light source is fixed. When the object is close to the light source, it will appear to be brighter than it does when the object is far from the light source. This will violate the assumption that the brightness value on the object surface only depends on the surface orientation. For best results, the object center should be located at the rotation center. However, within a small dynamic range, the effect is so small that it can be ignored.

In surface recovery, $image_0$ and $image_\alpha$ are used. How large should the rotation angle α be? In computing the p component at an image point, $p = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{Q^{-1}(E_\alpha)}{Q^{-1}(E_0)}$. If there is an error in $Q^{-1}(E_\alpha)$, the error will be scaled by a factor of $\frac{1}{\sin\alpha}$. So α should be large. On the other hand, the x coordinate of the image correspondent in $image_\alpha$ is computed by $x_\alpha = x_0 \cos\alpha - z_0 \sin\alpha$. If there is error in depth, a larger rotation angle will cause a larger error in the location of the image correspondent in $image_\alpha$. In this case, α should be small. The other considerations include that the two images should cover as much common surface area as possible so that larger surface area can be recovered from the two images. The conclusion is that the rotation angle should neither too large nor too small. Empirical experiments show that α should be about $\pi/18$.

7.4.7 Contour comparison

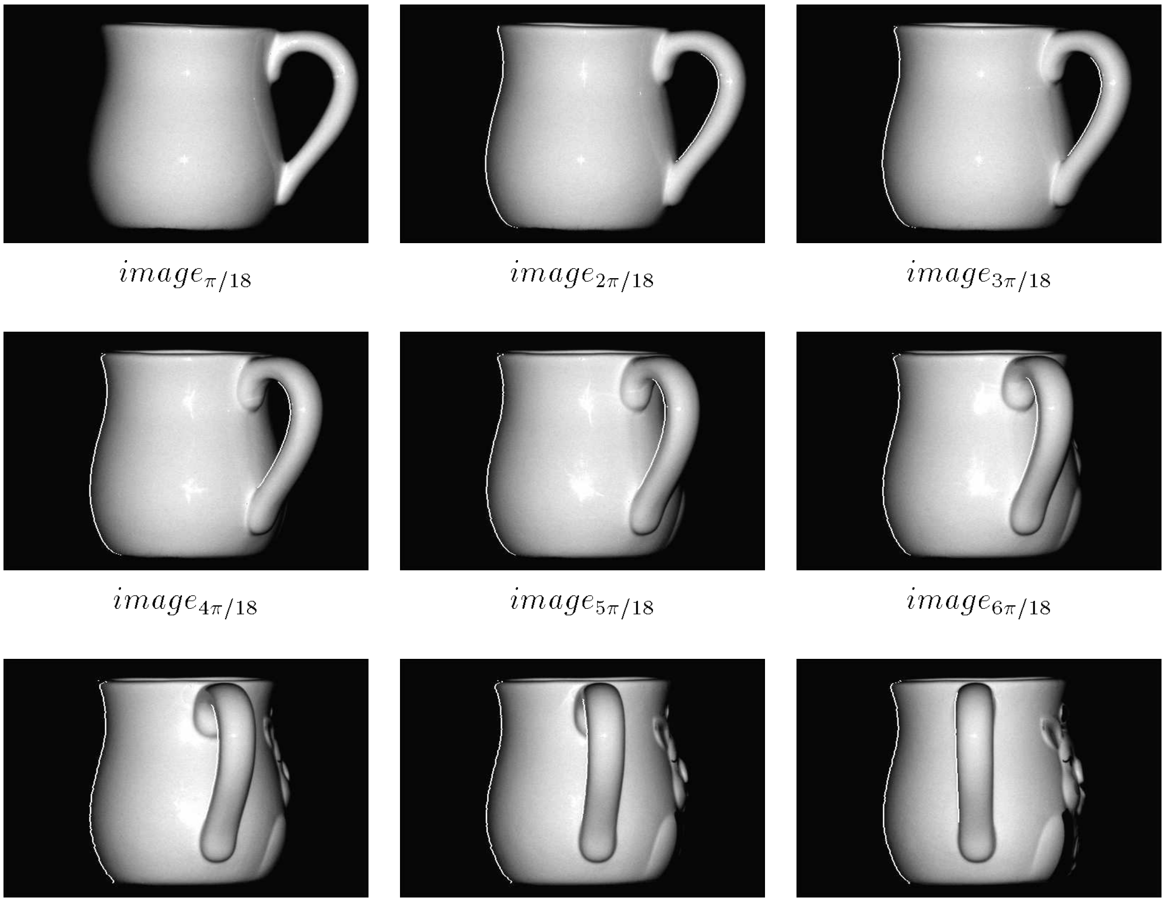
We recovered the surfaces of a clay vase and a porcelain cup from their images. Although there is no ground truth on the depth and surface orientation of these two objects, the occluding contours in the images of the objects can be used to judge how well the surfaces have been recovered. Assuming the recovered surface is rotated and viewed in the same way as the original surface, the recovered surface can be projected onto the original images of the object. The distance between the projected occluding contours and the original occluding contours shows the accuracy of the recovered surface.

The recovered surface is represented as the depth and surface orientation at each pixel in $image_0$ and these data are related to the 3D coordinates before object rotation. A 3D curve of $p = \tan(\pi/2 - \alpha)$ will be an occluding boundary after the recovered surface has been rotated by α. Thus the occluding contours in $image_\alpha$ are computed by searching for points of $p = \tan(\pi/2 - \alpha)$ on the recovered surface, then rotating these points by α around the y axis and projecting these points on $image_\alpha$. The search is conducted row by row in $image_0$. Generally, an occluding boundary is a part of continuous curve on the surface. To make the search more efficient, the results of the current row are used in the search in the next row. To find the more accurate 3D locations of these curves, the depth and surface orientation are interpolated between pixels in $image_0$.

Figure 7.59 and 7.60 show the occluding contours projected from the recovered surface on the original images of the cup. Only the occluding contours projected from the occluding boundaries of $p = +\infty$ are shown. To avoid confusion, the occluding contours in the original images are not shown in the images. The last image in Fig. 7.60 shows the projection of all 3D curves, which generate the occluding contours, on $image_0$. In the real images of the cup, some occluding contours are occluded by another part of the cup. In generating the occluding contours of the recovered surface, we don't consider the occluding effects. In some images (for example, $image_{13\pi/18}$), although the handle blocks some segments of the occluding contour on the body of the cup, those segments are still shown in the image. In the surface recovery of the cup, $image_0$ and $image_{\pi/18}$ are used. The depth and surface orientation at the image points on or close to the occluding contours in $image_0$ and $image_{\pi/18}$ cannot be recovered(see Section 4.5), so there is no occluding contour in $image_0$ and $image_{\pi/18}$.

From Fig. 7.59 and Fig. 7.60, we can have a rough visual impression on how close the recovered surface is to the original surface. The curves of constant p are not smooth and even not continuous at some points because of surface recovery error. From the last image in Fig. 7.60 and the occluding contours in $image_{16\pi/18}$ and $image_{17\pi/18}$, we can see that the surface orientation data at the joints between the handle and the body are not well recovered. Interreflection between the handle and the body is considered as the major problem in recovering the joints.

Table 7.10 gives a quantitative analysis of the distance between the original and the projected contours. The distance is measured from a projected contour point $(\tilde{x}_i, \tilde{y}_i)$ to the closest original contour point (x_i, y_i) in the horizontal direction. The r.m.s. distance



$image_{\pi/18}$ $image_{2\pi/18}$ $image_{3\pi/18}$

$image_{4\pi/18}$ $image_{5\pi/18}$ $image_{6\pi/18}$

$image_{7\pi/18}$ $image_{8\pi/18}$ $image_{9\pi/18}$

Figure 7.59: $image_{\pi/18}$–$image_{9\pi/18}$ superimposed with the occluding contours projected from the recovered surface

$image_{10\pi/18}$

$image_{11\pi/18}$

$image_{12\pi/18}$

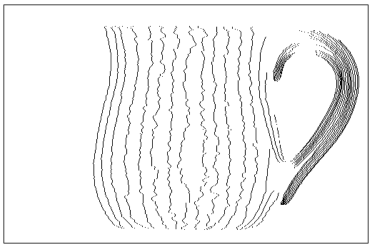
$image_{13\pi/18}$

$image_{14\pi/18}$

$image_{15\pi/18}$

$image_{16\pi/18}$

$image_{17\pi/18}$



Projection on $image_0$

Figure 7.60: $image_{10\pi/18}$–$image_{17\pi/18}$ superimposed with the occluding contours projected from the recovered surface. The last diagram shows the projection of the curves, which generate the occluding contours, on $image_0$.

image	$2\pi/18$	$3\pi/18$	$4\pi/18$	$5\pi/18$	$6\pi/18$	$7\pi/18$	$8\pi/18$	$9\pi/18$
r.m.s. distance	1.816	1.860	1.982	2.178	1.648	1.430	1.036	0.500
average distance	1.555	1.371	1.253	1.167	0.239	-0.094	-0.3445	-0.075

image	$10\pi/18$	$11\pi/18$	$12\pi/18$	$13\pi/18$	$14\pi/18$	$15\pi/18$	$16\pi/18$	$17\pi/18$
r.m.s. distance	1.085	2.072	2.388	2.610	3.041	2.477	2.178	2.082
average distance	0.594	0.582	1.604	1.775	2.512	1.981	1.861	1.718

Table 7.10: The first row is the subscripts for the images. The second and the third row are, respectively, r.m.s. and average distances between the original contours and the projected contours.

is calculated by $\sqrt{\frac{\sum(\tilde{x}_i - x_i)^2}{N}}$ and the average distance is calculated by $\frac{\sum(\tilde{x}_i - x_i)}{N}$ with N as the number of contour points used in the calculation. For some projected contour points, the corresponding original contour points cannot be found in the image because of occlusion effects and failure in contour detection. There are in total 8163 projected contour points in these images and 7499 of them are used in the calculation. The first row in the table is the subscripts for images. The second row and the third row are the r.m.s. distance and average distance between the original and the projected contours. In $image_{9\pi/18}$, both the r.m.s. and average distances are the smallest. This is because the depth on the $p = 0$ curve of the recovered surface is obtained directly from the the original occluding contour in $image_{9\pi/18}$ and the surface recovery starts from the $p = 0$ curve so that the accumulated error is the smallest compared to other curves of constant p. Considering that the distance from a surface point on the body of the cup to the rotation axis is roughly about 130 pixels, the r.m.s. and average distances listed in the table show that the depth recovery error is relatively small.

The contour comparison method described here can only give a rough evaluation on the recovered surface. However, for a convex object, if the accuracy on the locations of the occluding contours in images is known (this is true in the general case), from the difference between the original contours and the projected contours, the error range of the recovered surface can be determined. For general objects with concave areas, the surface recovery error cannot be determined by the contour comparison method.

7.5 Analysis of the Wire-frame Technique

The wire-frame technique described in Chapter 5 extracts a set of 3D curves from an image sequence. The experiments in Chapter 5 show that the technique is robust. In fact, a collinear light source is not necessary for extracting the 3D curves and the object surface does not have to be uniform. We will analyze the effects of different factors on the location error of the extracted curves.

7.5.1 Coplanar light source

A *coplanar light source* is a light source of which the illuminant direction is coplanar with the viewing direction and the rotation axis. To extract a wire frame, a coplanar light source is sufficient since the equi-brightness constraint still holds for points on a $p = 0$ curve. The image brightness value of a surface point under a coplanar light source can be expressed by the general surface reflectance function of the incident angle i, the emergent angle e and the phase angle g. As the light source and camera are fixed, the phase angle g is constant during object rotation. We can show that, for a surface point of $p = 0$, the angle i as well as the angle e is the same, therefore its image brightness value is the same, after the object has been rotated by two symmetric angles. Furthermore, under the assumption of orthographic projection, the corresponding occluding contour of a $p = 0$ curve under a coplanar light source is the same as that under a collinear light source. Since both the geometric and photometric constraints are unchanged, the wire-frame technique can be applied to surfaces illuminated under a coplanar light source. The wire frame extracted will be the same as that extracted under a collinear light source except the part of the wire frame in the self-shadowing area. The self-shadowing area is caused by the angle between the illuminant direction and viewing direction. If the angle is small the difference between the two wire frames will also be small.

7.5.2 Piecewise uniform surface

The assumption of uniform surface reflectance is not necessary for extracting a wire frame on the object surface. The surface reflectance can be piecewise uniform. The object could have areas of different colors or different materials. For an object of piecewise uniform reflectance, the equi-brightness constraint for $p = 0$ curves still holds and the corresponding occluding contours still can be extracted in the same way as before. The only thing we may need to do is to segment the images into different areas of the same reflectance to avoid mismatch in search for correspondence. Segmentation is unnecessary if good starting points are available to guide the search. This is always the case since the singular points are on $p = 0$ curves. It is easy to identify the singular points and to find their depth from the corresponding contour points. On the other hand, the boundaries between areas of different reflectance can be used as surface markings and conventional stereo techniques can be applied to compute the 3D location at the boundaries.

7.5.3 The depth on a wire frame

The depth on a $p = 0$ curve is determined by the x coordinates on its corresponding occluding contour. The corresponding contour can be a background-occluding contour or a self-occluding contour. A background-occluding contour is extracted by thresholding the background brightness. The position on an extracted background-occluding contour is mainly affected by the quantization of image space so that the position usually can be determined within one pixel accuracy. A self-occluding contour causes a sharp brightness drop in images. In the images we used in our experiments, the width of a sharp change caused by self-occluding is usually less than 4 to 5 pixels. The Canny edge detector is used to locate a self-occluding contour. An edge point generated by the Canny edge detector is at the middle of a sharp drop. Thus the maximum location error of a self-occluding contour located by using the Canny edge detector will be less than 2 to 2.5 pixels. The actual location of an occluding contour point should be at the image point where the sharp brightness drop happens. The error can be reduced by finding the exact location where the brightness drop starts.

7.5.4 The x coordinate on a wire frame

The error in the location of an occluding contour affects not only the depths but also the x coordinates computed on the corresponding $p = 0$ curve. The error in the location of an occluding contour point first affects the disparty of the two corresponding images points. Let (x_α, y_α) and $(x_{-\alpha}, y_{-\alpha})$ be the images of a $p = 0$ surface point in $image_\alpha$ and $image_{-\alpha}$, respectively. From the constraints on a $p = 0$ point, we have $D = x_\alpha - x_{-\alpha} = 2x_{\pi/2} \sin \alpha$ and $E(x_\alpha, y_\alpha) = E(x_{-\alpha}, y_{-\alpha})$, where D is the disparity of the two image points and $x_{\pi/2}$ is the x coordinate of the corresponding contour point in $image_{\pi/2}$(see Section 5.3). Since all the image points under consideration have the same y coordinates, the y coordinates will be omitted in the following expressions. If there is an error $\delta x_{\pi/2}$ in $x_{\pi/2}$, there will be an error $\delta D = 2\delta x_{\pi/2} \sin \alpha$ in disparity D. The disparity error in turn will cause an error in the x locations of the two corresponding image points determined by the search method described in Section 5.3. Let the computed disparity be

$$\tilde{D} = x_\alpha - x_{-\alpha} + \delta D, \tag{7.40}$$

and let the x coordinates of the two image points determined by searching for the corresponding image points which have the computed disparity $\tilde{D}$ and the same brightness value be $\tilde{x}_\alpha = x_\alpha + \delta x_\alpha$ and $\tilde{x}_{-\alpha} = x_{-\alpha} + \delta x_{-\alpha}$, respectively. Then

$$E(x_\alpha + \delta x_\alpha) = E(x_{-\alpha} + \delta x_{-\alpha})$$

and

$$\tilde{D} = (x_\alpha + \delta x_\alpha) - (x_{-\alpha} + \delta x_{-\alpha}) = x_\alpha - x_{-\alpha} + (\delta x_\alpha - \delta x_{-\alpha}). \tag{7.41}$$

Using linear approximation yields

$$E(x_\alpha) + \frac{\partial E(x_\alpha)}{\partial x_\alpha}\delta x_\alpha = E(x_{-\alpha}) + \frac{\partial E(x_{-\alpha})}{\partial x_{-\alpha}}\delta x_{-\alpha}. \tag{7.42}$$

Since $E(x_\alpha) = E(x_{-\alpha})$, Equation 7.42 can be simplified as

$$\frac{\partial E(x_\alpha)}{\partial x_\alpha}\delta x_\alpha = \frac{\partial E(x_{-\alpha})}{\partial x_{-\alpha}}\delta x_{-\alpha}. \tag{7.43}$$

The x coordinate of a $p = 0$ point is computed by $x_0 = \frac{x_\alpha + x_{-\alpha}}{2\cos\alpha}$. When there are errors in x_α and $x_{-\alpha}$, $\tilde{x}_0 = \frac{(x_\alpha+\delta x_\alpha)+(x_{-\alpha}+\delta x_{-\alpha})}{2\cos\alpha} = x_0 + \frac{\delta x_\alpha + \delta x_{-\alpha}}{2\cos\alpha} = x_0 + \delta x_0$. Then the error in x_0 is $\delta x_0 = \frac{\delta x_\alpha + \delta x_{-\alpha}}{2\cos\alpha}$. The error is related to the surface curvature at the $p = 0$ point. This can be proved from the following derivations.

From Equation 7.40 and 7.41, $\delta D = \delta x_\alpha - \delta x_{-\alpha}$. Using substitution $\delta x_{-\alpha} = \delta x_\alpha - \delta D$, from Equation 7.43, we get

$$\frac{\partial E(x_\alpha)}{\partial x_\alpha}\delta x_\alpha = \frac{\partial E(x_{-\alpha})}{\partial x_{-\alpha}}(\delta x_\alpha - \delta D). \tag{7.44}$$

Solving for δx_α,

$$\delta x_\alpha = -\frac{\partial E(x_{-\alpha})}{\partial x_{-\alpha}}\delta D / \left(\frac{\partial E(x_\alpha)}{\partial x_\alpha} - \frac{\partial E(x_{-\alpha})}{\partial x_{-\alpha}}\right).$$

Similarly,

$$\delta x_{-\alpha} = -\frac{\partial E(x_\alpha)}{\partial x_\alpha}\delta D / \left(\frac{\partial E(x_\alpha)}{\partial x_\alpha} - \frac{\partial E(x_{-\alpha})}{\partial x_{-\alpha}}\right).$$

From the above two equations, we get,

$$\delta x_0 = -\delta D\left(\frac{\partial E(x_\alpha)}{\partial x_\alpha} + \frac{\partial E(x_{-\alpha})}{\partial x_{-\alpha}}\right) / \left(\frac{\partial E(x_\alpha)}{\partial x_\alpha} - \frac{\partial E(x_{-\alpha})}{\partial x_{-\alpha}}\right) / (2\cos\alpha). \tag{7.45}$$

We can prove that (see Appendix D)

$$\frac{\partial E(x_\alpha)}{\partial x_\alpha} + \frac{\partial E(x_{-\alpha})}{\partial x_{-\alpha}} = -2Q'(\cos e_\alpha) t_0 q_0; \tag{7.46}$$

$$\frac{\partial E(x_\alpha)}{\partial x_\alpha} - \frac{\partial E(x_{-\alpha})}{\partial x_{-\alpha}} = -2Q'(\cos e_\alpha) r_0 (1 + q_0^2) \tan \alpha. \tag{7.47}$$

Combining Equation 7.45–7.47 and using substitution $\delta D = 2\delta x_{\pi/2} \sin \alpha$, finally we have

$$\delta x_0 = -\delta x_{\pi/2} \frac{t_0 q_0}{r_0(1 + q_0^2)}.$$

The error in the x location of a wire frame caused by the error in the location of the corresponding occluding contour has nothing to do with the surface reflectance function. It is determined by the second partial derivatives of the surface. Note that when r_0 is small the error is large and vice-versa.

The wire-frame technique has been applied to a synthetic image sequence of the ellipsoid. The ellipsoid in synthetic images is rendered with the reflectance function extracted from the real images and illuminated and viewed under the same conditions as the ellipsoid in the real images. The depth on a $p = 0$ curve is determined from the corresponding occluding contour extracted from the synthetic images. Table 7.11 shows the errors in the computed x locations of the $p = 0$ curves in the synthetic $image_{2k\pi/9}$, $(0 \leq k < 9)$, of the ellipsoid. The errors are mainly caused by the quantization effects of image brightness and image space. The subpixel accuracy on the x location proves that the wireframe technique is effective and accurate.

7.5.5 Rotation angle and surface curvature

Even if there is no disparity error, the quantity r_0 affects the accuracy of the x location on the wire frame. Let x_α and $x_{-\alpha}$ be the x locations of the images of a $p = 0$ point in $image_\alpha$ and $image_{-\alpha}$ respectively, and let D be the disparity $x_\alpha - x_{-\alpha}$. If there a

image	0	$4\pi/18$	$8\pi/18$	$12\pi/18$	$16\pi/18$	$20\pi/18$	$24\pi/18$	$28\pi/18$	$32\pi/18$
r.m.s. error	0.29	0.21	0.13	0.26	0.09	0.30	0.04	0.26	0.14
average error	0.06	0.07	−0.05	−0.08	0.02	0.10	0.01	−0.09	−0.04

Table 7.11: The first row is the subscripts for the images. The second and the third row are, respectively, r.m.s. and average errors in the x locations of the $p = 0$ curves in the corresponding synthetic images of the ellipsoid.

small perturbation δx on x_α, the x coordinate of the image point, which is computed by disparity D, in $image_{-\alpha}$ will be $x_{-\alpha} + \delta x$. Using linear approximation gives

$$E(x_\alpha + \delta x) - E(x_{-\alpha} + \delta x) = \left(E(x_\alpha) + \frac{\partial E(x_\alpha)}{\partial x_\alpha}\delta x \right) - \left(E(x_{-\alpha}) + \frac{\partial E(x_{-\alpha})}{\partial x_{-\alpha}}\delta x \right).$$

Since $E(x_\alpha) = E(x_{-\alpha})$,

$$E(x_\alpha + \delta x) - E(x_{-\alpha} + \delta x) = \left(\frac{\partial E(x_\alpha)}{\partial x_\alpha} - \frac{\partial E(x_{-\alpha})}{\partial x_{-\alpha}} \right) \delta x. \tag{7.48}$$

The difference of the brightness gradients in the above equation is related to the quantity r_0 as indicated in Equation 7.47. The larger the quantity r_0, the larger the difference. From Equation 7.48, we see that the larger the difference, the more sensitive the equ-brightness constraint to the perturbation δx and therefore the more accurate the locations of the two image points. Comparing the wire frame on the body of the cup with that on the handle (see Fig. 5.36), the wire frame on the handle is more smooth because the quantity r_0 on the handle is much larger.

The difference of the brightness gradients in Equation 7.47 also depends on the rotation angle α. To make the x location on the wire frame accurate, the rotation angle α should be large. However, a large rotation angle tends to cause occlusion on the curve in $image_\alpha$ and $image_{-\alpha}$. In our experiments, the rotation angle α used is $\pi/9$.

7.5.6 Effects of image noise

If we assume uniform distributed image noise, the effects of image noise will also depend on the difference of the brightness gradients of the two image points. In the presence of image noise, there will be errors in the x coordinates of the two image points computed by linear interpolation described in Section 5.3. In the following calculation, the definitions of all the variables are the same as those in 5.3 except the new variables and symbols. Let image noise be $n(\,)$, and the x location error of the two image points, which is caused by image noise, be δx. Similar to Equation 5.25 and 5.26, we have

$$E(x_\alpha + \delta x) = E(\dot{x}_\alpha) + n(\dot{x}_\alpha) + (\Delta E(x_\alpha) + \Delta n(x_\alpha))(x_\alpha + \delta x - \dot{x}_\alpha) \tag{7.49}$$

with $\Delta n(x_\alpha) = n(\ddot{x}_\alpha) - n(\dot{x}_\alpha)$, and

$$E(x_{-\alpha} + \delta x) = E(\dot{x}_{-\alpha}) + n(\dot{x}_{-\alpha}) + (\Delta E(x_{-\alpha}) + \Delta n(x_{-\alpha}))(x_{-\alpha} + \delta x - \dot{x}_{-\alpha}) \tag{7.50}$$

with $\Delta n(x_{-\alpha}) = n(\ddot{x}_{-\alpha}) - n(\dot{x}_{-\alpha})$. Since the two image points must satisfy the equi-brightness constraint,

$$E(x_\alpha + \delta x) = E(x_{-\alpha} + \delta x).$$

Substituting the two sides with Equation 7.49 and 7.50 and simplifying the equation by using Equation 5.27, we have

$$n(\dot{x}_\alpha) + \Delta E(x_\alpha)\delta x + \Delta n(x_\alpha)(x_\alpha + \delta x - \dot{x}_\alpha) = n(\dot{x}_{-\alpha}) + \Delta E(x_{-\alpha})\delta x + \Delta n(x_{-\alpha})(x_{-\alpha} + \delta x - \dot{x}_{-\alpha})$$

Solving for δx,

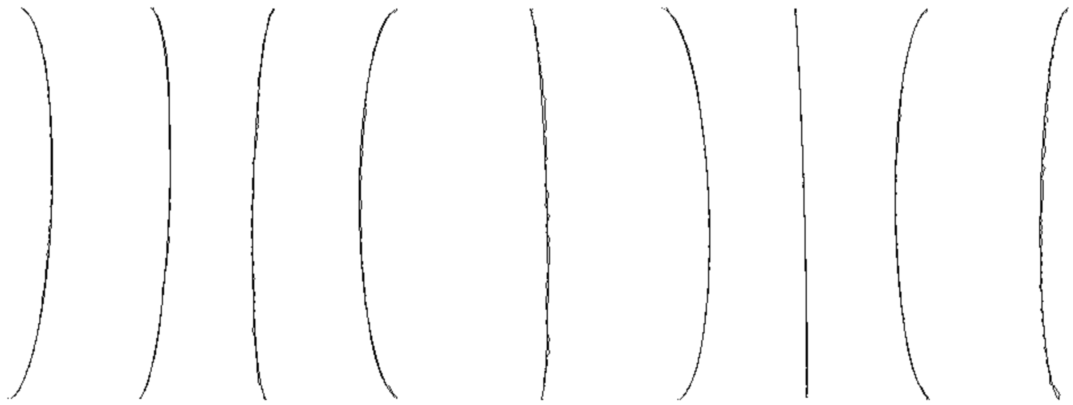
$$\delta x = \frac{n(x_{-\alpha}) - n(x_\alpha) + \Delta n(x_{-\alpha})(x_{-\alpha} - \dot{x}_{-\alpha}) - \Delta n(x_\alpha)(x_\alpha - \dot{x}_\alpha)}{\Delta E(x_\alpha) - \Delta E(x_{-\alpha}) + \Delta n(x_\alpha) - \Delta n(x_{-\alpha})}, \tag{7.51}$$

where $\Delta E(x_\alpha) - \Delta E(x_{-\alpha})$ can be considered as an approximation for $\frac{\partial E(x_\alpha)}{\partial x_\alpha} - \frac{\partial E(x_{-\alpha})}{\partial x_{-\alpha}}$. The above equation shows that the larger the difference between the brightness gradients of the two image points, the smaller the error in their locations affected by image noise and vice-versa. As the difference is associated with r_0, for the cup, we know that the difference of the image gradient for points on the handle is larger than that on the body. So the x location of the curves of constant p on the handle of the cup are more resistant to image noise than those on the body. This gives another explanation of why the wire frame on the handle is more smooth.

Figure 7.61 shows the extracted $p = 0$ curves in the real images, $image_{2k\pi/9}$, $(0 \le k < 9)$ of the ellipsoid. The extracted $p = 0$ curves are overlaid with the real $p = 0$ curves. Table 7.12 shows the errors in the x locations of the $p = 0$ curves in the real images, $image_{2k\pi/9}$, $(0 \le k < 9)$, of the ellipsoid. Image noise adds one more error source in the x location of the $p = 0$ curves computed from the real images. Comparing Table 7.12 with Table 7.11, we can see how much error is caused by image noise. From Fig. 7.61 and Table 7.12, we can see the errors in the x location of the $p = 0$ curves in $image_{16\pi/18}$ and $image_{32\pi/18}$ are larger than those in the other images. This is because the flatter side of the ellipsoid is facing the viewing direction in these images and the second partial derivative r_0 on the $p = 0$ curves is smaller. The smaller r_0 leads to the larger errors in the x location of the $p = 0$ curves. This result is consistent with the our analysis in the previous subsection.

image	0	$4\pi/18$	$8\pi/18$	$12\pi/18$	$16\pi/18$	$20\pi/18$	$24\pi/18$	$28\pi/18$	$32\pi/18$
r.m.s. error	0.63	0.64	0.78	0.91	1.24	0.44	0.33	0.57	1.14
average error	−0.34	0.32	−0.21	0.30	0.61	0.17	−0.22	0.37	0.54

Table 7.12: The first row is the subscripts for the real images of the ellipsoid. The second and the third row are, respectively, r.m.s. and average errors in the x location of the $p = 0$ curves in the corresponding image of the ellipsoid.



0 $4\pi/18$ $8\pi/18$ $12\pi/18$ $16\pi/18$ $20\pi/18$ $24\pi/18$ $28\pi/18$ $32\pi/18$

Figure 7.61: The extracted $p = 0$ curves and the real $p = 0$ curves of the ellipsoid. The label below each curve denotes the image in which the corresponding $p = 0$ curve is located.

7.6 Summary

The detailed analysis and evaluation on the photogeometric technique and wire-frame technique show that the two techniques are robust and efficient.

To analyze the surface recovery error caused by different factors, a manufactured ellipsoid with ground truth is used. Synthetic images are generated from the geometric model of the ellipsoid. The two techniques are applied to both real images and synthetic images of the ellipsoid. The surface recovery errors are calculated. The errors are listed in tables and displayed with a set of error maps.

The surface reflectance function can be made more accurate by using multiple sample points. However, a pseudo surface reflectance can be used to recover the p component and the depth.

The effects of image noise on surface recovery are analyzed by using the synthetic images of the ellipsoid at different noise levels. The photogeometric technique is applied to the synthetic images. Comparing the error in the surface recovered from the synthetic images with the error in the surface recovered from the real images, we find that the error caused by image noise only takes a small part of the total error. Mathematical analysis also shows that the photogeometric technique is resistant to image noise.

The effects of non-collinear light source are investigated for Lambertian, specular and hybrid surfaces of known surface orientation. When the illuminant direction of a non-collinear light source is fixed, the error in surface orientation recovery can be calculated. For a surface point on a Lambertian or specular surface, the error in the computed surface orientation is a function of its real surface orientation. For a surface point on a hybrid surface, the maximum error in the computed surface orientation is within a range defined by its real surface orientation.

The surface recovery error caused by non-uniform surface reflectance is also analyzed under the assumption of Lambertian, specular and hybrid surfaces. For a Lambertian surface, the non-uniform surface reflectance does not introduce error in computing the p component. Thus the depths calculated by integrating the p component are not affected by the non-uniform reflectance either. For specular and hybrid surfaces, the effects of the non-uniform reflectance are not as serious as we thought it might because the effects are partially compensated in computing the p component. However, the non-uniform reflectance does cause an error in computing the q component at a surface point. The error is proportional to the difference of the reflectance factor of the surface point and the reflectance factor of the sample surface point.

Mathematical analysis shows that the accumulated error in the depth calculated by integrating the p component is related to the surface curvature and integration direction. The accumulated error increases in one direction but compensates in the other. Since the direction that increases the error can be detected, the error can be reduced by choose the right integration direction. Experimental results verify the mathematical analysis.

To reduce surface recovery error, the image should be imaged as large as possible and the object center should be located in the rotation center. The rotation axis of the object can be located with subpixel accuracy and this accuracy is sufficient for surface recovery. The rotation angle between the two images used for surface recovery should neither too large nor too small. Empirical tests show that the angle should be about $\pi/18$.

When the precise shape of the object is not known, the occluding contours in the images can be used to judge the recovered surface. The comparison between the occluding contours in the original images and the occluding contours projected from the recovered surface gives the error between the real surface and recovered surface. As the error in locating the occluding contours in images can be determined, the range of the depth recovery error can be estimated from the difference between the two sets of contours.

The analysis on the effects of the different factors shows that, in surface recovery, the error in the q component is larger than the error in the p component. The error near the boundary of the object is larger than the error in the interior of the object. The surface recovery results from both real images and synthetic images confirm the analysis.

For the wire-frame technique, we have shown that a coplanar light source is sufficient and the object surface can be piecewise uniform. The error in the depth on the wire frame is determined by the error in the location of the corresponding occluding contours. The error in the x location on a wire frame is also affected by the location error in its corresponding occluding contour as well as the second partial derivatives. The effects of image noise on the x location of a curve depend on the brightness gradient difference. The brightness gradient difference depends on the surface curvature and the rotation angle between the two images used to extract the curve.

In the next chapter, we will show how the recovered surface can be used as surface model and how the surface reflectance function extracted from real images can be used for realistic rendering in computer graphics to achieve virtual reality.

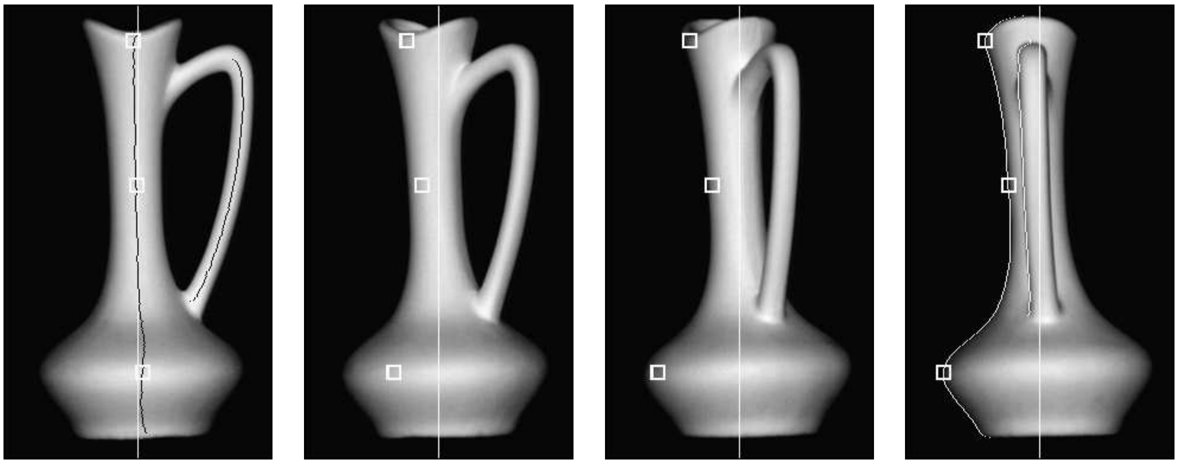
Chapter 8

Extracting the Surface and Shading Function from Real Images

8.1 Introduction

Computer vision and computer graphics are two complementary fields of computer science. Computer graphics deals with direct problems, that is, generating images from predefined shading models and geometric models, while computer vision studies inverse problems, that is, extracting reflectance and 3D geometric information from real images. Because of the nature of the two problems, graphics and computer vision have been on two separate tracks for many years. Since both of them are trying to achieve realistic descriptions of the physical world, they share some common aspects and their interaction is inevitable, especially in reflectance modeling and surface modeling. Both reflectance models [75, 51] in computer vision and shading models [16, 25] in computer graphics attempt to achieve physics-based relistic description of light reflection. So the research of these models can benefit each other. Both computer graphics and computer vision have to deal with surface modeling. The notion of a generalized cylinder was first introduced by Binford [4] in computer vision long ago and has been widely used in computer graphics. Terzopoulos's physics-based models, such as finite element models [47], adaptive meshes [68] and deformable superquadrics [67], can be directly applied to animation in computer graphics. The oriented particle surface model developed by Szeliski *et al.* [63, 62] is useful in both computer vision and computer graphics.

We attempt to integrate computer vision with computer graphics in both reflectance



a. $image_0$ b. $image_{\pi/6}$ c. $image_{\pi/3}$ d. $image_{\pi/2}$

Figure 8.62: Tracking the sample points over an image sequence. The center of a square box is the image of a sample point. The vertical line is the rotation axis. The white curves in (d) are the detected occluding contours. The black curves in (a) are $p = 0$ curves.

modeling and surface modeling for more realistic visualization. The surface reflectance function is extracted from an image sequence of a rotating object illuminated under a collinear light source. The extracted reflectance function is used to recover surface shape of the object. A shading function is estimated from the extracted surface reflectance function by an optimization method. Then the estimated shading function is used for rendering the recovered object surface under the general illuminant and viewing conditions. Experiments with objects with diffuse surfaces and specular surfaces show that the technique is feasible and promising.

8.2 Surface Reflectance Function

The surface reflectance function of an object is extracted from the image sequence taken during the rotation of the object. The imaging conditions and the method for extracting surface reflectance function are the same as those described in Sections 4.1

and 4.2. Nineteen images of a rotating vase are taken during $\pi/2$ rotation with $\pi/36$ rotation between two successive images. Figure 8.62 shows four images in the image sequence. Three singular points in $image_0$ are detected and their depths are computed from the corresponding contour points in $image_{\pi/2}$. The 3D surface points corresponding to the singular points are taken as sample points for extracting the surface reflectance function. Their images and brightness values are tracked over every image in the image sequence. The center of a small square box in image (a)-(d) of Fig. 8.62 marks the location of a sample point. The surface reflectance function is built from the correspondence from the brightness values of the sample points in an image to the emergent angles of the sample points. The average of the brightness values of the three sample points are used. The continuous line in Fig. 8.63 shows the surface reflectance function of the vase.

8.3 Surface Recovery

The surface of the vase is recovered by using the photogeometric technique. The depth and surface orientation is recovered at every pixel in $image_0$. To recover the surface of the vase, the 3D locations of the two $p = 0$ curves (see black lines in Fig 8.62(a)) are recovered first. Starting from the $p = 0$ curves, the computation on the depth and surface orientation is then extended in the x direction until the background is reached. This process may not cover some areas, such as the top of the body part. The computation is extended in the y direction to reach those areas. Figure 8.64 shows the surface depth and orientation of the recovered vase.

8.4 Estimating the Shading Function

In computer vision and computer graphics, the terminology used to describe the relation between image brightness and surface orientation is different. In computer vision, the

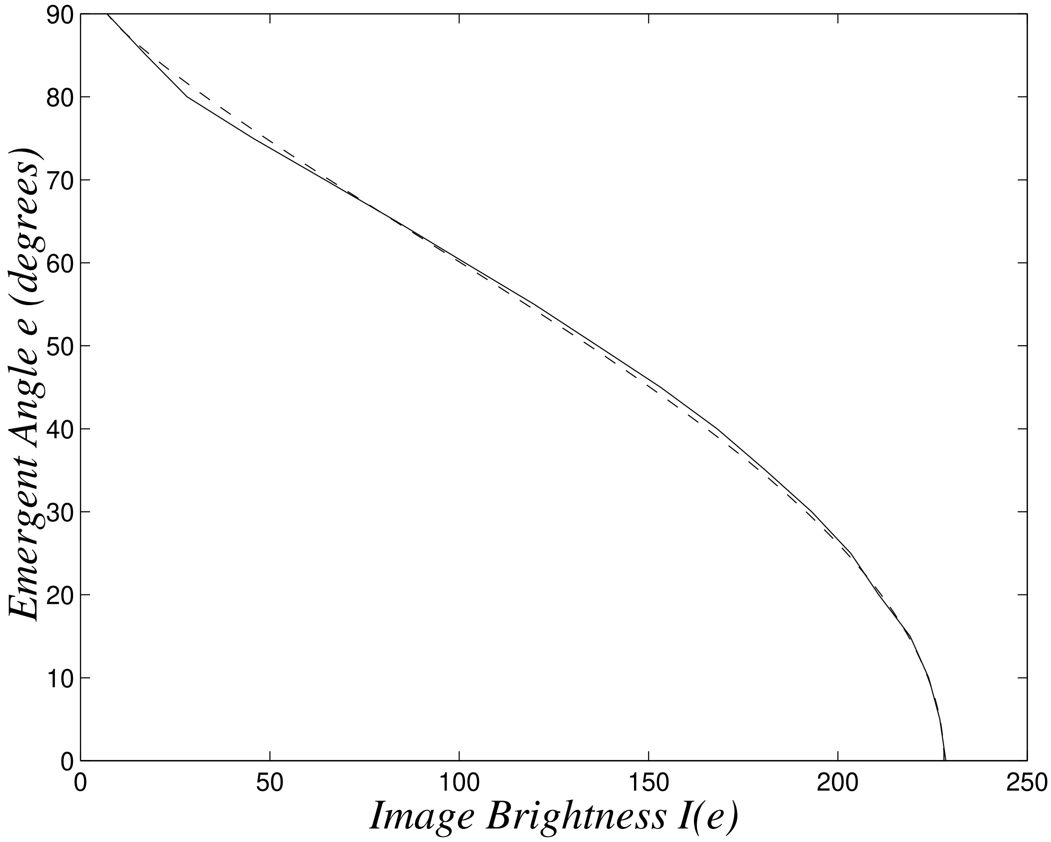
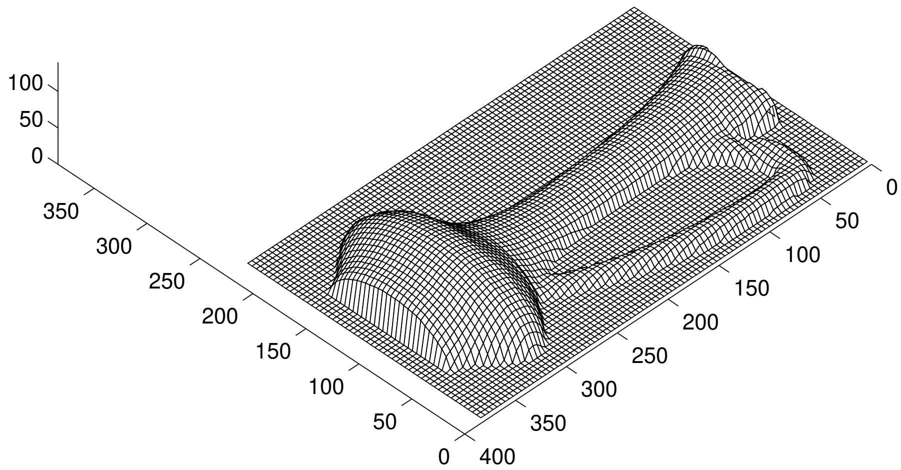
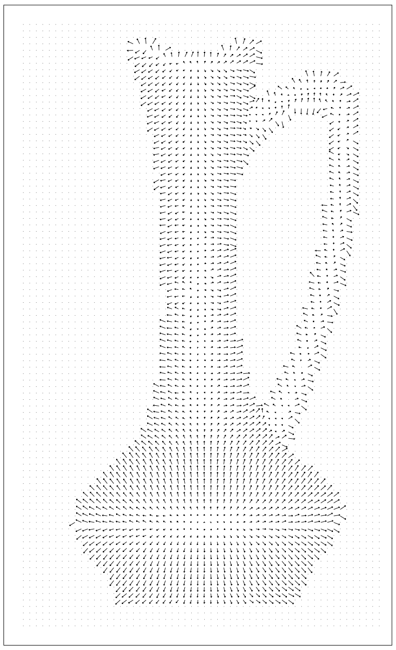


Figure 8.63: The solid line shows the surface reflectance function extracted from images of the vase. The dashed line shows the estimated shading function of Phong-Blinn model.

relation is called the surface reflectance function. In computer graphics, the relation is called the shading function. The shading function is used to render object models and generate synthetic images. One of the objectives in computer graphics is to generate virtual images which look like real images. In computer graphics, a shading function usually has a set of parameters and these parameters are determined heuristically. Some work has been done in trying to get physics-based shading function by analytical modeling [16, 25] or by using measurement devices [73]. We attempt to get the shading function from real images. The surface reflectance function extracted from real images



a. The depth plot



b. The surface orientation plot

Figure 8.64: The surface depth and orientation of the recovered vase

gives a description of how bright a real surface point in the real physical world will be when a surface is illuminated and viewed under certain conditions. However, the surface reflectance function extracted from images can only give the relation between brightness value and surface orientation under the fixed illuminating and viewing directions under which these images are taken. It can't be directly used in computer graphics to render the objects under arbitrary illuminating and viewing directions. The surface reflectance function can be considered as a shading function under a collinear light source and then the parameters for the shading function can be estimated from the surface reflectance function. Once the shading function is estimated, it can be used to render object surfaces under any illuminating and viewing directions.

The surface reflectance function we extracted is a function under a collinear light source and it has the form

$$E = Q(\cos e)$$

with e as the emergent angle at a surface point. One of the most widely used shading models in computer graphics is the Phong-Blinn shading model [6]. It has the form

$$I = I_a + I_d(N \cdot L) + I_s(N \cdot H)^n, \tag{8.52}$$

where N is the surface normal, L is the illuminant direction vector, and H is the vector halfway between the viewing direction and the illuminant direction. In the function, I_a, I_d and I_s are coefficients for the ambient component, diffuse component, and specular component respectively. The exponent n in the function represents the roughness of the surface. In computer graphics rendering, these parameters are determined by *ad hoc* methods. We consider the surface reflectance function extracted from the real images as the shading function under a collinear light source. In this case,

$$N \cdot L = \cos e \quad \& \quad N \cdot H = \cos e$$

and

$$I = I_a + I_d \cos e + I_s \cos^n e.$$

Now our task is to find a set of values for I_a, I_d, I_s and n which minimizes the difference of the two functions under a collinear light source. The value for I_a can be determined from the background brightness in the real images. When $\cos e = 1$, the surface reflectance function takes the maximum brightness value and so should the shading function. Let the maximum brightness value be I_{max}; then

$$I_{max} = I_a + I_d + I_s \quad \& \quad I_s = I_{max} - I_a - I_d.$$

The shading function under a collinear light source can be rewritten as

$$I = I_a + I_d \cos e + (I_{max} - I_a - I_d) \cos^n e.$$

Since I_a and I_{max} are known, only I_d and n need to be determined.

The values for I_d and I_{max} are determined by minimizing the least-squares error of the two functions which can be expressed as

$$f(\cos e, I_d, n) = \sum [Q(\cos e) - (I_a + I_d \cos e + (I_{max} - I_a - I_d) \cos^n e)]^2.$$

Starting from an initial guess for n, I_d can be calculated by solving $\partial f(\cos e, I_d, n)/\partial I_d$. By using one dimensional search with first derivatives, a new value of n which minimizes $f(\cos e, I_d, n)$ can be found. By using the new value of n, a new value for I_d can be calculated. Iterating the above process, the values of I_d and n which minimize the least-squares error can be determined. The parameters for the shading function estimated from the surface reflectance function of the vase are $I_a = 7$, $I_d = 80$, $I_s = 141$ and $n = 1.4$. Thus the estimated shading function is

$$I = 7 + 80(N \cdot L) + 141(N \cdot H)^{1.4}$$

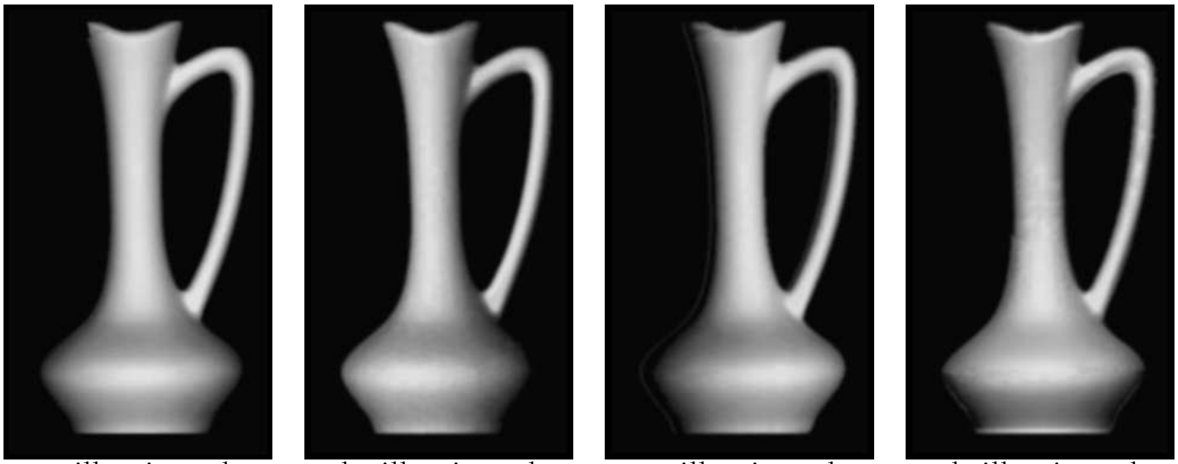
and shown as a dashed line in Fig. 8.63. Though the surface of the vase is close to a diffuse surface there is still a weak specular surface reflectance component in the estimated shading function.

8.5 Rendering with the Estimated Shading Function

Using the estimated shading function and the surface orientation data recovered from the image sequence, we generate synthetic images of the recovered surface by rendering the surface under different illuminant directions but fixed viewing direction. Figure 8.65 shows two groups of images. The images in the upper row are synthetic images generated under different illuminant directions. The images in the lower row are the corresponding real images. The illuminant direction and viewing direction for the synthetic image and the real image in the same column are similar. The location and orientation of the object in the synthetic image and the corresponding real image are similar. The synthetic images are generated by a simple shading algorithm which only considers local shading, i.e., the shading directly caused by the light source illumination. To simulate the blurring effect of the camera, the synthetic images are filtered with a Gaussian filter of $\sigma = 1$. Comparing the synthetic images with the real images in the same column, we see that the differences between the synthetic images and real images are very small. The small differences show that the shading function is a good approximation of the real surface reflectance function and it gives realistic rendering.

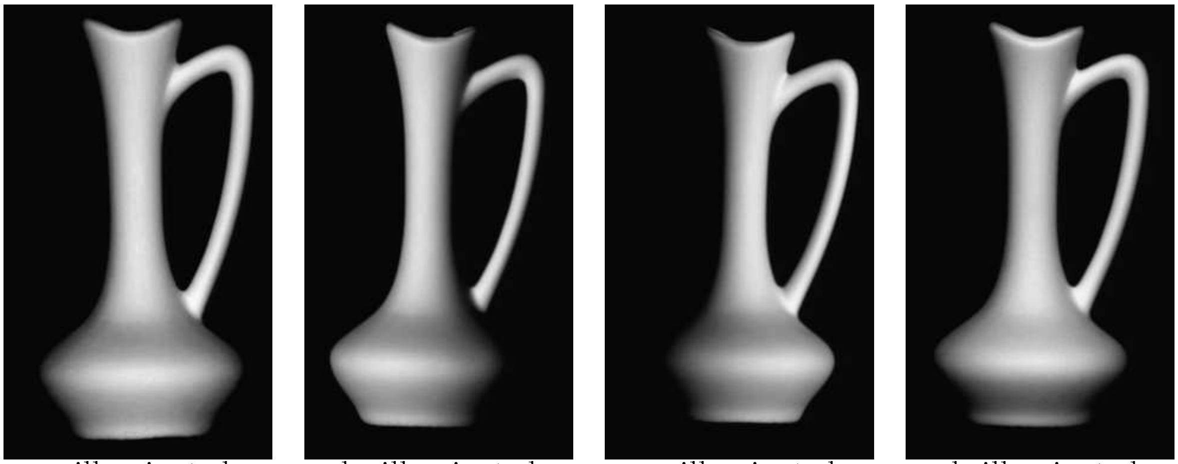
8.6 Experiment with a Specular Surface

We performed another experiment with a porcelain cup and tried to estimated a shading function of the Phong-Blinn model from the surface reflectance function extracted from the images of the cup. The surface of the cup is quite specular. Some images from



a. illuminated at $(0, 0, 1)$ | b. illuminated at $(-\tan\frac{\pi}{6}, 0, 1)$ | c. illuminated at $(\tan\frac{\pi}{6}, 0, 1)$ | d. illuminated at $(0, \tan\frac{\pi}{10}, 1)$

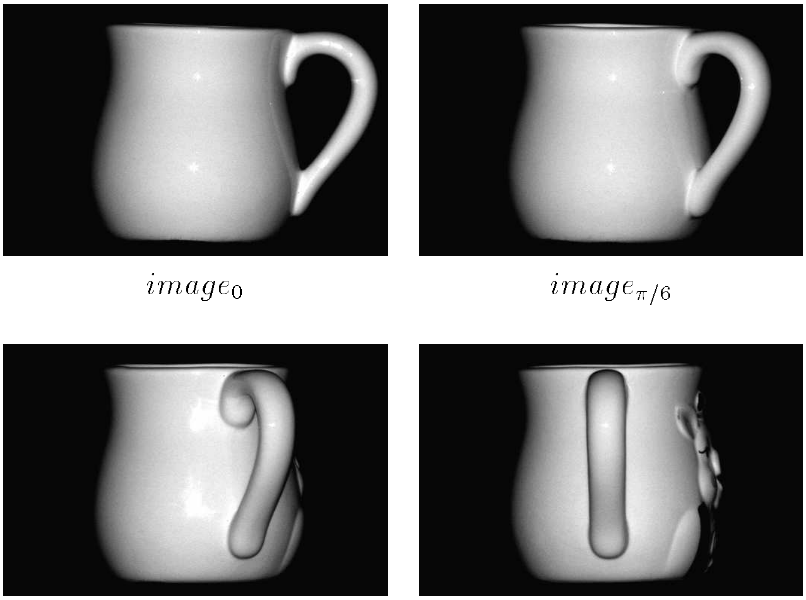
Synthetic images of the recovered surface



a. illuminated at $(0, 0, 1)$ | b. illuminated at $(-\tan\frac{\pi}{6}, 0, 1)$ | c. illuminated at $(\tan\frac{\pi}{6}, 0, 1)$ | d. illuminated at $(0, \tan\frac{\pi}{10}, 1)$

Real images of the original object

Figure 8.65: The synthetic images of (a)-(d) are rendered with the shading function estimated from the surface reflectance function extracted from the real images. The illuminant and viewing conditions for generating a synthetic image are similar to those used in taking the corresponding real image in the same column.



$image_0$ $image_{\pi/6}$

$image_{\pi/3}$ $image_{\pi 2}$

Figure 8.66: Four images of an image sequence of a rotating cup.

the image sequence of the rotating cup are shown in Fig. 8.66. The reflectance function extracted from the images is shown in Fig. 8.67 as a solid line. The depth plot and the orientation map of the recovered surface are shown in Fig. 5.38 and Fig. 5.39. Since the surface of the cup is highly specular, the singular points in the images have saturated brightness values. The value for I_{max} should be larger than the saturated brightness value. An initial larger value is used as a guess for I_{max}. To estimate a shading function from the surface reflectance function of the cup, the least-squares error of the two functions under a collinear light source is minimized. The initial value of I_{max} is set to 297 and the same minimization method (see Section 8.4) is used to get the shading function

$$I = 7 + 220(N \cdot L) + 70(N \cdot H)^{2000}.$$

The function is shown in Fig. 8.67 as a dashed line.

In fact within certain region, the least-squares error is not sensitive to the change of

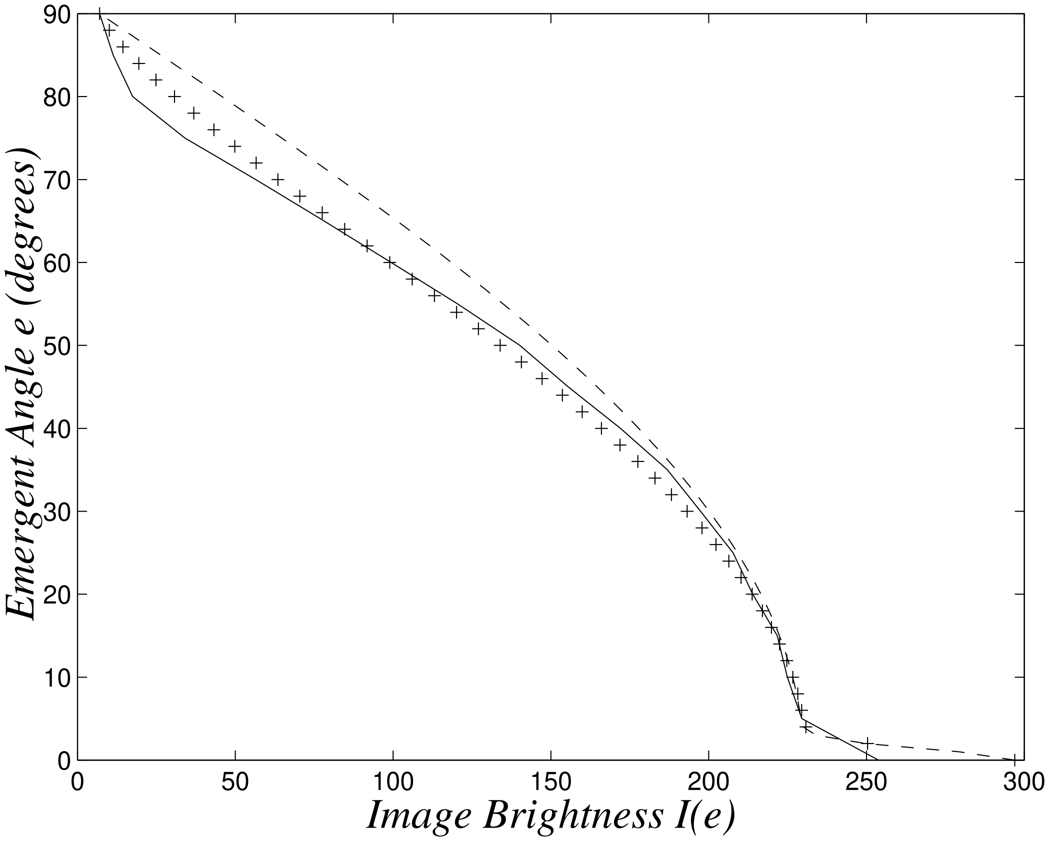


Figure 8.67: The solid line shows the surface reflectance function extracted from images. The dashed line shows the estimated shading function of the Phong-Blinn model. The plus line shows a better shading function of different model which is closer to the surface reflectance function.

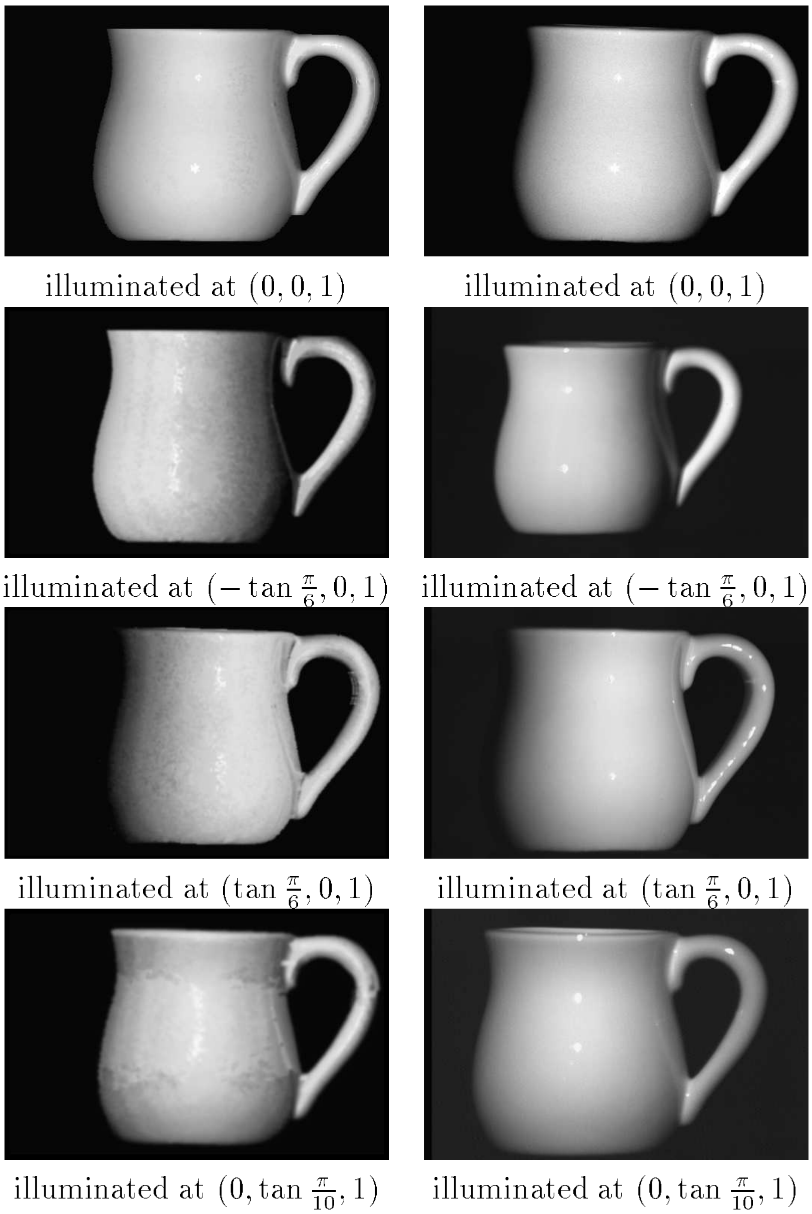


Figure 8.68: The left column is the synthetic images of the recovered surface and the right column is the real images of the original object. The synthetic images are rendered with the estimated shading function. The illuminant and viewing conditions for generating a synthetic image are similar to those used for taking the corresponding real image in the same row.

I_{max} and n. When n is large, say $n > 500$, the specular component is a spike-like function and its effects are quite localized. The major part of the least-squares error is caused by the diffuse component in the shading function. The large least-squares error between the two functions reveals that the Phong-Blinn model is not the right shading model for the surface of the cup.

There are two groups of images in Fig. 8.68. Each image on the left side is a synthetic image and each image on the right side is the real image corresponding to the left side synthetic image. The synthetic images are generated by using the recovered surface orientation data of the cup and the estimated shading function. The location and orientation of the object in a synthetic image and the corresponding real image are similar. Each image in the synthetic images is rendered under the similar illuminating and viewing directions to those under which the corresponding real image is taken. Visual comparison can be made between a synthetic image and its corresponding real image. The specular spots in the synthetic images show that the specular spike is modeled reasonably well. The differences between the synthetic images and real images are visible. The brightness in the synthetic images is not as smooth as that in the real images. The specular spots in the synthetic images are different from those in the real images. The differences are mainly caused by error in the recovered surface orientation. The orientation data, especially, the q component is not smooth.

8.7 Summary

The experimental results indicates that our technique works. However, the work described here is preliminary. The depth and surface orientation are recovered only from one view and they are not continuous at some points along the y direction. The discontinuity in surface orientation is visible from the synthetic images of the recovered surface. We have tried to obtain smooth surfaces by using surface fitting techniques without achieving satisfactory results. The two surface fitting techniques [26, 18] we used require that the distance between adjacent data points be uniform. But the data obtained by our surface recovery technique does not meet the requirement. Fitting the depth and surface orientation data with smooth surfaces is not trivial so we take this as a future research direction.

The reflectance function of some surfaces cannot be model properly by the Phong-Blinn shading model. The four component model in computer vision [48] gives a better description of the surface reflectance of the porcelain cup. The model contains four components: ambient, diffuse, specular and specular spike. A shading function which fits the reflectance function of the cup better is

$$I = 7 + 10(N \cdot L) + 210(N \cdot H)^{1.3} + 70(N \cdot H)^{2000}.$$

It contains a weak specular component and a strong specular spike. The function is shown in Fig. 8.67 as a plus line.

The integration of computer vision and computer graphics is inevitable. The positive initial results demonstrate the potential of computer vision techniques to enhance realistic modeling in computer graphics.

Chapter 9

Conclusion

This thesis explores geometric and photometric constraints in an image sequence of a rotating object illuminated under a collinear light source and develops techniques for surface reflectance extraction and surface recovery by integrating geometric and photometric constraints. The first section of this chapter summarizes the major contributions of this thesis. The second section discusses the strength and weakness of the proposed techniques. The last section talks about the extensions and future research directions.

9.1 Contributions

The major contributions of this thesis are the following:

- We have presented a new method which directly extracts surface reflectance from images.

  All surface recovery techniques using brightness or shading require the surface reflectance function of the object to be recovered. The surface reflectance function can be obtained from a calibration process, or extracted from images. The calibration process requires that the shape of the calibrating object is known and the surface reflectance function of the calibrating object is the same as that of the object to be recovered. The existing methods for extracting surface reflectance from images assume that the reflectance function has a certain form and fit the form with the image data. However, the method described in this thesis directly extracts the surface reflectance function from images of the object to be recovered.

It assumes only that the surface reflectance function is isotropic and monotonic. This assumption is true for most surfaces. Therefore the method is simple, reliable, and accurate.

- We have developed an effective and practical imaging process for extracting surface reflectance and shape of a rotating object.

  In a controllable experimental environment, we can arrange the illuminant of the light source and the viewing condition of the camera, as well as the motion of the object. A proper arrangement will make the imaging process simple and the surface recovery task easier. By using a collinear light source and an approximation to orthographic image projection, the surface reflectance function is simplified and can be extracted directly from images of an rotating object. Both geometric and photometric constraints as well as their relations in the images also simplified so that geometric and photometric constraints can be integrated in a natural way for surface recovery.

- We have demonstrated a technique which recovers surface depth and orientation simultaneously.

  There are several kinds of image features available for surface recovery when an object is rotated in front of a camera. Most techniques just use one of them. Techniques using contour cannot recover concave regions. Techniques using flow or stereo require surface marks or texture on the object surface. Techniques using shading are vulnerable to image noise. Integrating these image feature will yield more powerful and robust techniques. So far no other work on integrating several image features from the image sequence of a rotating object has been reported. The photogeometric technique described in this thesis integrates contour and brightness in images in a natural way and recovers the surface depth and orientation simultaneously at each pixel in an image. The integration of other image features, such as stereo and flow, is a straight forward extension of the techniques proposed in this thesis.

- We have developed a technique which extracts a 3D wire frame on an object surface from its images without knowing the surface reflectance function.

  The wire-frame technique is the result of the further integration of the geometric and photometric constraints. The technique does not need the surface reflectance of the object to be recovered. It can work on an object of piecewise uniform surface. The illuminant direction does not need to be collinear but to be coplanar with the rotation axis and viewing direction.

- We have designed a new method for extracting the shading function from real images for realistic graphics rendering.

  The shading function used in graphics is often based on heuristic approach. This thesis proposes a method for extracting the shading function from real images for realistic surface rendering. The work is preliminary but it is feasible and promising.

9.2 Discussion

The two techniques are feasible and effective. The computation is simple and efficient. The depth data obtained is reliable and resistant to image noise. However, in computing the surface orientation, the q component is less accurate than the p component. The q component is more sensitive to image noise, non-uniform reflectance of the surface, and non-collinear illumination as well as other error sources. The p component is mainly determined by the ratio of $Q^{-1}(E_\alpha)$ to $Q^{-1}(E_0)$, while the q component is determined by the ratio and a term of $Q^{-1}(E_0)$. Whenever an error source causes changes in the brightness value E_0 and E_α, it will cause a larger error in computing the q component.

The wire-frame technique cannot extract a $p = 0$ curve in a concave area since the $p = 0$ curve in a concave area cannot be seen as an occluding contour in images. The wire-frame technique also requires a $p = 0$ curve be seen from the two images taken as the object is rotated by two symmetric angles. Therefore the wire frame extracted from the images may not be complete. To get a complete wire frame, the photogeometric technique can be used after the surface reflectance function has been extracted.

The imaging geometry is approximated by orthographic projection. The techniques will fail under perspective projection. Under perspective projection, the viewing direction changes as the viewing angle changes over the surface so that the phase angle g changes. As a result, even under the illumination of a collinear light source, the emergent angle e will not equal to the incident angle i except in the direction of the optical axis. The image brightness value is no longer the function only of the surface orientation. Therefore all the photometric constraints used in the techniques for surface recovery no longer exist. In addition, some geometric constraints we used also disappear under the perspective projection.

The light source for the photogeometric and the wire-frame techniques is not necessary to be collinear with the viewing direction, that is, the optical axis of the light source does not have to be aligned with the optical axis of the camera. *A collimated light source* (where the illuminant direction of the light source is parallel to the viewing direction of the camera) is sufficient.

The photogeometric technique cannot recover depth and surface orientation accurately in the area of high curvature along the x direction. Since the technique uses the first order Taylor series approximation to compute the depth at the next image point, the error caused by ignoring the terms above the first order will be larger in the area of higher curvature.

Interreflection still poses a problem in surface recovery. To overcome the interreflection problem, new techniques involving sophisticated computational methods must be developed.

The potential applications of our two techniques are computer graphics modeling, fast prototyping, surface inspection, object recognition and computer aided design.

9.3 Future Research Directions

The photogeometric technique can be improved to achieve possible real-time speed. The only time consuming operation is Gaussian filtering which can be sped up by using specialized hardware. The surface recovery in the x direction can be parallelized. The computation of surface orientation can be implemented much faster by using a lookup table. The wire-frame technique requires much less computation and it is much easier to implement in real-time.

A direct extension of our techniques is surface recovery of an object with arbitrary motion and rotation. As long as the motion and rotation of the object can be controlled or estimated, the change of 3D location and surface orientation of surface points in 3D space can be determined. It is possible to determine the 3D location and surface orientation of some particular surface points and then extract surface reflectance function from these surface points by tracking their surface orientation and brightness values. Once the surface reflectance function is obtained, the surface can be recovered from two images of the object. The basic principles described in this thesis can be extended and applied to surface recovery under arbitrary motion and rotation.

It is possible to extract the surface reflectance function and recover the surface by using only four images. Three images, $image_{\pi/2}$, $image_{\alpha}$ and $image_{-\alpha}$ can be used to extract a $p = 0$ curve. The brightness value and surface orientation at every point on the

$p = 0$ curve can be used to build the surface reflectance function. The two images, $image_0$ and $image_\alpha$ can be used for surface recovery by using the photogeometric technique.

The surface recovered from images is represented as surface depth and orientation data at image pixels. One extension of our work is to transform the depth and orientation data recovered from different viewing directions into a concise description of a 3D surface model. There are several algorithms which fit a set of 3D points with smooth surface model. In our case, we obtain both 3D location and surface orientation at each point so that fitting the points with smooth surfaces should be relatively easy.

The work described in Chapter 8 can be improved by using the appropriate shading models. To get a more accurate estimation of the parameters of a shading function, more images illuminanted and viewed from different directions can be used.

Interreflection poses a common problem for all techniques which use image brightness or shading information. Although our techniques are relatively resistant to interreflection, they still suffer from it. The interreflection problem is complicated but still tractable in some cases. The surface reflectance function extracted from images and the surface recovered by using the photogeometric technique make a good start in solving the interreflection problem.

Bibliography

[1] John Amanatides, John Ruchanan, and *et al.* Optik user's manual, version 2.6. 1993.

[2] P. Beckmann and A. Spizzichino. *The Scattering of Electromagnetic Waves from Rough Surfaces.* Macmillan, New York, 1963.

[3] M. Bichsel and A. P. Pentland. A simple algorithm for shape from shading. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1992*, pages 459–465, 1992.

[4] T. O. Binford. Visual perception by computer. In *IEEE Conference on Systems and Control*, Miami, 1971.

[5] A. Blake. Surface descriptions from stereo and shading. *Image and Vision Computing*, 3:183–191, 1985.

[6] James F. Blinn. Models of light reflection for computer synthesized pictures. In *Proc. SIGGRAPH, 1977*, pages 192–198, 1977.

[7] Edmond Boyer. Object models from contour sequences. In *Proc. 4th European Conference on Computer Vision*, pages 109–118, April 1996.

[8] M. J. Brooks and B. K. P. Horn. Shape and source from shading. In *Proc. 9th Int. Joint Conf. on Artificial Intelligence*, pages 932–936, Los Angeles, CA, 1985.

[9] M. T. Chiaradia, A. Distante, and E. Stella. Three-dimensional surface reconstruction integrating shading and sparse stereo data. *Optical Engineering*, 28(9):935–942, September 1989.

[10] P. H. Christensen and L. G. Shapiro. 3-dimensional shape from color photometric stereo. *International Journal of Computer Vision*, 13(2):213–227, 1994.

[11] Roberto Cipolla and Andrew Blake. Surface shape from the deformation of apparent contours. *International Journal of Computer Vision*, 9(2):83–112, 1992.

[12] J. Clark and A. Yuille. *Data Fusion for sensory information processing systems.* Kluwer Academic Publisher, 1990.

[13] James J. Clark. Active photometric stereo. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1992*, pages 29–34, June 1992.

[14] E. N. Coleman Jr. and R. Jain. Shape from shading for surfaces with texture and specularity. In *Proc. 7th Int. Joint Conf. on Artificial Intelligence*, pages 652–657, 1981.

[15] E. N. Coleman Jr. and R. Jain. Obtaining 3-dimensional shape of textured and specular surfaces using four-source photometry. *Computer Graphics and Image Processing*, 18:309–328, 1982.

[16] R. L. Cook and K. E. Torrance. A reflectance model for computer graphics. *ACM Transactions on Graphics*, 1:7–24, 1982.

[17] P. Dupuis and J. Oliensis. Direct method for reconstructing shape from shading. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1992*, pages 453–458, 1992.

[18] Herbert Edelsbrunner and Ernst P Mucke. Three dimensional alpha shape. *ACM Transactions on Graphics*, 13:43–72, 1994.

[19] J.D. Foley, Andries van Dam, Steven K. Feiner, and John F. Hughes. *Computer Graphics: Principles and Practice*. Addison-Wesley, 1990.

[20] David Forsyth and Andrew Zisserman. Reflections on shading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(7):671–679, July 1991.

[21] R. T. Frankot and R. Chellappa. A method for enforcing integrability in shape from shading algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10:439–451, 1988.

[22] P. Fua and Y.G. Leclerc. Using 3-dimensional meshes to combine image-based and geometry-based constraints. In *Proc. 3rd European Conference on Computer Vision*, pages 282–291, May 1994.

[23] P. Giblin and R. Weiss. Reconstruction of surfaces from profiles. In *Proc. 1th International Conference on Computer Vision*, 1987.

[24] H. Hayakawa. Photometric stereo under a light source with arbitrary motion. *Journal of the Optical Society of America, A*, 11:3079–3089, 1994.

[25] Xiao D. He, Kenneth E. Torrance, Francois X. Sillion, and Danald P. Greenberg. A comprehensive physical model for light reflection. In *Proc. SIGGRAPH, 1991*, pages 187–196, 1991.

[26] H. Hoppe and T. DeRose *et al* . Piecewise smooth surface reconstruction. In *Proceedings of SIGGRAPH '94, July 24–29*, Orlando, FL, 1994.

[27] B. K. P. Horn. Obtaining shape from shading information. In P. H. Winston, editor, *The Psychology of Computer Vision*, pages 115–155. McGraw–Hill, New York, 1975.

[28] B. K. P. Horn. Understanding image intensities. *Artificial Intelligence*, 8:201–231, 1977.

[29] B. K. P. Horn. Hill-shading and the reflectance map. *Proceedings of the IEEE*, 69:14–47, 1981.

[30] B. K. P. Horn. *Robot Vision*. MIT Press, Cambridge, MA, 1986.

[31] B. K. P. Horn. Height and gradient from shading. *International Journal of Computer Vision*, 5:37–75, 1990.

[32] B. K. P. Horn and M. J. Brooks. The variational approach to shape from shading. *Computer Vision Graphics and Image Processing*, 33:174–208, 1986.

[33] B. K. P. Horn and M. J. Brooks, editors. *Shape from Shading*. MIT Press, Cambridge, MA, 1989.

[34] B. K. P. Horn and R. W. Sjoberg. Calculating the reflectance map. *Applied Optics*, 18:1770–1779, 1979.

[35] Darrell R. Hougen and Narendra Ahuja. Estimation of the light source distribution and its use in integrated shape recovery from stereo and shading. In *Proc. 4th International Conference on Computer Vision*, pages 148–155, May 1993.

[36] K. Ikeuchi. Determining a depth map using a dual photometric stereo. *International Journal of Robotics Research*, 6(1):15–31, 1987.

[37] K. Ikeuchi and K. Sato. Determining reflectance function parameters using range and brightness images. In *Proc. 3rd International Conference on Computer Vision*, pages 340–343, 1990.

[38] Y. Iwahori, A. Bagheri, and R.J. Woodham. Neural network implementation of photometric stereo. In *Vision Interface 95*, pages 81–88, Quebec, PQ, 1995.

[39] Yuji Iwahori, Hidezumi Sugie, and Naohiro Ishii. Reconstructing shape from shading under point light source illumination. In *Proc. 10th International Conference on Pattern Recognition*, pages 83–87. IEEE, 1990.

[40] J. J. Koenderink and A. J. van Doorn. Geometrical modes as a general method to treat diffuse interreflections in radiometry. *Journal of the Optical Society of America*, 73(6):843–850, 1983.

[41] Y. G. Leclerc and A. F. Bobick. The direct computation of height from shading. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1991*, pages 552–558, 1991.

[42] C. Lee and A. Rosenfeld. Improved methods of estimating shape from shading using the light source coordinate system. TR-1277, U. Maryland Dept. Computer Science, College Park, MD, 1983.

[43] Kyoung Mu Lee and C.-C. Jay Kuo. Shape from shading with a linear triangular element surface model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(8):815–822, August 1993.

[44] Simon Lee and Michael Brady. Integrating stereo and photometric stereo to monitor the development of glaucoma. *Image and Vision Computing*, 9:39–44, February 1991.

[45] Y. Li. *Orientation-based representations of shape and attitude determination.* PhD thesis, The University of British Columbia, Vancouver, BC, 1993. Ph.D., supervisor R.J. Woodham.

[46] Matlab. High-performance numeric computation and visualization software. 1993.

[47] T. McInerney and D. Terzopoulos. A finite element model for 3d shape reconstruction and nonrigid motion tracking. In *Proc. 4th International Conference on Computer Vision*, pages 518–523. IEEE, 1993.

[48] S. Nayar, K. Ikeuchi, and T. Kanade. Surface reflection: physical and geometrical perspectives. In *Proceedings: Image Understanding Workshop*, pages 185–212. DARPA, 1990.

[49] S. K. Nayar, K. Ikeuchi, and T. Kanade. Shape from interreflections. *International Journal of Computer Vision*, 6(3):173–195, 1991.

[50] Shree K. Nayar, Katsushi Ikeuchi, and Takeo Kanade. Determining shape and reflectance of hybrid surfaces by photometric sampling. *IEEE Transactions on Robotics and Automation*, 6(4):418–431, 1990.

[51] Michael Oren and Shree K. Nayar. Seeing beyond Lambert's law. In *Proc. 3rd European Conference on Computer Vision*, pages 269–280, May 1994.

[52] Michael Oren and S.K.Nayar. Generalization of the Lambertian model and implications for machine vision. *International Journal of Computer Vision*, 14(3):227–251, 1995.

[53] Sharath Pankanti, Anil K. Jain, and M. Tuceryan. On integration of vision modules. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1994*, pages 316–322, June 1994.

[54] A. P. Pentland. Local shading analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:170–187, 1984.

[55] A. P. Pentland. Linear shape from shading. *International Journal of Computer Vision*, 4:153–162, 1990.

[56] T. Poggio, E. Gamble Jr., and J. J. Little. Parallel integration of vision modules. *Science*, 242(4877):436–440, October 1988.

[57] T. Poggio, V. Torre, and C. Koch. Computational vision and regularization theory. *Nature*, 317:314–319, 1985.

[58] C. E. Siegerist. Near-real-time implementation of multiple light source optical flow. Master's thesis, The University of British Columbia, Vancouver, BC, 1996. M.Sc., supervisor R.J. Woodham.

[59] W. M. Silver. *Determining shape and reflectance using multiple images.* Ms thesis, MIT Dept. Electrical Engineering and Computer Science, Cambridge, MA, 1980.

[60] Fredric Solomon and Katsushi Ikeuchi. Extracting the shape and roughness of specular lobe objects using four light photometric stereo. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1992*, pages 466–471, June 1992.

[61] R. Szeliski. Shape from rotation. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1991*, pages 625–630, June 1991.

[62] R. Szeliski and D. Tonnesen. Surface modeling with oriented particle systems. In *Computer Graphics, (SIGGRAPH'92)*, pages 185–193, 1992.

[63] R. Szeliski, D. Tonnesen, and D. Terzopoulos. Modeling surfaces of arbitrary topology with dynamic particles. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1993*, pages 82–87, New York, NY, USA, 1993.

[64] R. Szeliski and R. Weiss. Robust shape recovery from occluding contours using a linear smoother. In *Image Understanding Workshop*, pages 939–948, Whshington, D. C., 1993.

[65] Hemant D. Tagare and Rui J.P. deFigueiredo. Simultaneous estimation of shape and reflectance maps from photometric stereo. In *Proc. 3rd International Conference on Computer Vision*, pages 340–343, 1990.

[66] Hemant D. Tagare and Rui J.P. deFigueiredo. A theory of photometric stereo for a class of diffuse non-Lambertian surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(2):133–152, 1991.

[67] D. Terzopoulos and D. Metaxas. Dynamic 3d models with local and global deformations: deformable superquadrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(7):703–714, July 1991.

[68] D. Terzopoulos and M. Vasilescu. Sampling and reconstruction with adaptive meshes. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1991*, pages 70–75. IEEE, 1991.

[69] K. Torrance and E. Sparrow. Theory for off-specular reflection from roughened surfaces. *Journal of the Optical Society of America*, 57:1105–1114, 1967.

[70] Regis Vaillant and Olivier D. Faugeras. Using extremal boundaries for 3-D object modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):157–172, February 1992.

[71] Omar E. Vega and Yee Hong Yang. Shading logic: A heuristic approach to recover shape from shading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(6):592–597, 1993.

[72] T. Wada, Y. Fukagawa, and T. Matsuyama. Shape from shading with interreflections under proximal light source: 3d shape reconstruction of unfolded book surface from a scanner image. In *Proc. 5th International Conference on Computer Vision*, pages 66–71, 1995.

[73] G. J. Ward. Measuring and modeling anisotropic reflection. In *Proc. SIGGRAPH, 1992*, pages 265–272, 1992.

[74] L. B. Wolff. Diffuse reflection. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1992*, pages 472–478, June 1992.

[75] L. B. Wolff. Diffuse-reflection model for smooth dielectric surfaces. *Journal of the Optical Society of America*, 11(11):2956–2968, November 1994.

[76] L. B. Wolff. Generalizing Lambert's law for smooth surfaces. In *Proc. 4th European Conference on Computer Vision*, pages 40–53, April 1996.

[77] L. B. Wolff and E. Angelopoulou. Three-dimensional stereo by photometric ratios. *Journal of the Optical Society of America, A*, 11(11):3069–3078, 1994.

[78] R. J. Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineering*, 19:139–144, 1980.

[79] R. J. Woodham. Analysing images of curved surfaces. *Artificial Intelligence*, 17:117–140, 1981.

[80] R. J. Woodham. Determining surface curvature with photometric stereo. In *IEEE Conf. Robotics & Automation*, pages 36–42, Scottsdale, AZ, 1989.

[81] R. J. Woodham. Multiple light source optical flow. In L. Wolff, S. Shafer, and G. Healey, editors, *Physics-Based Vision: Principles and Practice (Vol III: Shape Recovery)*, pages 417–421. Jones and Bartlett Publishers, Inc., Boston, MA, 1992. Reprint of *Proc. 3rd Int. Conf. Computer Vision* pp. 42–46 (1990).

[82] R. J. Woodham. Gradient and curvature from photometric stereo including local confidence estimation. *Journal of the Optical Society of America*, 11(11):3050–3068, November 1994.

[83] J. Y. Zheng. Acquiring 3-D models from sequences of contours. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(2):163–178, 1994.

[84] J. Y. Zheng, Y. Fukagawa, and N. Abe. Shape and model from specular motion. In *Proc. 5th International Conference on Computer Vision*, pages 72–79, 1995.

[85] J. Y. Zheng and Fumio Kishino. Verifying and combining different visual cues into a 3D model. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1992*, pages 777–780, June 1992.

[86] Q. Zheng and R. Chellappa. Estimation of illuminant direction, albedo, and shape from shading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(7):680–702, 1991.

Appendix A

Noise Effects on Specular and Hybrid Surfaces

For a uniform hybrid surface, let its surface reflectance function be $E = Q(\cos e)$ and its inverse function be $\cos e = Q^{-1}(E)$. In the presence of image noise process n, $\tilde{E} = E + n$ and the estimated value for $\cos e$ is

$$\cos \tilde{e} = Q^{-1}(\tilde{E}) = Q^{-1}(E + n).$$

Since the function $E = Q(\cos e)$ is a continuous function, so is its inverse $\cos e = Q^{-1}(E)$. Assuming $n \ll E$ and applying linear approximation to Q^{-1} yield

$$\cos \tilde{e} = Q^{-1}(E + n) = Q^{-1}(E) + \frac{d(Q^{-1}(E))}{dE} n = \cos e + \frac{n}{Q'(\cos e)}.$$

Usually, $0 \leq \cos e \leq 1$ and $0 \leq E \leq 255$. If Q' is relatively uniform with the defined range, $Q'(\cos e) \gg 1$ and $\frac{n}{Q'(\cos e)\cos e} \ll 1$ when n is small.

Let the brightness value of a surface point in $image_0$ be $\tilde{E}_0 = E_0 + n_0$, the brightness value of the surface point in $image_\alpha$ be $\tilde{E}_\alpha = E_\alpha + n_\alpha$ with n_0 and n_α two independent Gaussian noise processes of zero means. Using $\tilde{E}_0$ and $\tilde{E}_\alpha$ to estimate $\cos e_0$ and $\cos e_\alpha$ gives

$$\cos \tilde{e}_0 = Q^{-1}(\tilde{E}_0) = \cos e_0 + \frac{n_0}{Q'(\cos e_0)} \quad \& \quad \cos \tilde{e}_\alpha = Q^{-1}(\tilde{E}_\alpha) = \cos e_\alpha + \frac{n_\alpha}{Q'(\cos e_\alpha)}.$$

Then the p component at the point can be estimated by

$$\frac{1}{\tan \alpha} - \frac{1}{\sin \alpha} \frac{Q^{-1}(\tilde{E}_0)}{Q^{-1}(\tilde{E}_\alpha)} = \frac{1}{\tan \alpha} - \frac{1}{\sin \alpha} \frac{\cos e_\alpha (1 + \frac{n_\alpha}{Q'(\cos e_\alpha)\cos e_\alpha})}{\cos e_0 (1 + \frac{n_0}{Q'(\cos e_0)\cos e_0})}.$$

As $\frac{n_0}{Q'(\cos e_0)\cos e_0} \ll 1$, using the binomial series expansion, we have

$$\frac{1}{1+\frac{n_0}{Q'(\cos e_0)\cos e_0}} \approx 1 - \frac{n_0}{Q'(\cos e_0)\cos e_0}.$$

The computed p component at the point can written as

$$\tilde{p} = p + \frac{1}{\sin\alpha}\left(\frac{n_\alpha}{Q'(\cos e_\alpha)\cos e_0} - \frac{n_0}{(Q'(\cos e_0)\cos e_0)^2} - \frac{n_0 n_\alpha}{Q'(\cos e_0)Q'(\cos e_\alpha)\cos^2 e_0}\right),$$

where p is the actual value at the point and the second term is the error caused by image noise.

The depth $\tilde{z}$ computed by integrating $\tilde{p}$ along the x direction is

$$\tilde{z} = z + \frac{1}{\sin\alpha}\int_{x_0}^{x}\left(\frac{n_\alpha}{Q'(\cos e_\alpha)\cos e_0} - \frac{n_0}{(Q'(\cos e_0)\cos e_0)^2} - \frac{n_0 n_\alpha}{Q'(\cos e_0)Q'(\cos e_\alpha)\cos^2 e_0}\right)dx.$$

The first term z is the accurate depth and the second term is the depth error caused by image noise. Since n_0 and n_α are independent, the deviation of the integration of $\frac{n_0 n_\alpha}{Q'(\cos e_0)Q'(\cos e_\alpha)\cos^2 e_0}$ will be zero. Comparing the magnitudes of $Q'(\cos e_0)$ and $Q'(\cos e_\alpha)$ with the magnitudes of n_0 and n_α, we can derive the same conclusion we had on the effects of image noise on a Lambertian surface (see Subsection 7.4.1).

Appendix B

Effects of Non-collinear Illumination on a Specular Surface

For a uniform specular surface, the surface reflectance function can be expressed as

$$E = g(\cos j).$$

The specular angle j $(0 \leq j \leq \pi/2)$ is the angle from the surface normal to the specular vector. The specular vector is halfway between the viewing direction and the illuminant direction. Let the illuminant direction of a non-collinear light source be $(0, -q_l, 1)$, and q_l sufficient small; then the specular vector is $(0, -q_h, 1)$ with $q_h = q_l/2$. Assume the surface reflectance function is built, in the same way as before, from a sample point S whose image in $image_0$ is a singular image point under the non-collinear illumination. Since S has the maximum brightness in $image_0$, its initial surface orientation is the same as the specular vector, that is $(0, -q_h, 1)$. After rotation by α,

$$p_\alpha(S) = \tan\alpha, \quad \& \quad q_\alpha(S) = \frac{q_h}{\cos\alpha},$$

$$\cos j_\alpha(S) = \frac{(-p_\alpha(S), -q_\alpha(S), 1) \cdot (0, -q_h, 1)}{\sqrt{p_\alpha^2(S) + q_\alpha^2(S) + 1}\sqrt{q_h^2 + 1}} = \frac{q_h^2 + \cos\alpha}{q_h^2 + 1},$$

and

$$E_\alpha(S) = g(\cos j_\alpha(S)) = g\left(\frac{q_h^2 + \cos\alpha}{q_h^2 + 1}\right).$$

If we approximate the non-collinear light source as a collinear light source and consider $\cos\tilde{e}(S) = \cos\alpha$ as the estimated value for $\cos e(S)$, the surface reflectance function built from the sample points and its reverse are

$$E = Q(\cos\tilde{e}) = g\left(\frac{q_h^2 + \cos\tilde{e}}{q_h^2 + 1}\right) \quad \& \quad \cos\tilde{e} = Q^{-1}(E) = g^{-1}(E)(q_h^2 + 1) - q_h^2.$$

Let a surface point have surface orientation $(-p_0, -q_0, 1)$; then the image brightness value of this surface point is

$$E_0 = g(\cos j_0) = g\left(\frac{q_0 q_h + 1}{\sqrt{q_h^2+1}\sqrt{p_0^2+q_0^2+1}}\right). \tag{B.53}$$

After rotation by α, the brightness value of this surface point is

$$E_\alpha = g(\cos j_\alpha) = g\left(\frac{q_0 q_h + \cos\alpha - p_0\sin\alpha}{\sqrt{q_h^2+1}\sqrt{p_0^2+q_0^2+1}}\right). \tag{B.54}$$

To compute the p component for the point,

$$\tilde{p} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{Q^{-1}(E_0)}{Q^{-1}(E_\alpha)} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{\cos\tilde{e}_0}{\cos\tilde{e}_\alpha} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{g^{-1}(E_0)(q_h^2+1) - q_h^2}{g^{-1}(E_\alpha)(q_h^2+1) - q_h^2}$$

Substituting E_0 and E_α with Equation B.53 and B.54, we have

$$\tilde{p} = \frac{1}{\tan\alpha} - \frac{1}{\sin\alpha}\frac{(q_0 q_h + \cos\alpha - p_0\sin\alpha)\sqrt{(q_h^2+1)} - q_h^2(\sqrt{(p_0^2+q_0^2+1)}}{(q_0 q_h + 1)\sqrt{(q_h^2+1)} - q_h^2(\sqrt{(p_0^2+q_0^2+1)}}.$$

The rest of the calculation is similar to that in deriving surface orientation error caused by non-collinear illumination for a Lambertian surface (see Subsection 7.4.2). Noting that $q_h = q_l/2$, finally we have

$$\delta p = \frac{q_l}{2} q_0 \frac{\cos\alpha - p_0\sin\alpha - 1}{\sin\alpha};$$

$$\delta q = \pm\frac{q_l}{2}[\frac{p_0(1-\cos\alpha)}{\sin\alpha} - (1+q_0^2)].$$

Appendix C

Effects of Non-collinear Illumination on a Hybrid Surface

The surface reflectance function of a uniform hybrid surface is a function of both the incident and specular angle. It can be written as

$$E = A + f(\cos i) + g(\cos j).$$

The three terms in the function are the ambient component, Lambertian component and specular component, respectively. The ambient component is a constant which represents the object surface reflection caused by the illumination of the reflected light in the environment.

Let the illuminant direction of a non-collinear light source be $(0, -q_l, 1)$, and q_l sufficient small; then the specular vector is $(0, -q_h, 1)$ with $q_h = q_l/2$. Taking the light source as a collinear light source and tracking brightness values of some sample points as described in Section 4.2, the surface reflectance function for a hybrid surface can be built. Let surface point S be a sample surface point whose image in $image_0$ is a singular point. Since S gives the maximum brightness value in $image_0$ and both g and f are monotonic increasing positive functions, its surface orientation will lie in between the illuminant direction and the specular vector. Thus if the surface orientation of S is $(0, -q_{lh}, 1)$, q_{lh} is between q_l and q_h. After α rotation of the object, the brightness value of S is

$$E_\alpha(S) = A + f(\cos i_\alpha(S)) + g(\cos j_\alpha(S)).$$

with

$$\cos i_\alpha(S) = \frac{q_l q_{lh} + \cos\alpha}{\sqrt{q_l^2 + 1}\sqrt{q_{lh}^2 + 1}} \tag{C.55}$$

and

$$\cos j_\alpha(S) = \frac{q_h q_{lh} + \cos\alpha}{\sqrt{q_h^2+1}\sqrt{q_{lh}^2+1}}. \tag{C.56}$$

By tracking the brightness value of the sample point S and taking the rotation angle α as the approximation for the emergent angle of S, we build the surface reflectance function

$$E_\alpha(S) = Q(\cos\alpha) = Q(\cos\tilde{e}(S)).$$

Applying the inverse function to estimate the $\cos e$ from image brightness value, we have

$$\cos\tilde{e} = Q^{-1}(E).$$

We will prove that, for any surface point, the range of $\cos\tilde{e}$ estimated from the above inverse function can be determined from its real surface orientation.

Let a surface point d have orientation $(-p(d), -q(d), 1)$. Then the brightness value of this point

$$E(d) = A + f(\cos i(d)) + g(\cos j(d))$$

with

$$\cos i(d) = \frac{q(d)q_l + 1}{\sqrt{q_l^2+1}\sqrt{p(d)^2+q(d)^2+1}} \tag{C.57}$$

and

$$\cos j(d) = \frac{q(d)q_h + 1}{\sqrt{q_h^2+1}\sqrt{p(d)^2+q(d)^2+1}}. \tag{C.58}$$

If $E(d) = E_\alpha(S)$,

$$A + f(\cos i(d)) + g(\cos j(d)) = A + f(\cos i_\alpha(S)) + g(\cos j_\alpha(S)).$$

Since both f and g are monotonic increasing and positive functions, either

$$\cos i(d) \leq \cos i_\alpha(S) \quad \& \quad \cos j_(d) \geq \cos j_\alpha(S)$$

or

$$\cos i(d) \geq \cos i_\alpha(S) \quad \& \quad \cos j_(d) \leq \cos j_\alpha(S).$$

Applying Q^{-1} to the brightness value $E(d)$, we get the estimated value $\cos \tilde{e}(d)$ for $\cos e(d)$ and

$$\cos \tilde{e}(d) = Q^{-1}(E(d)) = Q^{-1}(E_\alpha(S)) = \cos \alpha. \tag{C.59}$$

Theorem 1. If $\cos i(d) \leq cosi_\alpha(S)$ and $\cos j(d) \geq \cos j_\alpha(S)$,

$$\frac{(q(d)q_h + 1)\sqrt{q_h^2 + 1}}{\sqrt{p(d)^2 + q(d)^2 + 1}} - q_{lh}q_h \geq \cos \tilde{e}(d) \geq \frac{(q(d)q_l + 1)\sqrt{q_l^2 + 1}}{\sqrt{p(d)^2 + q(d)^2 + 1}} - q_{lh}q_l. \tag{C.60}$$

Proof: From Equation C.59 and C.55,

$$\cos \tilde{e}(d) = \cos \alpha = \cos i_\alpha(S)\sqrt{q_l^2 + 1}\sqrt{q_{lh}^2 + 1} - q_l q_{lh}.$$

Since $\cos i(d) \leq \cos i_\alpha(S)$,

$$\cos \tilde{e}(d) \geq \cos i(d)\sqrt{q_l^2 + 1}\sqrt{q_{lh}^2 + 1} - q_l q_{lh}.$$

Substituting $\cos i(d)$ with Equation C.57 and simplifying the expression yield

$$\cos \tilde{e}(d) \geq \frac{(q_l q(d) + 1)\sqrt{q_l^2 + 1}}{\sqrt{p(d)^2 + q(d)^2 + 1}} - q_{lh}q_l. \tag{C.61}$$

From Equation C.59 and C.56,

$$\cos \tilde{e}(d) = \cos \alpha = \cos j_\alpha(S)\sqrt{q_h^2 + 1}\sqrt{q_{lh}^2 + 1} - q_h q_{lh}.$$

Since $\cos j(d) \geq \cos j_\alpha(S)$,

$$\cos \tilde{e}(d) \leq \cos j(d)\sqrt{q_h^2 + 1}\sqrt{q_{lh}^2 + 1} - q_h q_{lh}.$$

Substituting $\cos j(d)$ with Equation C.58 and simplifying the expression give

$$\cos \tilde{e}(d) \leq \frac{(q_h q(d) + 1)\sqrt{q_h^2 + 1}}{\sqrt{p(d)^2 + q(d)^2 + 1}} - q_{lh}q_h. \tag{C.62}$$

Combing Equation C.61 and C.62, we finish the proof.

Theorem 2. If $\cos i(d) \geq cosi_{\alpha}(S)$ and $\cos j(d) \leq \cos j_{\alpha}(S)$,

$$\frac{(q(d)q_h + 1)\sqrt{q_h^2 + 1}}{\sqrt{p(d)^2 + q(d)^2 + 1}} - q_{lh}q_h \leq \cos \tilde{e}(d) \leq \frac{(q(d)q_l + 1)\sqrt{q_l^2 + 1}}{\sqrt{p(d)^2 + q(d)^2 + 1}} - q_{lh}q_l. \tag{C.63}$$

The proof of Theorem 2 is similar to the proof of Theorem 1.

The right side in Expression C.60 and C.63 is the estimated value for $\cos \tilde{e}(d)$ when the brightness value at point d is assumed to contain only the Lambertian component. The left side in Expression C.60 and C.63 is the estimated value for $\cos \tilde{e}(d)$ when the brightness value at point d is assumed to contain only the specular component. The above two theorems state that the estimated value for $\cos e$ from the brightness at a point on a hybrid surface is within a range determined by the values computed under a Lambertian surface assumption and a specular surface assumption.

Now, we can estimate the error caused by noncollinear illumination in computing surface orientation for points on a hybrid surface. Let a surface point d have surface orientation $(-p_0, -q_0, 1)$ and its brightness values in $image_0$ and $image_{\beta}$ be E_0 and E_{β} respectively. The estimated values for $\cos e_0$ and $\cos e_{\beta}$ from the brightness values E_0 and E_{β} will be $\cos \tilde{e}_0$ and $\cos \tilde{e}_{\beta}$. From Theorem 1 and 2, $\cos \tilde{e}_0$ is within the range defined by either Expression C.60 or Expression C.63. We can prove that it does not matter which range $\cos \tilde{e}_0$ is in, the final results are similar. So we only consider

$$\frac{(q_0 q_h + 1)\sqrt{q_h^2 + 1}}{\sqrt{p_0^2 + q_0^2 + 1}} - q_{lh}q_h \geq \cos \tilde{e}_0 \geq \frac{(q_0 q_l + 1)\sqrt{q_l^2 + 1}}{\sqrt{p_0^2 + q_0^2 + 1}} - q_{lh}q_l.$$

After the object rotation by β, there are two cases for $\cos \tilde{e}_{\beta}$, either

$$\frac{(q_0 q_h - \cos\beta + p_0 \sin\beta)\sqrt{q_h^2+1}}{\sqrt{p_0^2+q_0^2+1}} - q_{lh}q_h \geq \cos \tilde{e}_{\beta} \geq \frac{(q_0 q_l - \cos\beta + p_0 \sin\beta)\sqrt{q_l^2+1}}{\sqrt{p_0^2+q_0^2+1}} - q_{lh}q_l$$

or

$$\frac{(q_0 q_h - \cos\beta + p_0 \sin\beta)\sqrt{q_h^2+1}}{\sqrt{p_0^2+q_0^2+1}} - q_{lh}q_h \leq \cos \tilde{e}_{\beta} \leq \frac{(q_0 q_l - \cos\beta + p_0 \sin\beta)\sqrt{q_l^2+1}}{\sqrt{p_0^2+q_0^2+1}} - q_{lh}q_l.$$

The p component at point d is computed by $p = \frac{1}{\tan\beta} - \frac{1}{\sin\beta}\frac{Q^{-1}(E_\beta)}{Q^{-1}(E_0)} = \frac{1}{\tan\beta} - \frac{1}{\sin\beta}\frac{\cos\tilde{e}_\beta}{\cos\tilde{e}_0}$. Since $\cos\tilde{e}_\beta$ and $\cos\tilde{e}_0$ are within certain ranges, the computed p component will also be within a certain range. The maximum and minimum values for the p component are determined by the minimum and maximum values of $\frac{\cos\tilde{e}_\beta}{\cos\tilde{e}_0}$. The maximum and minimum values of $\frac{\cos\tilde{e}_\beta}{\cos\tilde{e}_0}$ are determined by $\frac{max(\cos\tilde{e}_\beta)}{min(\cos\tilde{e}_0)}$ and $\frac{min(\cos\tilde{e}_\beta)}{max(\cos\tilde{e}_0)}$, respectively.

In the first case, using the maximum value of $\frac{\cos\tilde{e}_\beta}{\cos\tilde{e}_0}$ to compute the p component, the estimated value $\tilde{p}$ is given by,

$$\tilde{p} = \frac{1}{\tan\beta} - \frac{(q_h q_0 + \cos\beta - p_0\sin\beta)\sqrt{q_h^2+1} - q_{lh}q_h\sqrt{(p_0^2+q_0^2+1)}}{\sin\beta[(q_l q_0 + 1)\sqrt{q_l^2+1} - q_{lh}q_l\sqrt{(p_0^2+q_0^2+1)}]}.$$

Substituting $\frac{q_l}{2}$ for q_h and aq_l ($a = q_{lh}/q_l$) for q_{lh}, $\tilde{p}$ becomes a function of q_l. Since q_l is sufficient small, the error δp can be computed by using the first derivative of the function with respect to q_l. Then

$$\delta p = \frac{d\tilde{p}}{d(q_l)}(0)q_l = q_l q_0\frac{\cos\beta - p_0\sin\beta - 0.5}{\sin\beta}.$$

If the minimum value of $\frac{\cos\tilde{e}_\beta}{\cos\tilde{e}_0}$ is used to compute the p component, then

$$\tilde{p} = \frac{1}{\tan\beta} - \frac{(q_l q_0 + \cos\beta - p_0\sin\beta)\sqrt{q_l^2+1} - q_{lh}q_l\sqrt{(p_0^2+q_0^2+1)}}{\sin\beta[(q_h q_0 + 1)\sqrt{q_h^2+1} - q_{lh}q_h\sqrt{(p_0^2+q_0^2+1)}]}.$$

Similarly,

$$\delta p = q_l q_0\frac{0.5(\cos\beta - p_0\sin\beta) - 1}{\sin\beta}.$$

So the error in computing the p component for a surface point of orientation $(-p_0, -q_0, 1)$ on a hybrid surface is within the range defined by

$$\begin{aligned}\delta p \quad = \quad & [min\left(q_l q_0\tfrac{\cos\beta - p_0\sin\beta - 0.5}{\sin\beta}, q_l q_0\tfrac{0.5(\cos\beta - p_0\sin\beta)-1}{\sin\beta}\right), \\ & max\left(q_l q_0\tfrac{\cos\beta - p_0\sin\beta - 0.5}{\sin\beta}, q_l q_0\tfrac{0.5(\cos\beta - p_0\sin\beta)-1}{\sin\beta}\right)].\end{aligned}$$

The maximum and minimum values computed for the q component are determined by using $\frac{max(\cos\tilde{e}_\beta)}{min(\cos\tilde{e}_0)}$ and $\frac{min(\cos\tilde{e}_\beta)}{max(\cos\tilde{e}_0)}$. Similar to the calculation of δp,

$$\begin{aligned}\delta q \;=\; & [min\left(\pm q_l(p_0\tfrac{0.5-\cos\beta}{\sin\beta} - (1+q_0^2)), \pm q_l(p_0\tfrac{1-0.5\cos\beta}{\sin\beta} - (1+q_0^2))\right), \\ & max\left(\pm q_l(p_0\tfrac{0.5-\cos\beta}{\sin\beta} - (1+q_0^2)), \pm q_l(p_0\tfrac{1-0.5\cos\beta}{\sin\beta} - (1+q_0^2))\right)].\end{aligned}$$

In the above calculation, the extreme values $\frac{\cos\tilde{e}_\beta}{\cos\tilde{e}_0}$ are used to compute δp and δq. The extreme values are obtained by assuming that point d gives Lambertian reflection in one image and specular reflection in the other image. This assumption is not valid for real surfaces. Therefore the actual surface orientation error is less than that computed with the extreme values.

In the second case, δp computed by using the maximum and minimum values of $\frac{\cos\tilde{e}_\beta}{\cos\tilde{e}_0}$, respectively, are

$$\delta p = q_l q_0 \frac{\cos\beta - p_0\sin\beta - 1}{\sin\beta}$$

and

$$\delta p = q_l q_0 \frac{\cos\beta - p_0\sin\beta - 1}{2\sin\beta}.$$

The error δq computed by using the maximum and minimum values of $\frac{\cos\tilde{e}_\beta}{\cos\tilde{e}_0}$, respectively, are

$$\delta q = \pm q_l[\frac{p_0(1-\cos\beta)}{\sin\beta} - (1+q_0^2)]$$

and

$$\delta q = \pm\frac{q_l}{2}[\frac{p_0(1-\cos\beta)}{\sin\beta} - (1+q_0^2)].$$

Therefore, in the second case, the error in computing the p component for a surface point of orientation $(-p_0, -q_0, 1)$ on a hybrid surface is within the range defined by

$$\begin{aligned}\delta p \;=\; & [min\left(q_l q_0\tfrac{\cos\beta - p_0\sin\beta - 1}{\sin\beta}, q_l q_0\tfrac{\cos\beta - p_0\sin\beta - 1}{2\sin\beta}\right), \\ & max\left(q_l q_0\tfrac{\cos\beta - p_0\sin\beta - 1}{\sin\beta}, q_l q_0\tfrac{\cos\beta - p_0\sin\beta - 1}{2\sin\beta}\right)],\end{aligned}$$

and the error in computing the q component for the point is within the range defined by

$$\begin{aligned}\delta q \;=\; & [min\left(\pm q_l[\tfrac{p_0(1-\cos\beta)}{\sin\beta} - (1+q_0^2)], \pm\tfrac{q_l}{2}[\tfrac{p_0(1-\cos\beta)}{\sin\beta} - (1+q_0^2)]\right), \\ & max\left(\pm q_l[\tfrac{p_0(1-\cos\beta)}{\sin\beta} - (1+q_0^2)], \pm\tfrac{q_l}{2}[\tfrac{p_0(1-\cos\beta)}{\sin\beta} - (1+q_0^2)]\right)]\end{aligned}$$

Comparing the above δp and δq with those computed for a Lambertian and a specular surface, we see the surface orientation error in the second case is between the error computed under a Lambertian surface assumption and the error computed under a specular surface assumption. This is because in the second case the extreme values are obtained by assuming that a surface point gives Lambertian reflection or specular reflection in both images.

Finally, let $min(\delta p)$ be the minimum and $max(\delta p)$ be the maximum of the following:

$$\begin{array}{lll} q_l q_0 \frac{\cos\beta - p_0\sin\beta - 0.5}{\sin\beta}, & & q_l q_0 \frac{0.5(\cos\beta - p_0\sin\beta)-1}{\sin\beta}, \\ q_l q_0 \frac{\cos\beta - p_0\sin\beta - 1}{\sin\beta}, & and & q_l q_0 \frac{\cos\beta - p_0\sin\beta - 1}{2\sin\beta} \end{array}$$

and $min(\delta q)$ be the minimum and $max(\delta q)$ be the maximum of the following:

$$\begin{array}{lll} \pm q_l(p_0\frac{0.5-\cos\beta}{\sin\beta} - (1+q_0^2)), & & \pm q_l(p_0\frac{1-0.5\cos\beta}{\sin\beta} - (1+q_0^2)), \\ \pm q_l[\frac{p_0(1-\cos\beta)}{\sin\beta} - (1+q_0^2)] & and & \pm\frac{q_l}{2}[\frac{p_0(1-\cos\beta)}{\sin\beta} - (1+q_0^2)]. \end{array}$$

The errors in the p and q components computed from the brightness values of a surface point of orientation $(-p_0, -q_0, 1)$ on a hybrid surface is within the ranges defined by

$$\delta p = [min(\delta p), max(\delta p)] \quad \& \quad \delta q = [min(\delta q), max(\delta q)].$$

Appendix D

Brightness Gradient Difference of the Correspondence of a $p = 0$ Point

Let (x_α, y_α) and $(x_{-\alpha}, y_{-\alpha})$ be the image correspondence of a $p = 0$ point (x_0, y_0, z_0) in $image_\alpha$ and $image_{-\alpha}$ respectively, and let $(-p_\alpha, -q_\alpha, 1)$ and $(-p_{-\alpha}, -q_{-\alpha}, 1)$ be the surface orientations at the two points in the image correspondence respectively; then

$$E(x_\alpha, y_\alpha) = Q(\cos e_\alpha) \quad \& \quad E(x_{-\alpha}, y_{-\alpha}) = Q(\cos e_{-\alpha})$$

with

$$\cos e_\alpha = \frac{1}{\sqrt{p_\alpha^2 + q_\alpha^2 + 1}} \quad \& \quad \cos e_{-\alpha} = \frac{1}{\sqrt{p_{-\alpha}^2 + q_{-\alpha}^2 + 1}}$$

. From the equi-brightness constraint on a $p = 0$ point, we have $\cos e_\alpha = \cos e_{-\alpha}$ and $E(x_\alpha, y_\alpha) = E(x_{-\alpha}, y_{-\alpha})$. Since we are only interested in the image points in the same row, we will drop the y coordinate in the following expressions. From above, we have

$$\frac{\partial E(x_\alpha)}{\partial x_\alpha} = \frac{dQ(\cos e_\alpha)}{d(\cos e_\alpha)} \frac{\partial \cos e_\alpha}{\partial x_\alpha} = -\cos^3 e_\alpha Q'(\cos e_\alpha)(r_\alpha p_\alpha + t_\alpha q_\alpha) \tag{D.64}$$

with $r_\alpha = \partial p_\alpha / \partial x_\alpha$ and $t_\alpha = \partial q_\alpha / \partial x_\alpha$; and

$$\frac{\partial E(x_{-\alpha})}{\partial x_{-\alpha}} = \frac{dQ(\cos e_{-\alpha})}{d(\cos e_{-\alpha})} \frac{\partial \cos e_{-\alpha}}{\partial x_{-\alpha}} = -\cos^3 e_{-\alpha} Q'(\cos e_{-\alpha})(r_{-\alpha} p_{-\alpha} + t_{-\alpha} q_{-\alpha}) \tag{D.65}$$

with $r_{-\alpha} = \partial p_{-\alpha} / \partial x_{-\alpha}$ and $t_{-\alpha} = \partial q_{-\alpha} / \partial x_{-\alpha}$. Let the surface orientation of the $p = 0$ point be $(-p_0, -q_0, 1)$, then $r_0 = \partial p_0 / \partial x_0$ and $t_0 = \partial q_0 / \partial x_0$. The terms p_α, q_α, r_α, t_α, $p_{-\alpha}$, $q_{-\alpha}$, $r_{-\alpha}$ and $t_{-\alpha}$ can be expressed in terms of p_0, q_0, r_0 and t_0 by using Equations 3.3–3.5. As p_0, q_0, r_0 and t_0 are functions of x_0 and y_0,

$$r_\alpha = \frac{\partial p_\alpha}{\partial x_\alpha} = \frac{\partial p_\alpha}{\partial x_0} \frac{\partial x_0}{\partial x_\alpha}$$

and

$$t_\alpha = \frac{\partial q_\alpha}{\partial x_\alpha} = \frac{\partial q_\alpha}{\partial x_0}\frac{\partial x_0}{\partial x_\alpha}.$$

From Equation 3.5

$$r_\alpha = \frac{r_0}{(\cos\alpha - p_0 \sin\alpha)^2}\frac{\partial x_0}{\partial x_\alpha}. \tag{D.66}$$

Since $x_0 = x_\alpha \cos\alpha + z_\alpha \sin\alpha$,

$$\frac{\partial x_0}{\partial x_\alpha} = \cos\alpha + p_\alpha \sin\alpha. \tag{D.67}$$

Using Equation 3.5 to substitute p_α and simplifying the expression yields

$$\frac{\partial x_0}{\partial x_\alpha} = \frac{1}{\cos\alpha - p_0 \sin\alpha}.$$

Combining Equation D.66 with Equation D.67, we get

$$r_\alpha = \frac{r_0}{(\cos\alpha - p_0\sin\alpha)^3}.$$

For t_α, we have

$$t_\alpha = \frac{t_0(\cos\alpha - p_0\sin\alpha) + r_0 q_0 \sin\alpha}{(\cos\alpha - p_0\sin\alpha)^3}.$$

Since $p_0 = 0$,

$$r_\alpha p_\alpha + t_\alpha q_\alpha = \frac{r_0(1+q_0^2)\sin\alpha + t_0 q_0 \cos\alpha}{\cos^4\alpha}. \tag{D.68}$$

Similarly,

$$r_{-\alpha} p_{-\alpha} + t_{-\alpha} q_{-\alpha} = \frac{-r_0(1+q_0^2)\sin\alpha + t_0 q_0 \cos\alpha}{\cos^4\alpha}. \tag{D.69}$$

Substituting $r_\alpha p_\alpha + t_\alpha q_\alpha$ and $r_{-\alpha} p_{-\alpha} + t_{-\alpha} q_{-\alpha}$ in Equation D.64–D.65 with Equation D.68–D.69 and noting that $\cos e_\alpha = \cos e_{-\alpha}$ and $Q'(\cos e_\alpha) = Q'(\cos e_{-\alpha})$, finally we have

$$\frac{\partial E(x_\alpha)}{\partial x_\alpha} + \frac{\partial E(x_{-\alpha})}{\partial x_{-\alpha}} = -2Q'(\cos e_\alpha) t_0 q_0$$

and

$$\frac{\partial E(x_\alpha)}{\partial x_\alpha} - \frac{\partial E(x_{-\alpha})}{\partial x_{-\alpha}} = -2Q'(\cos e_\alpha) r_0 (1+q_0^2)\tan\alpha.$$