INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

Better to illuminate than merely to shine,
to deliver to others contemplated truths than merely to contemplate.

– Saint Thomas Aquinas

University of Alberta

Using a Raster Display Device for Controlled Illumination

by

Nathan James Funk

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Master of Science.**

Department of Computing Science

Edmonton, Alberta
Fall 2005

To my parents,
I strive to be as great as you.

Abstract

Computer vision attempts to find an abstract representation of a scene from images. The ability to control the lighting of the scene simplifies various tasks in the field. This thesis develops a new controlled lighting apparatus which uses a raster display device as a light source. The setup has the advantage over other alternatives in that it is relatively inexpensive and uses commonly available components.

Two applications are examined. The first is shape reconstruction using photometric stereo. Experiments on synthetic and real images demonstrate how the depth map of an object can be recovered using only a camera and an LCD display. The second application studied is the experimental evaluation of lighting estimation techniques. The use of the setup for this purpose is demonstrated with the evaluation of Singh & Ahuja's method. Further development of the core idea of this thesis is expected to benefit a range of applications.

Acknowledgements

First and foremost I would like to thank my advisor Herb Yang for his involvement in my research. He gave me the original idea for this thesis and continuously provided essential advice for all the issues that arose during the development of this work. His immediate response to questions, his positive attitude and his encouragement when I was working on tough problems made the past two years a very enjoyable experience. Through his continuous emphasis on quality I was motivated to strive for excellence in my work.

I would also like to thank my fellow lab members Cheng Lei, Hai Mao, Daniel Neilson, Xuejie Qin, Danielle Sauer, and Jason Selzer. They provided me with helpful feedback on various aspects of this thesis work.

Prof. Martin Jägersand and Dana Cobzas were also both very helpful with their advice for my research. I would like to thank Neil Birkbeck for his input while we were both writing our theses.

A great contribution to enriching my experience here at the University of Alberta was made through the friendships with my fellow graduate students John Arnold, Baochun Bai, Shane Bergsma, Paul Berube, Colin Cherry, Jessica Enright, Markian Hlynka, Dan Lizotte, Peter McCracken, Chris Parker, Luca Pireddu, Bill Rosgen, Frano Sailer, Mark Schmidt, Brian Tanner, Robyn Taylor and Qin Wang. Sharing our experiences as graduate students was a great source of motivation and encouragement.

Finally, I would like to thank my parents to whom this thesis is dedicated. I owe them endless thanks for their guidance in my youth and for serving as role models in my adulthood.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

One of the main reasons why computer vision is difficult is because many factors affect the appearance of a scene. The image of a scene is simultaneously affected by the lighting, the geometry and the material properties of the objects in the path of the light. With the goal of estimating these factors given only an image, it is challenging and often impossible to uniquely determine the properties of all elements. The reason is that a wide range of different combinations of lighting, geometry and material properties can all produce the same image.

For this reason many computer vision techniques assume that some of these factors are already known prior to the analysis. For example, when estimating scene geometry, it is common to assume that the lighting conditions are known and that the types of materials in the scene have certain properties [46, 50]. Doing so reduces the number of unknown variables in the model, simplifying the task to be accomplished.

Lighting conditions can be complex in real scenes. Without any knowledge of the lighting, it is challenging to determine other properties of the scene. Lighting estimation methods have been proposed to counteract this problem, but they often rely on limiting assumptions about the scene. Therefore, it is desirable to be able to control the lighting of a scene to known conditions. It permits understanding the behaviour of computer vision algorithms and may contribute to future development of algorithms that take advantage of this feature. Since many shape recovery methods assume known lighting conditions, controlling the lighting of a scene to the desired state allows application of these techniques. Shape from shading and photometric stereo are examples of shape recovery methods that can benefit from the ability to control the lighting. Furthermore, lighting estimation methods can be evaluated given the ability of specifying the scene illumination. The recovered lighting conditions can be compared to the known lighting to measure the recovery method's performance.

Controlled lighting can be accomplished by positioning a large set of light sources in a manner such that a wide range of different lighting conditions are achievable [13]. Or one can use a smaller set of light sources and devise a method of accurately moving them to desired locations [15]. Two key factors are the *accuracy* of the positioning and the *ease of control* of the sources. This thesis

presents a novel method which meets these requirements through the use of a raster display device. Such devices include CRT monitors, LCD screens, and LCD projectors in combination with a projection screen. All these devices provide a dense grid of accurately positioned light emitting cells. They can be easily controlled with the use of a computer equipped with a standard graphics processor. The wide availability of raster display devices is also a definite advantage over constructing a rig specialised for controlling lighting.

The controlled illumination framework is implemented using an LCD screen for the purposes of this thesis. Its use is demonstrated and analysed through the application in two shape recovery methods, and with the performance evaluation of a lighting estimation method.

Both shape recovery methods estimate the shape of an object, based solely on a set of images taken under different lighting conditions. The display is used to generate these lighting configurations. To the best knowledge of the author, the use of display devices for this purpose has so far not been proposed in literature.

The first method recovers the shape in a two-step process. First the surface orientations are obtained, then the depth is determined from these surface orientations. For experiments using this method on a 22mm diameter sphere, the shape is estimated with 1.8mm average error in its depth from the camera. For a 180mm real model of the Stanford bunny, the average depth error is 10mm. In analogous experiments on rendered images of the models, depth errors below 1mm are achieved. The accuracy depends on a depth estimate which specifies the distance of the object to the camera. The results assume that this depth estimate is accurate. The necessity of a depth estimate is one of the limitations of the first method.

The second shape recovery method developed does not rely on a depth estimate. It recovers the depth in a single step and is thus called the *direct depth recovery* method. This is a novel approach which currently only performs well on synthetic images. Future enhancements might allow a robust performance on real images as well.

Both shape recovery methods are limited in that they only operate on diffuse reflecting surfaces, require an enclosure to reduce ambient light, and involve capture times of over one minute. Future work is expected to reduce or completely eliminate some of these requirements. This would lead to many applications in recovering shape using only a camera and display device. For example the shape of a person's face could be recovered at a bank machine for security purposes. Or a computer user could obtain a 3D model of any object placed in front of their computer monitor and camera.

The evaluation of a lighting estimation method presented in this thesis analyzes the performance of a technique proposed by Singh & Ahuja [38]. The published material for this technique, similar to many other lighting recovery papers, only examines the performance on synthetic images. So without real experiments it is difficult to know how well these techniques perform on images of real surfaces taken with real cameras. The controlled lighting setup can be used to examine many lighting estimation methods and compare their performance through the use of clearly defined evaluation

measures. The experiments performed in this thesis showed that Singh & Ahuja's method can achieve angular accuracies ranging from 1° to 17° for estimating the position of a point light source.

The main contributions of this thesis include:

- the development of a novel apparatus for controlling illumination with a raster display,
- the implementation of a shape recovery method based on photometric stereo, including the presentation of a new approach for estimating depth from surface orientation,
- the development of a new shape recovery method for direct depth recovery, and
- the experimental evaluation of Singh & Ahuja's method, as well as modifications to their technique.

Background information and related research is provided in Chapter 2 of this thesis. The controlled illumination setup as well as the associated calibration methods are discussed in Chapter 3. The next two chapters discuss the two application areas studied: shape recovery (Chapter 4) and the evaluation of lighting estimation methods (Chapter 5). Both of these chapters include an experimental analysis. The results are summarized in Chapter 6 together with a discussion of potential future work.

Chapter 2

Background and Related Work

Controlled lighting has so far not been explored in a systematic fashion but merely in an ad hoc manner to assist in specific applications. In particular, the role of lighting in various computer vision algorithms has rarely been investigated. Debevec has been incrementally refining a controlled lighting environment called the Light Stage. The first version of the Light Stage was used to capture the reflectance of human faces [11]. An apparatus was constructed to simultaneously spin and lower a single spot light over the surface of a sphere, with the light directed at a person in the centre. A video camera captured the image of the face while the position of the light source covered a wide range of positions. The result is a set of images of the face, each under different lighting conditions. The captured images are then used to create maps of how light is reflected by the skin. Finally, these maps can be used to synthesize images of the face under arbitrary lighting conditions.

To improve the speed of the acquisition process, the Light Stage 2 was constructed with a semi-circular arm which was spun around a person in centre. Along the arm, a set of strobe lights are directed at the person. This setup produces similar results as the previous one, but greatly reduces the acquisition time. The most recent version of the Light Stage [13] is constructed as a geodesic dome with red, green, blue and infrared LEDs evenly distributed over the sphere surface facing inward. This allows instantaneous control over the illumination of a person or object placed in the centre of the dome. This setup is targeted at the film industry for lighting actors under arbitrary illumination conditions. Similar light domes have been constructed for other purposes as well. For example Kawasaki et al. [29] use a light dome for an image based rendering approach.

Another controlled lighting apparatus is proposed by Furukawa et al. [15]. It is similar in construction to the Light Stage 2. A set of digital cameras and halogen lamps are placed on two concentric arcs. Each of the arcs can be rotated around the object, allowing controlled illumination which they utilize in their appearance-based object modeling approach.

The most similar controlled illumination apparatus to the one proposed in this thesis is found in the field of environment matting. Environment matting is concerned with capturing images of an object such that it can be composited on arbitrary backgrounds. Zongker et al. [53] propose an environment matting technique that uses three CRT monitors surrounding an object. A camera

captures images of the object while the monitors display a set of patterns. Zhu & Yang [52] describe an improved environment matting method with a similar setup, as shown in Figure 2.1. However neither of these papers explore the more general aspects of using display devices as illumination sources.

Figure 2.1: The environment matting setup developed by Zhu & Yang [52]. A digital camera is directed at a refractive object placed in front of a computer monitor.

The remainder of this chapter provides the background necessary for understanding the subjects in later chapters. First the image formation process is outlined as it is a fundamental concept, essential for the understanding of all computer vision techniques. The following section on camera models discusses how the camera affects the captured image and how it can be used as an instrument for measuring light intensities from a scene. The section on camera calibration provides information on both the geometric and radiometric properties of cameras.

A general overview of shape recovery methods is provided to show how the photometric stereo method compares to other approaches. The photometric stereo method is outlined in Section 2.5. This section includes a description of the original photometric stereo method as well as ten modified methods. The output of the photometric stereo algorithm is a map of surface orientations. To obtain the final shape of the object, a final step is required, and is outlined in Section 2.6.

Finally, to provide some background information about lighting estimation, the basic problem is outlined and the details of the evaluated techniques are discussed.

2.1 Image Formation

Understanding the image formation process is essential for understanding existing computer vision techniques and for developing new ones. Horn [24] provides a good overview of the process from the perspective of computer vision. This section focuses on only the topics immediately applicable to this thesis.

An image can be understood as a 2D pattern of light intensities in the image plane of a camera. Originating from light sources, the light is reflected off objects in the scene and a small fraction of this reflected light is received by the camera. In a sense, the camera acts as a measurement instrument of the light from the scene.

Instead of using the term brightness to describe light intensity, the term *radiance* is used to refer to the energy flux per unit area radiating from a surface. And *irradiance* refers to the amount of energy falling on a unit surface area. Radiance and irradiance can be measured in units of Watts/m^2.

For simplicity sake, while making many assumptions about the scene, the path of the light can be broken down into three stages as illustrated in Figure 2.2:

1. Radiance of the light source to scene irradiance
2. Scene irradiance to scene radiance
3. Scene radiance to image irradiance

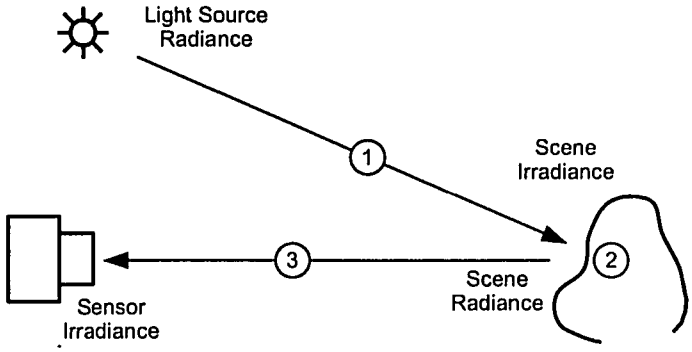

Figure 2.2: A simplified illustration of the path of light from a light source to the camera. This model assumes that there is a single reflection of the light along the path from the light source to the camera.

The true path of the light can be much more complex than in this model. Light will often not just be reflected once before entering the camera lens. *Interreflection* is the processes of light being reflected multiple times between objects in the scene. This effect is not considered in this model. Materials also do not always only reflect light, but also absorb and refract light hitting their surfaces.

Since this thesis does not consider these more complex interactions, the background information is focused on the components of the simple model. The following sections discuss the three stages as outlined earlier.

2.1.1 Light Source Radiance to Scene Irradiance

The light source radiance is determined by the light source model and parameters such as the intensity and size. The scene irradiance is determined by how this light travels through space. In this thesis the *point light source* model is used. It is assumed to be localised at a point in space, with its light spreading equally in all directions.

With increasing distance from the source, the intensity decreases according to the *inverse-square law*. This law states that the energy of the light on a unit surface area is proportional to the inverse square of the distance to the point source. It can be explained by observing that the total energy on a sphere centred around the point source is independent of the size of the sphere. Since the surface area of a sphere is proportional to the square of its radius, the intensity per unit surface area decreases accordingly with increasing radius.

2.1.2 Scene Irradiance to Scene Radiance

The physical interaction of light with surfaces is complex. The atomic structure of an object determines how light is reflected, refracted, transmitted and absorbed. All common surface models only approximate the true interaction of light with the surface.

A wide range of surfaces can be modeled with the use of a bidirectional reflectance distribution function (BRDF). Assuming that the direction of incoming light is specified by the spherical coordinates (θ_i, ϕ_i), the BRDF $f(\theta_i, \phi_i, \theta_e, \phi_e)$ determines how much of the incoming light is reflected in the direction (θ_e, ϕ_e). Many surface models are a subset of all possible BRDFs. This section focuses on one of the simplest surface models, which is however also commonly found in real objects.

Lambertian reflection which is also known as *diffuse reflection* is a model that surfaces such as paper and matt paint nearly exhibit. The main feature of this reflectance model is that the intensity of the reflected light is independent of the direction in which it is reflected. In other words, the BRDF is constant over all (θ_e, ϕ_e) directions. Light that hits a point on the surface is assumed to be scattered equally in all directions.

For a Lambertian surface lit by a point light source, the radiance of the reflected light R can be calculated from the surface albedo ρ, the surface normal $\hat{n}$, and the light source vector $\vec{L}$ as

$$R = \rho \max(0, \vec{L} \cdot \hat{n}). \tag{2.1}$$

The light source vector $\vec{L}$ points from the surface to the light source. Its magnitude represents the intensity of the light source. This equation results from the geometry of light falling on an inclined surface. The radiance R reaches a maximum when the surface is perpendicular to the light source.

As the angle between the surface normal and the light direction increases, the radiance decreases until it reaches zero.

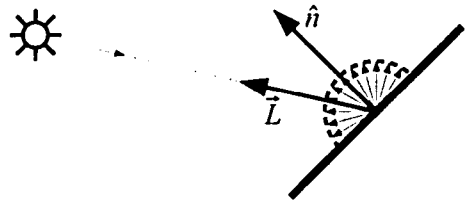

Figure 2.3: An illustration of the Lambertian reflectance model. The radiance R from the surface is determined by the surface albedo ρ, the surface normal $\hat{n}$, and the light source vector $\vec{L}$.

2.1.3 Scene Radiance to Image Irradiance

After the light is reflected from the surface, it is typically scattered in many directions, and only a small amount of the reflected light enters the camera. Since the image irradiance is the measured quantity, we need to know how it relates to the radiance from the surface.

Horn [24] shows that for a simple camera model with a single thin lens, the image irradiance I is proportional to the radiance R in direction of the camera. He derives the equation

$$I = R\frac{\pi}{4}\left(\frac{d}{f}\right)^2 \cos^4 \alpha, \tag{2.2}$$

where d is the size of the aperture and f the focal length. The *off-axis angle* α is measured between the optical axis and a ray passing through the focal point and a point on the surface. According to the equation, the radiance/irradiance ratio I/R is proportional to $\cos^4 \alpha$. This is commonly referred to as the *cos4 law*. The effect of this term is that the I/R ratio decreases for pixels further from the image centre. For example, when taking an image of a blue sky, where the irradiance is approximately equal from all directions, the corners of the image often appear darker than the centre. An interesting aspect of the above equation is that for a surface point located on a specific ray, the distance from the focal point does not affect I/R.

Real camera lenses can be much more complex than the model used by Horn as will be pointed out in the next section. In addition to the cos4 law, having multiple lenses arranged in sequence introduces *vignetting* which also darkens the corners of the image with respect to the centre. This effect can be even more significant than the cos4 law if not compensated for properly. Most lenses are constructed in a way to minimize the vignetting effect but for wide-angle lenses vignetting is difficult to avoid.

2.2 Camera Models

The main camera components in the path of the light are the lenses, the aperture and the image plane. A camera model combines the models of these elements. Lenses can be modeled with a *thin-lens model* which assumes that the lens is planar. It is not physically possible, yet still provides a close approximation to real lenses. A more realistic model is the *thick-lens* model where lenses are modeled as volumes which refract the light.

The *aperture* is a hole which determines how much light enters the camera. Since image sensors and film measure high intensities more accurately, it is typically beneficial to open the aperture as wide as possible. The drawback is that increasing the aperture size reduces the range of the scene that is in focus. An accurate camera model incorporates a set of thick-lens models with an aperture as proposed by Kolb et al. [30]. Such a model can simulate focusing and vignetting for example. This however comes at the cost of high mathematical complexity.

For many applications, a much simpler camera model can be employed. The *pinhole camera* is such a model. It is an ideal model with an infinitely small aperture and no lenses. Due to the small aperture, the entire scene is in focus. Since the benefits of the simple mathematical model outweigh the drawbacks such as the inability to simulate focus and vignetting, this model is employed in this thesis.

Hartley and Zisserman [20], as well as Horn [24] provide a good overview of the pinhole model. Figure 2.4 shows the standard model. Light from the scene enters the camera through a point referred to as the *focal point* or *camera centre*. This point is also the origin of the camera coordinate system, with the z-axis pointing towards the scene. The light is projected onto the image plane, located at $z = -f$. This results in a *perspective projection* of the scene onto the image plane.

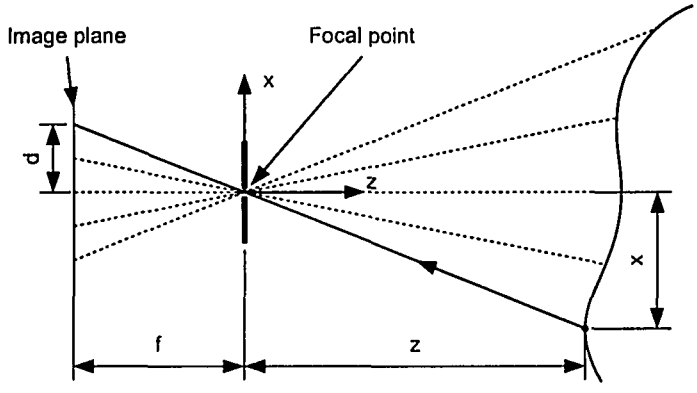

Figure 2.4: The pinhole camera is an ideal camera that projects all light through a point called the *focal point*. The coordinate system is such that the focal point is at the origin, the image plane is the plane where $z = -f$ and the z-axis coincides with the optical axis of the camera.

The only parameter of the pinhole camera is the *focal length* f. Using simple geometry one can

see that a point $p = (x, y, z)^T$ in the scene is projected to $p' = (-f\frac{x}{z}, -f\frac{y}{z}, -f)^T$ on the image plane. One can also map a point on the image plane to a line through the focal point. The line can be expressed as a normalized point $p_n = (x/z, y/z, 1)^T$ through which the line passes.

2.3 Camera Calibration

Camera calibration is the task of determining the parameters of the camera model. The term *geometric calibration* is used here to refer to the process of determining the geometric formation of the image. This is commonly referred to as *intrinsic calibration*. This process allows pixel coordinates to be mapped to rays in the scene and points in the scene to be mapped to pixel coordinates. The *radiometric calibration* relates the pixel intensity values to the irradiance on the image sensor. These mappings are all essential in using the camera as a measurement instrument for the scene.

The following sections give a brief overview of the calibration techniques used. It should be noted that there is much more detailed information about camera calibration available in other published material [12, 20, 22, 33].

2.3.1 Geometric Calibration

The goal of geometric camera calibration is to determine the mapping between the 3D world coordinates and the 2D image coordinates. In other words, it determines how a 3D object is projected onto the image plane. Typically this information is not available with the camera. Although parameters such as the focal length can be adjusted on a lens, reading the value off the camera is usually not as accurate as using a calibration method.

Most calibration methods use a set of images of a calibration object. The images are analysed by extracting positions of features such as corners and edges from the calibration object. Then the parameters are estimated by attempting to closely fit the camera model to the observed positions. Parameters are grouped as *extrinsic* and *intrinsic* parameters. The extrinsic parameters include the position and orientation of the calibration object with respect to the camera. Alternatively, the parameters can specify the orientation and position of the camera with respect to the world coordinates. And the intrinsic parameters describe the internal camera parameters. Typically these include the effective focal length, principal point, image skew, and various distortion parameters such as radial distortion.

Tsai's paper [43] is the first major advancement in camera calibration. A planar calibration pattern is used to estimate both the intrinsic and extrinsic parameters. After that article, a number of other camera calibration papers have been published.

This thesis uses the MATLAB Camera Calibration Toolbox written by Jean-Yves Bouguet [5], which is based on a paper by Heikkilä and Silvén [22]. They use a cube with a grid of dots on every side for their method. The calibration toolbox simplifies their approach to operate on a planar *checkerboard pattern*. A set of images of this pattern taken from different view points is used for

the calculation of the parameters. The implementation requires some manual interaction to select 4 corner points in each image. After that, the remaining corner points are extracted automatically and used as an input for the calibration algorithm.

The intrinsic parameters estimated by the MATLAB Camera Calibration Toolbox are:

Focal length: the focal length in both the x and y direction.

Principal point: the pixel coordinates of the point where the z-axis and the image plane intersect.

Skew coefficient: the angle between the x and y axis.

Distortion parameters: up to 5 distortion parameters including radial and tangential distortion coefficients.

The accuracy of the estimated parameters is also calculated. By recalculating the location of corner points and adding or removing images the accuracy of the estimated parameters can be improved.

After the calibration, it is easily possible to map 3D points in the camera's reference frame to image coordinates. One can also do the reverse, and map a 2D image coordinate to a ray which passes through the camera's effective focal point. This ability is foundational for many computer vision methods including photometric stereo as used in this thesis.

2.3.2 Radiometric Calibration

Camera response

As mentioned earlier, radiometric calibration relates the *pixel intensity values* from a camera to the *actual irradiance on the sensor*. Unfortunately, in most cases, these values are not linearly related. There are multiple steps in the measurement of the irradiance, some of which are not necessarily linear. Figure 2.5 shows the signal flow from the sensor to the storage files.

Most sensors used in digital cameras now are either CCD- or CMOS-based. Within a certain range of irradiance on the sensor, the response is near linear. However, above a certain irradiance level, the sensor becomes *saturated* and the output of the sensor is no longer proportional to the irradiance. Film also saturates at a certain level. So, in images taken with either a digital or film camera, there can be regions that are saturated. For example when taking an image of a scene which includes an uncovered light bulb, the region of the bulb is typically saturated. It is too bright in comparison to the rest of the scene to be captured accurately. The *dynamic range* of an image sensor specifies the range of intensities it can measure without saturating.

In addition to the sensor non-linearities, the internal processing of the camera to the raw data also affects the "measurements" in a non-linear fashion. This step is referred to here as *development processing* due to its similarity to the development of a negative into a print. The pixel values are mapped such that more of the available pixel values correspond to low intensities than to high

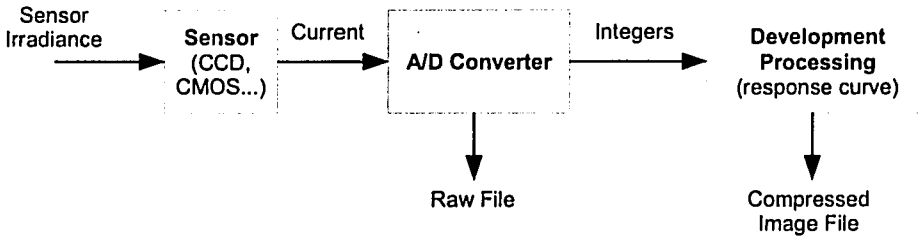

Figure 2.5: The simplified signal flow of a typical digital consumer camera. The image sensor outputs a current dependent on the irradiance on its surface. The A/D converter digitizes this analog input. Some cameras allow storing this information directly in a "raw file." The digitized values are further processed by the camera through application of a response curve termed the *development process* before writing them to a compressed image file.

intensities. There are two reasons for this. CRTs respond in a non-linear power law relationship to the input voltage. So when CRTs were first used for displaying images, the images were stored to compensate for the non-linearity of the monitors. That way, the pixel values in the image could be linearly converted to a voltage and result in a correct display of the image. The second reason is that the human visual system is non-linear in its sensitivity. It is more sensitive at low intensities than at high intensities. So non-linear mapping is an effective way of compressing the image data without loosing visual quality. It is now standard procedure to apply the non-linear mapping in consumer grade cameras where the main application is viewing the images on a display or as a print.

Standards have been established that specify how images should be stored and displayed to ensure correct appearance on a wide range of devices [40]. The process of correcting for the non-linear response of displays is referred to as *gamma correction*. Gamma correction is a simple transform where the input value is raised to the power of γ to obtain the output. The output value is calculated with

$$output = input^{\gamma}. \tag{2.3}$$

According to the sRGB standard [40], CRT displays should display intensities with a gamma value of 2.2. So, for the intensity on a CRT display to be proportional to the image irradiance in the camera, the camera needs to apply a gamma of 1/2.2 for the effects of the display response to be canceled out. The development process needs to consider the non-linear response of the sensor to ensure that the combined effect of the sensor response and processing results in an appropriate response. The combined response of all individual internal responses is commonly referred to in literature as the camera *response curve*.

Figure 2.6 shows the response functions of the three main components. The sensor saturation and development response curve are the most important nonlinearities in the processing. For computer vision purposes, nonlinearities in the response of the camera are an obstacle. Ideally, the camera

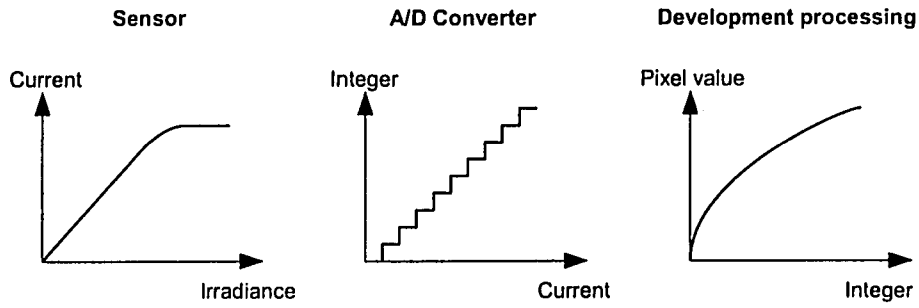

Figure 2.6: Response functions of the main camera components. The sensor and development processing steps introduce non-linearities in the irradiance measurement which need to be considered for computer vision applications.

would serve as a linear measuring device for the irradiance on the sensor. Most importantly, the response curve needs to be considered. For professional grade cameras, the gamma correction can typically be enabled or disabled. So the development response curve is linear. Intensity data from the sensor with little or no processing can be obtained directly. But with consumer grade cameras, getting the unprocessed data is not always possible.

Only high-end digital consumer cameras allow storing the data prior to gamma correction in a so-called *raw file*. For many cameras, the raw file format contains a 12-bit value for each colour channel for each pixel. If raw images are not an option, something needs to be done to reverse the gamma correction of the camera. Although most cameras use a gamma of around 1/2.2, it is often inaccurate to simply apply a gamma of 2.2 to obtain the image irradiance from the pixel values. Cameras vary in their processing of the intensity data, so calibration of the response becomes necessary.

Recovering the camera response curve

When working with intensity data that has a gamma applied to it, it is essential to determine how the values from the camera map to the irradiance. Data that has not been processed with a gamma curve can be handled in two ways. If the sensor responds linearly within the range of the measured irradiance values, then it is safe to work directly from the raw data. If the data is partially above the linear range, yet still below complete saturation of the sensor, determining the response of the sensor is helpful. In any case, if the irradiance goes beyond the saturation level of the camera, the information is irrecoverably lost.

The term radiometric calibration is used not only for recovering the camera response curve but also for determining the noise level of a camera. There is published work on this particular aspect prior to 1994, but most of the work on recovering the response curve of a camera was done later. The first camera response curve calibration papers are associated with high dynamic range (HDR)

photography [12, 33]. The dynamic range is the irradiance range that an image can hold. For HDR photography, the camera response is used to combine a set of images with differing exposure levels to a single image with a higher dynamic range than the individual images.

The first techniques use a set of photos taken at different exposure levels (just like those used for HDR photography) to determine the response curve [12, 33]. The basic principle is best explained by an example: Capture two images A and B of a still scene, but expose image B twice as long as image A. The position and orientation of the camera stay the same, only the exposure changes. We can now assume that the irradiance in image B is double that of image A.

Since the camera response curve is described by the mapping of the sensor irradiance to the pixel values, it is our goal to recover this relationship. If we denote the pixel value in image j at pixel i with Z_{ij}, then the irradiance I_{ij} is determined by

$$I_{ij} = g(Z_{ij}), \tag{2.4}$$

where g is the unknown mapping that we want to determine. From image A we get $I_{iA} = g(Z_{iA})$ and from image B, $I_{iB} = g(Z_{iB})$. Since we know that $I_{iB} = 2I_{iA}$ due to the doubled exposure time, we can form a constraint on the function g:

$$g(Z_{iB}) = 2g(Z_{iA}). \tag{2.5}$$

So, each pair of pixels from image A and B forms a constraint on the camera curve g. From these constraints, the shape of the camera curve can be determined, up to an unavoidable ambiguity of a scaling factor between the irradiance and pixel values. Mann and Picard [33] suggest a method of determining the function that meets the constraints but do not explicitly describe it. Debevec and Malik [12] however clearly outline how the suggested method can be implemented. They also introduce a smoothness constraint to counteract the effects of image noise and potential lack of samples at certain pixel values. The implementation of their method is used in this thesis and is freely available as part of the HDR-shop software [10].

More recent research in the field of camera curve calibration has been done by Grossberg and Nayar [17, 18, 19]. They base their analysis on a database of response functions of real cameras. From this, they create an empirical model of the response functions. Finally, when calibrating the response of a new camera, a set of parameters for the empirical model is estimated. They claim that the results are better than those of previous methods such as Debevec and Malik's.

2.4 Shape Recovery

The goal of shape recovery methods is to obtain a 3D model of a scene using a device that captures information of the scene. Currently, there are a variety of shape recovery methods, each with their own set of strengths and weaknesses. They differ in the following aspects:

Accuracy: The accuracy with which the surface is reconstructed. There is no standard method of measuring the accuracy. A common method is to compare the recovered and the true depth maps.

Applicable surface types: Some methods only operate on a small range of different surface types.

Input: Examples of input are a single image, multiple images from the same position, or multiple images from different positions.

Output: Some shape recovery methods recover not only shape but also surface albedo and surface reflectance information.

Active/Passive: *Active* methods project energy onto the scene, e.g. laser scanning, whereas *passive* methods do not.

Scale of scene: Most methods are limited in the scale of objects they can recover.

This section is intended to provide a brief overview of some common methods and their features.

2.4.1 Laser Range Scanning

Laser range scanning has become a popular method and is now one of the premier shape recovery methods. This can be mostly attributed to its accuracy and applicability to a relatively wide range of surfaces. The basic principle is to direct a laser beam into the scene and measure the properties of the reflected light. So naturally, for the scanning to operate properly, enough light must be reflected to be able to take measurements. This means that generally the method does not perform well on shiny surfaces. Measurements are taken at one point on the surface at a time, so the laser needs to be swept over the object to be scanned. Besl and Jain [4] offer a good overview and categorization of different laser range methods:

Pulsed mode: determines the distance by measuring the time lag between sending and receiving a light pulse

Amplitude modulated: measure distance by examining the phase difference between the received and reference signals

Triangulation-based: use a camera offset from the laser source to triangulate the position of the surface point

It is also noteworthy that traditional laser range scanning does not recover surface attributes such as colour or BRDF.

2.4.2 Structured Light

Structured light methods are similar in a sense to triangulation-based laser range scanning. Light is projected in to the scene with a known structure (e.g. stripes, a set of points, or a grid). A camera viewing the scene from an offset position then captures images of the scene lit by the projected light. Through triangulation, it is then possible to determine surface point locations.

An example of a structured light approach is Bouguet and Perona's work [6]. Their method recovers objects placed on a desk, lit by a desk lamp. A pencil, stick or wand is waved in front of the lamp to cast a shadow on the object. Then, based on the geometry of the shadows cast onto the object, its shape is recovered.

2.4.3 Multiple Views

Multiple view based shape recovery is a general term for shape recovery methods that use multiple cameras to triangulate surface point positions. Typically they are passive. The most popular method is *stereo vision* which triangulates point positions from the disparity between two views. Stereo vision requires features in the image to be matched which is not always trivial (this is referred to as the *correspondence problem*). Hartley and Zisserman [20] provide a detailed discussion on multiple view geometry which can be applied to any number of views.

2.4.4 Shading-based Methods

A subset of shape recovery methods measure the reflected radiance from surfaces to constrain the possible orientations of the surface. This is possible through consideration of the surface BRDF. The complexity of the possible surface BRDFs directly affects the complexity of the approach. Early approaches focus on Lambertian surfaces [24, 28, 46] and later methods extend the application to more general BRDF models [1, 23, 39, 41].

Since the illumination of the scene strongly influences the appearance of the reflected radiance, the lighting is typically assumed to be known. Some methods attempt to recover the illumination configuration and shape together [8, 27].

Shape from Shading

Shape from shading is a passive method that attempts to recover shape from a single image [25]. Early methods assumed that the light is a distance point light source at a known position and the surface is Lambertian. Even under these assumptions accurate results are difficult to achieve due to the ambiguity of image information. Under these conditions, an infinite number of different local surface orientations can produce the same image intensity so additional smoothness constraints are required to obtain a result. Zhang et al. [50] provide a comparison of six shape from shading methods with an evaluation of their results.

Photometric Stereo

Photometric stereo uses multiple images lit under different illumination conditions to determine surface orientation. The images are all taken from the same view point. The original approach [46] assumes Lambertian surfaces and distant point illumination. For each image, the light source position is known. In comparison to shape from shading methods, this approach has the advantage of having additional information through the multiple images. This results in a higher accuracy and robustness.

Since the output of photometric stereo is surface orientation, not depth, a separate depth from surface orientation step is necessary. The next two sections go into detail about photometric stereo and depth from surface orientation methods as they form the foundation of this thesis.

2.5 Photometric Stereo

The original photometric stereo (PS) method was proposed by Woodham in 1978 [46] and later refined [47, 48]. After the initial development of the PS method, numerous modifications to the technique have been proposed by a wide range of researchers [1, 2, 16, 21, 23, 34, 37, 39, 41]. This section first focuses on the original method since it is the one applied in this thesis. Other modified are discussed at the end of this section.

2.5.1 Simple Photometric Stereo

The basic goal of PS is to obtain a map of surface normals of an object. The input into the algorithm is a set of images, each taken from the same view but with the scene lit under different lighting conditions. For each image, the only light source is a distant point light source. This has the effect that the vector to the light source is constant over the entire surface of the object. Figure 2.7 illustrates the basic input and output of the PS method.

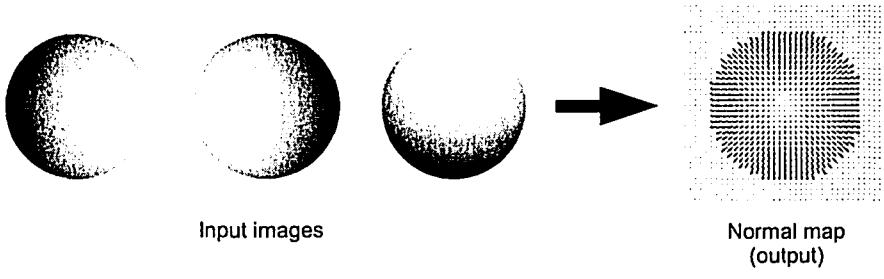

Figure 2.7: An example of the input and output of the photometric stereo algorithm. In this example, three images of a sphere under different lighting conditions serve as the input. The expected output is a map of the surface normals of the sphere.

The most important assumptions made are:

- The positions of the light sources are known.
- The visible surfaces are Lambertian.
- There are no interreflections. Points on the surface are lit solely by direct light from the light source.

An interesting aspect of PS is that it is a *local* method. The calculations are performed independently at every pixel. The surface normal at each pixel is calculated only from the pixel values at that location in the set images. For now, assume that the pixel value can be unambiguously mapped to a surface radiance value. The following question arises: What can be known about the surface normal and albedo given a set of radiance measurements with associated light vectors?

First, it might be easier to determine the answer to the following question: **What can be known about the surface normal and albedo from a *single* radiance measurement and light vector?**

Woodham [46] based the original formulation of photometric stereo on the idea of *reflectance maps*. A reflectance map $R(p,q)$ maps a specific surface orientation (p,q) to a surface radiance value. The surface is defined as $f(x,y)$, so the surface orientation can be described by the partial derivatives $p = \frac{\partial f(x,y)}{\partial x}$ and $q = \frac{\partial f(x,y)}{\partial y}$. The reflectance map combines the BRDF and light source vector into one function. Instead of using Woodham's original formulation, this section derives the photometric stereo method from the formulation of Lambertian surface reflection as a dot-product of the light vector and surface normal. The derivation is clearer using this formulation.

As defined in Section 2.1, the radiance from a Lambertian surface R can be calculated with

$$R = \rho \max(0, \vec{L} \cdot \hat{n}). \tag{2.6}$$

where ρ is the surface albedo, $\vec{L}$ is the light source vector, and $\hat{n}$ is the surface normal. Our earlier question becomes: What can be known about ρ and $\hat{n}$ given R and $\vec{L}$? The equation shows that when $R = 0$, either $\rho = 0$ or $\hat{n}$ is a vector that results in a dot product that is ≤ 0. Hence, when $R = 0$ very little can be determined about the surface.

Instead, consider the case where $R > 0$. With this constraint, the max function can be omitted since ρ is also assumed to be positive. The function can be further simplified by defining $\vec{N}$ as the product of the albedo and the surface normal

$$\vec{N} = \rho \hat{n}, \tag{2.7}$$

which results in the simplified definition of Lambertian reflection:

$$R = \vec{L} \cdot \vec{N}. \tag{2.8}$$

The dot product can finally be expanded to result in

$$R = L_x N_x + L_y N_y + L_z N_z. \tag{2.9}$$

This equation is very useful due to the simple linear relationship between the $\vec{N}$ components and the radiance R. One can start to see an answer to the earlier question. A known R value which is greater than 0 defines a constraint on the three components of $\vec{N}$. However it does not provide enough information to fully determine $\vec{N}$.

So in summary, the answer to the above question "What can be known about ρ and $\hat{n}$ given R and $\vec{L}$?", is:

- If $R = 0$, then either $\rho = 0$ or $\vec{L} \cdot \hat{n} \leq 0$.
- If $R > 0$, then $R = \vec{L} \cdot \vec{N}$.

Now we can try to answer the original question: **What can be known about the surface normal and albedo given a *set* of radiance measurements with associated light vectors?** Assuming that we had a second radiance measurement denoted by R_2 from the surface lit by a different light vector $\vec{L}_2$, would we be able to uniquely determine $\vec{N}$ then? The answer is no. Even with a second image, it would not be possible to uniquely determine $\vec{N}$. But once the number of images is increased to three, the situation changes. The next steps show why this is the case.

Every image adds an additional constraint on the surface normal. By denoting the image number with a subscript, we can write this as:

$$\begin{aligned} R_1 &= L_{1x}N_x + L_{1y}N_y + L_{1z}N_z, \\ R_2 &= L_{2x}N_x + L_{2y}N_y + L_{2z}N_z, \\ R_3 &= L_{3x}N_x + L_{3y}N_y + L_{3z}N_z, \end{aligned}$$

or using matrix algebra as,

$$\begin{bmatrix} R_1 \\ R_2 \\ R_3 \end{bmatrix} = \begin{bmatrix} L_{1x} & L_{1y} & L_{1z} \\ L_{2x} & L_{2y} & L_{2z} \\ L_{3x} & L_{3y} & L_{3z} \end{bmatrix} \begin{bmatrix} N_x \\ N_y \\ N_z \end{bmatrix}. \tag{2.10}$$

For an arbitrary number of images n, equation 2.10 becomes

$$\begin{bmatrix} R_1 \\ \vdots \\ R_n \end{bmatrix} = \begin{bmatrix} L_{1x} & L_{1y} & L_{1z} \\ \vdots & \vdots & \vdots \\ L_{nx} & L_{ny} & L_{nz} \end{bmatrix} \begin{bmatrix} N_x \\ N_y \\ N_z \end{bmatrix}. \tag{2.11}$$

Finally, if we denote the vector of radiance values by $\vec{R}$ and the matrix of light source vectors by $\mathbf{L}$, then equation 2.11 becomes

$$\vec{R} = \mathbf{L}\vec{N}. \tag{2.12}$$

Now, determining the vector $\vec{N}$ is a just matter of matrix algebra. Each image adds a row in the linear system of equations. With three images, $\vec{N}$ can be calculated with

$$\vec{N} = \mathbf{L}^{-1}\vec{R}. \tag{2.13}$$

From $\vec{N}$, the albedo ρ and normal $\hat{n}$ can be extracted with

$$\begin{aligned} \rho &= |\vec{N}| \\ \hat{n} &= \frac{\vec{N}}{\rho} \end{aligned}$$

Calculating ρ and $\hat{n}$ this way requires that $\mathbf{L}$ is invertible. For $\mathbf{L}$ to be invertible, the rows of $\mathbf{L}$ must be linearly independent. Three light vectors $\vec{L}_1$, $\vec{L}_2$ and $\vec{L}_3$ are linearly independent if $c_1\vec{L}_1 + c_2\vec{L}_2 + c_3\vec{L}_3 = 0$ implies that the scalars c_1, c_2 and c_3 are all zero. Geometrically, this means that the light vectors are required to be non-coplanar.

With more than 3 rows, the system can become over-constrained and a least-squares approach becomes most applicable to determine $\vec{N}$. This approach finds $\vec{N}$ such that $|\vec{R} - \mathbf{L}\vec{N}|$ is at a minimum. It is worth repeating that the equations here are only accurate for $R > 0$.

2.5.2 Extending and Modifying Traditional Photometric Stereo

Traditional PS makes many assumptions about the scene which are typically not met for real scenes. The presence of non-Lambertian surfaces, shadows, and interreflections all cause the PS method to perform poorly, and in some cases fail all together. This is no surprise since mathematically PS relies on a relatively simplistic model of the scene. Furthermore, the assumption that the illumination is a set of distant point light sources can be limiting.

Naturally, in the past two decades, many methods have been proposed to extend or modify the approach. They focus on considering non-Lambertian surfaces and less restricted lighting conditions. Only few methods have been proposed to handle shadows.

Non-Lambertian Surfaces

One of the earlier methods was proposed by Tagare and deFigueiredo [41] in 1991. It operates on a range of diffuse non-Lambertian surfaces under specific lighting conditions. Their approach is not general enough to account for all possible surfaces and illumination conditions, but they claim that it is a first step towards a theory of non-Lambertian PS. They also show that it is inappropriate to attempt recovery of non-Lambertian surfaces under the Lambertian assumption. Solomon and Ikeuchi [39] discuss a method for using four-source PS on objects with specular highlights. They also propose a method for determining the surface roughness based on a simple specular model.

Georghiades [16] proposes a method for recovering the shape and surface parameters for the Torrance-Sparrow model. This method can also be used on certain surfaces when the light source positions are unknown. Barsky and Petrou [1] propose a method to handle non-Lambertian surfaces and also consider shadows.

One of the most recent techniques is described by Mulligan and Brolly [34]. They implement a photometric stereo method which also considers non-Lambertian BRDFs. Their method, which is based on Magda's work [32], utilizes the inverse-square law for direct depth recovery. Finally,

Hertzmann and Seitz [23] have recently developed a method entitled "Photometric Stereo by Example." They use objects with known geometry with the same or similar BRDF as the object to be recovered. Their method handles arbitrary and spatially varying BRDFs and also performs a segmentation of different surface types under unknown lighting conditions.

Modified Illumination Assumptions

The traditional PS method assumes that all the light sources are distant point sources at known locations. Some of the methods mentioned above such as [16] and [23] operate under more general or unknown lighting conditions. There are also methods that operate with the original Lambertian surface assumption, but with less restricting lighting assumptions. Hayakawa [21] shows that it is possible to recover a Lambertian surface from a set of images lit by a light source with arbitrary motion. Basri and Jacobs [2] propose a method that uses a spherical harmonics based representation of the illumination. Due to the nature of Lambertian reflection, they show that surfaces can be recovered under more general illumination conditions than previous methods.

Belhumeur et al. [3] show that when viewing a height field lit by a distant point source under orthographic projection, the height field can be distorted using what they call the *bas-relief transformation* without affecting its appearance under specific lighting conditions. As a conclusion they state that photometric stereo with unknown light source directions is subject to an ambiguity that does not allow recovery beyond the bas-relief transformations. So if the lighting is unknown, photometric stereo under these conditions can not recover a unique height field without making some assumptions.

Shadows

As pointed out earlier, it is difficult to determine anything about the surface when the measured surface radiance is zero. So when a shadow is cast on an object either from another object or by itself, the shadowed region contains less information about the surface orientation than if it was lit by a light source. A simple method of approaching this problem is to simply ignore the shadowed regions and add additional light sources so that all regions of the object are lit by at least three light sources.

There is also the possibility to use shadows as an additional source of information. Schlüns [37] proposes a method to increase the performance of the traditional three-source PS method to more accurately recover the surface orientation in regions that are shadowed by one light source and lit by the two others.

2.6 Depth from Surface Orientation

The result of PS is a surface normal vector and albedo value at every pixel location in the image. The goal of the depth recovery step is to accurately recover a depth value at each pixel. The result is

a 2D map of the depth over the image domain. The surface orientation does however not uniquely specify the shape of the scene unless additional assumptions are made.

Most methods express the surface normals in form of a surface gradient and refer to their technique as a *depth from gradient* method [24, 14, 26]. They assume that the surface can be expressed as a function of x and y, where the $x-y$ plane is perpendicular to the optical axis of the camera. So the depth is $z = f(x, y)$ and the surface gradient is expressed as the pair of partial differentials

$$\begin{aligned} p &= \frac{\partial z}{\partial x}, \text{and} \\ q &= \frac{\partial z}{\partial y}. \end{aligned}$$

The surface gradient values are either recovered directly at each pixel by the photometric stereo algorithm, or they can be determined from the surface normals with:

$$\begin{aligned} p &= -n_x/n_z, \text{and} \\ q &= -n_y/n_z, \end{aligned}$$

where the surface normal is $\hat{n} = (n_x, n_y, n_z)^T$. With these mathematical definitions in place, the goal of this step can be expressed as follows: **Given either $\hat{n}$ or p and q at each pixel, obtain z at each pixel.**

There are some difficulties associated with expressing surface orientations as gradients, since at some locations n_z might be very small or zero. In an implementation this can cause numerical instability and division by zero. Furthermore, expressing the depth as $z = f(x, y)$ assumes that the image is projected with orthogonal projection, which is rare for real images.

2.6.1 Assumptions

One problem in recovering depth from surface orientation is that the surface normals do not necessarily dictate the relationship of the depth at two adjacent pixels. For example, one pixel might be on the edge of an object, with its adjacent pixel on a different object. The surface normal at either pixel does not provide any information about the relative depth between them. Even if the two adjacent pixels are on the same object, there is nothing that necessitates their depth values to be related. Only if regions of objects are assumed to be smooth can one start using the normals to determine the relationship between adjacent depth values. The normal map does not dictate the shape of these regions of smoothness, so a further assumption needs to be made for this as well.

The surface normals also do not provide an absolute depth value from the camera. An object could be large and far away, or small and close to the camera while the associated normal map is the same. This adds another assumption which is the depth of at least one point in each smooth region in the image.

These assumptions are not thoroughly discussed in literature although they are important for the application of PS. Most methods implicitly assume that the entire image is one smooth region.

Naturally they perform poorly when this assumption is not met.

Another problem with recovering depth from surface orientation is that there might not exist a depth map that matches the gradient field. In this case, the gradient is *non-integrable*, and an assumption needs to be made on how to best fit the given gradient values. Some methods add an *integrability constraint* for this purpose [14, 24].

2.6.2 Existing Methods

A range of different techniques exist. Most of them are either based on Horn's [24] or Frankot and Chellappa's [14] work.

Horn [24] proposes a method to minimize the difference between the given gradient data and the gradient field of the recovered surface using variational calculus. He does not however discuss the implementation of the method. A more comprehensive variational approach is presented by Terzopoulos [42]. He uses a multiresolution technique including smoothness constraints while also considering depth discontinuities. His paper also includes a discussion of the discretisation of his variational method. More recently, Horovitz and Kiryati [26] present a method similar to those of Horn and Terzopoulos, however add the ability to use control points. These control points permit fixing the scene to given depth values, counteracting the bias problems associated with photometric stereo.

Frankot and Chellappa [14] propose a method to enforce an integrability constraint, by performing the integration in the Fourier domain. Their method is fast in comparison to Horn's, however the constraints imposed often result in unsatisfactory results.

A good comparison of a wider range of techniques together with a new method will be published by Robles-Kelly and Hancock [36] this year. They implemented the most popular techniques and compare them on synthetic and real data.

2.7 Lighting Estimation

2.7.1 Overview of Lighting Estimation Methods

The estimation of the scene lighting is another important part of computer vision. Many shape reconstruction techniques rely on known lighting conditions, and only few attempt to recover both shape and lighting simultaneously. For this reason, in cases where lighting conditions are unknown, a separate lighting estimation step is required for many techniques to work. Lighting estimation is also useful for certain applications such as augmented reality. This thesis will not propose a new method or modify an existing technique, but merely evaluate a method as a demonstration of the developed controlled lighting setup.

Lighting estimation methods differ in the assumptions they make about the scene and about the lighting. Most early techniques assume the scene is lit by a single distant point light source. Methods

Table 2.1: A comparison of nine lighting estimation techniques.

	Year	Allows arbitrary geometry	Geometry can be unknown	Recovered lighting	Allows textured surfaces
Pentland [35]	1982	✓	✓	SDPS	
Weinshall [45]	1990	✓	✓	SDPS	
Yang & Yuille [49]	1991	✓	✓	SDPS	
Hougen & Ahuja [27]	1993	✓		Distribution	✓
Chojnacki et al. [9]	1994	✓	✓	SDPS	
Singh & Ahuja [38]	1998	✓		Distribution	
Zhang & Yang [51]	2001			MDPS	
Wang & Samaras [44]	2003	✓		MDPS	
Li, Lin, Lu & Shum [31]	2003	✓		MDPS	✓

proposed in the 1990s and later also consider multiple sources or a more general lighting distribution. Some methods assume that the scene geometry and surface properties are known (e.g. the scene contains a sphere with a Lambertian surface), while others attempt to operate on an unknown scene.

Table 2.1 compares the attributes of several techniques. The recovered lighting is categorized as either single distant point source (SDPS), multiple distant point sources (MDPS), or a distribution.

For demonstrating the use of the controlled lighting setup in evaluating lighting estimation methods, a technique which recovers light distributions is most appropriate. It can be used to show recovery of single point sources, multiple point sources and distributions. Hougen & Ahuja's method [27] recovers a distribution of point sources and is based on solving a least squares problem using the pixel intensities as constraints. Singh & Ahuja [38] present an iterative method for recovering a lighting distribution. Compared to Hougen & Ahuja, they demonstrate the recovery of a more dense distribution of sources and show a better evaluation of their results. For this reason Singh & Ahuja's technique is evaluated in this thesis. Its details are described in the following section.

2.7.2 Singh & Ahuja

This method [38] recovers a distant illumination distribution. It assumes that the observed surface is Lambertian with a constant albedo. A specific shape is not assumed, although the geometry of the object must be known. This technique is different from most others in that it is an iterative approach. It repeatedly estimates the lighting distribution by adjusting the previous estimate to fit within a set of constraints.

The algorithm behind this technique is not new. The POCS framework (Projection Onto Convex Sets) [7] is used for restoring signals that were distorted by noise. It is also applied to image restoration. The similarity between image recovery and illumination estimation is that both can be formulated as a deconvolution problem.

The lighting distribution is described with the function $L(\theta_s, \phi_s)$. The image is represented by

$I(i,j)$. A simple model of the image formation process is constructed by discretising the domain of L. The two spherical coordinate parameters are replaced by integers i and j, and the distribution over this discrete domain is referred to as $\tilde{L}$. Note that this form of sampling on the sphere is not uniform. Towards the two poles, the sampling rate becomes higher. Singh & Ahuja do not mention whether a more uniform sampling approach would have any benefits. A convolution kernel $h(i,j,k,l)$ which includes the BRDF and geometric factors is formulated to further simplify the equation describing the image formation. Finally, the image can be written as

$$I(i,j) = \sum_{(k,l)} h(i,j,k,l)\tilde{L}(k,l). \tag{2.14}$$

The basic concept of POCS is to define a set of constraints C that apply to the function to be recovered. In the case of lighting recovery, two constraints are that the recovered lighting is non-negative, and that the recovered lighting produces the same image as the original lighting. More specifically, in this paper, a constraint $C_{i,j}$ is defined for each pixel (i,j) of the image. A projection operator P is derived for each constraint. This operator, when applied to an estimate, will alter the estimate enough for the associated constraint to be satisfied. If all projection operators could be applied at once the recovered estimate would meet all the constraints. But since only one projection is applied at once, each projection might cause previous constraints to no longer be satisfied. For this reason it is necessary to iterate.

For each iteration step, all of the projection operators are applied. This can be written as

$$\tilde{L}_{k+1}(i,j) = P_A P_{N_1,N_2} P_{N_1,N_2-1} \cdots P_{1,2} P_{1,1}(\tilde{L}_k(i,j)), \tag{2.15}$$

where P_A is the amplitude projection and $P_{i,j}$ is the residual projection for each pixel (i,j).

The amplitude constraint simply constrains the illumination to a range from 0 to A:

$$C_A = \{L(i,j) : 0 \le L(i,j) \le A \quad \forall(i,j)\}. \tag{2.16}$$

The associated projection operator P_A is

$$P_A[x(k,l)] = \begin{cases} 0: & x(k,l) < 0 \\ x(k,l): & 0 \le x(k,l) \le A \\ A: & A < x(k,l) \end{cases}. \tag{2.17}$$

For this constraint it is easy to see how the projection operators function. Values that are below the acceptable range are moved to the lower limit, and values above the range are reduced to the upper limit A.

The residual constraint is a little more involved. Here, for every pixel, the intensity generated from the estimated lighting is compared to the original intensity value. A maximum residual magnitude δ_o is allowed, although in experiments it is set to 0. The constraint is expressed as

$$C_{i,j} = \{L(k,l) : |I(i,j) - \sum_{(k,l)} h(i,j,k,l)L(k,l)| < \delta_o)\}, \tag{2.18}$$

and the associated projection operator is defined as

$$P_{i,j}[x(k,l)] = \begin{cases} x(k,l) + \dfrac{d^x(i,j) - \delta_o}{\sum_m \sum_n h^2(i,j,m,n)} h(i,j,m,n) : & d^x(i,j) > \delta_o \\ x(i,j) : & |d^x(i,j)| \leq \delta_o \\ x(k,l) + \dfrac{d^x(i,j) - \delta_o}{\sum_m \sum_n h^2(i,j,m,n)} h(i,j,m,n) : & d^x(i,j) < -\delta_o \end{cases} \tag{2.19}$$

where

$$d^x(i,j) = I(i,j) - \sum_{(k,l)} h(i,j,k,l)x(k,l). \tag{2.20}$$

It is noted that the result of the POCS algorithm depends on the initial guess of the distribution. The algorithm converges to the feasible solution closest to the initial guess.

Singh & Ahuja implemented the POCS approach and ran a few simulations using the framework. The results they present show multiple circular and rectangular light sources, with up to two in each case. The recovered distribution is a slightly blurred version of the original lighting (for 50 iterations). They note that the recovered lighting features artifacts when the residual projections are applied in order. By applying these projections in a random order, the artifacts were almost completely removed and the rate of convergence was greatly improved.

Chapter 3

Using a Raster Display Device for Controlled Illumination

This chapter describes an apparatus for controlled illumination by a display device. The general aspects of using a display device as a light source are discussed, and the details of the experimental setup are provided. A mathematical model is described in Section 3.3 and calibration procedures are discussed in Section 3.4.

3.1 Display Devices as Light Sources

Currently, the most common raster display devices are *CRT monitors, LCD screens* and *LCD projectors. Plasma screens* also qualify as raster display devices, yet their current popularity is limited. This chapter is meant to provide a theoretical foundation that can be applied to any of these devices, although the focus will be on LCD screens as it is the device investigated experimentally.

Each of these devices has different characteristics. A CRT monitor contains a "cathode ray tube" in which three electron beams are used to light up phosphors on the display surface. The light from LCD displays originates from a backlight which is covered by a layer of liquid crystals between two polarization filters. The liquid crystals are controlled to rotate the polarized light and thereby adjust the amount of light passing through that layer. The screen is divided up into *pixels* which are again divided up into red, green and blue cells (or subpixels). Each are controlled individually to adjust the perceived colour. LCD projectors are based on a similar principle as LCD screens, except that the light is projected through the liquid crystal layer onto a screen. Lenses are used to focus the light such that the projected image is focused on the screen plane. Naturally, the LCD projector is similar to the other devices only when combined with the screen.

What all these devices have in common is that they can be modeled as a grid of light sources. Assuming that the screen is flat, all light sources are within the same plane. At each pixel, the radiance from the screen can be controlled individually. The fact that each pixel is actually subdivided into red, green and blue cells is ignored in this thesis. For sake of simplicity, the pixels are treated as

a single light source with no particular wavelength. In reality, the light from each pixel spans a range of wavelengths. Considering this fact and utilizing the individual control of each cell is expected to be beneficial, yet it is not explored here.

It would be wrong to assume that the light radiance from the pixels of any raster display device is constant in all directions. For example, most LCD screens appear brighter when viewed from the centre than when viewed at an angle. The radiance-direction dependency is an effect of the physical construction of the device. Light sources that do not emit light equally in all directions are referred to as *directional* and the particular behaviour is termed *directionality*. This term is adopted here to apply to display devices as well. This chapter examines how to determine the directionality properties of LCD screens.

LCD screens have certain features and limitations that need to be considered when using them as light sources. First, the intensity of the backlight is typically not controllable. It constantly emits light, and only the liquid crystal layer can be controlled to attenuate that light. The layer can however not completely block all light, so even when a pixel is set to "black," a limited amount of light will still be emitted from the screen. LCD monitors typically list a *contrast ratio* among their features. This value is the ratio between the highest and lowest radiance the screen can emit. The brightness of the backlight does not affect this ratio since the radiance of a black pixel is affected in proportion to the radiance of a white pixel. Contrast ratios range from around 500:1 for low-grade consumer monitors to over 1500:1 for industrial screens usable in sunlight. In any case, it should be noted that even when the entire screen is set to black, the LCD screen will still emit some light.

If one wants to examine the lighting from only a few pixels on the screen, it is necessary to compensate for the additional light from all other pixels which are set to black. This can be achieved by taking one image of the scene lit with the entire screen set to black, and a second image with the desired pixels on. Then, by subtracting the first image from the second, the contributions of the desired pixels on the scene can be determined. This also eliminates potential contributions from ambient lighting of the scene. It does however require that the ambient lighting conditions do not change from the first image to the second.

3.2 Experimental Setup

The setup used for experimentation consists of a mid-grade consumer LCD monitor, and a high-end consumer digital SLR camera. The apparatus is enclosed by black cloth to reduce the amount of ambient light entering the scene. A diagram of the setup is shown in Figure 3.1, and a picture of the real apparatus is provided in Figure 3.2.

A NEC MultiSync LCD 1760NX monitor was used for all the experiments. It is a 17-inch active matrix, thin film transistor (TFT) display with a native resolution of 1280 x 1024. The listed white luminance is 260 cd/m^2 and the contrast ratio is 450:1. The camera is a Canon EOS 300D (Digital Rebel). It includes a 22.7 x 15.1 mm CMOS sensor with 6.3 million effective pixels. The total sensor

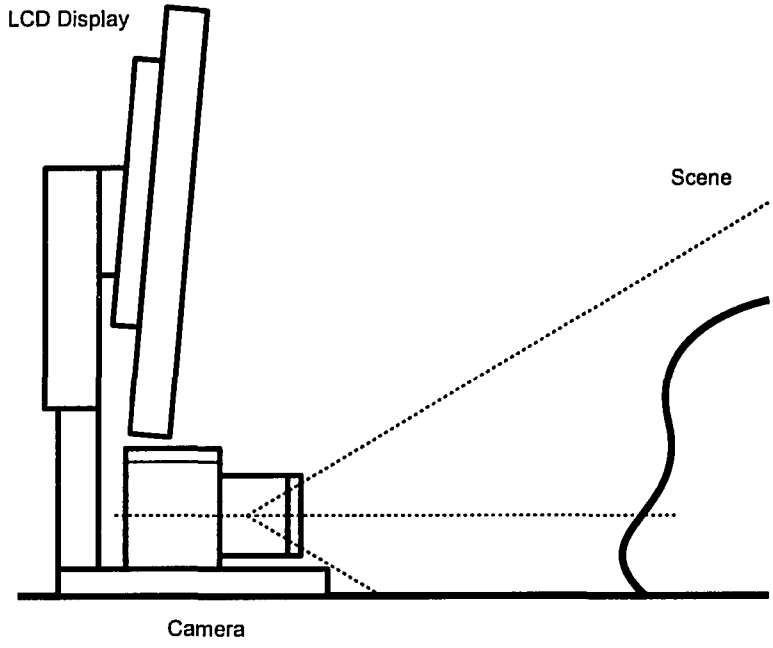

Figure 3.1: A diagram of the experimental setup.

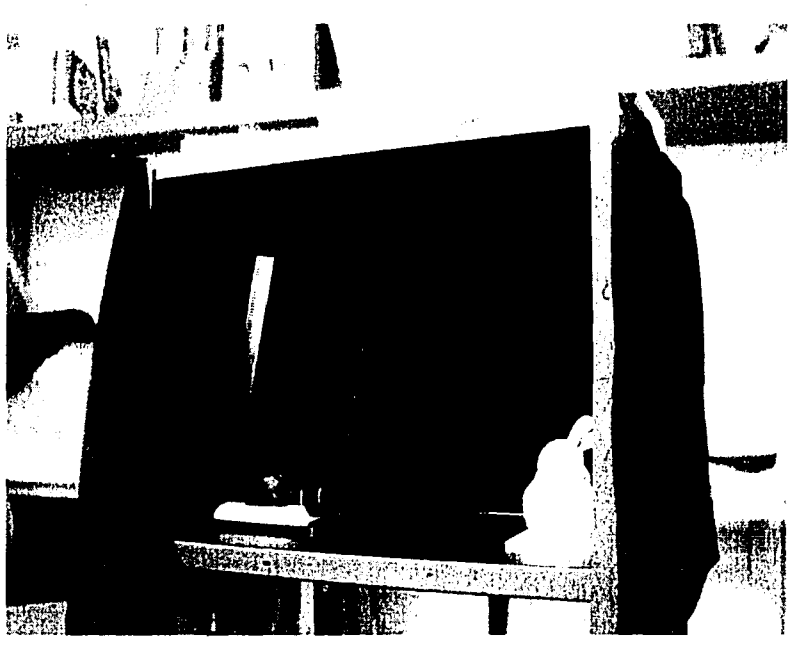

Figure 3.2: The experimental setup. Visible in this picture are the enclosure, the screen, the camera (below the screen) and the captured object.

resolution is 3152 x 2068 and the effective resolution is 3072 x 2048. Finally, the lens is a Canon EF-S18-55mm f/3.5-5.6 zoom lens with focal length of 18mm to 55mm, a maximum aperture of f/3.5-5.6, and a minimum aperture of f/22-36. Detailed specifications of the display, camera, and lens can be found in Appendix 1.

3.3 Mathematical Model

This section defines a model of the scene illumination as well as a model of the camera imaging process. These models can be applied to any specific use of the controlled illumination device. The scene illumination model defines the irradiance at a point in the scene given a specific lighting configuration on the display device. The camera model allows determining the irradiance from the scene given an image.

The path of the light from the screen to the camera is modeled as a five step process:

1. Screen directionality → screen radiance
2. Inverse square drop-off to object → scene irradiance
3. Interaction with scene → scene radiance
4. Surface radiance → sensor irradiance
5. Camera response → pixel values

In this section, steps 1, 2, 4 and 5 are described. Step 3, the interaction with the surface is dependent on the assumptions made about the scene. These assumptions are application specific and vary depending on how the apparatus is employed.

The derivations assume that all points are expressed in the camera's coordinate system. This requires a calibration of the screen position and orientation with respect to the camera, which is discussed in Section 3.4.2.

3.3.1 Determining the Scene Irradiance

Since the screen radiance at each pixel i is assumed not to be the same in all directions, it is modeled as an arbitrary function over a hemisphere $R_P(\theta, \phi)$ and is expressed as

$$R_P(\theta, \phi) = R_{unatten} a_i f(\theta, \phi), \tag{3.1}$$

where $R_{unatten}$ is the unattenuated radiance from the backlight, a_i is the pixel attenuation factor, and $f(\theta, \phi)$ is the *directionality function*. The pixel attenuation factor ranges from 0 to 1 depending on the pixel value. The association between the pixel value and the attenuation factor will not be discussed here since only factors of 0 or 1 will be considered. The directionality function $f(\theta, \phi)$ is

determined through a calibration procedure discussed in Section 3.4.3. From here on, it is assumed that $f(\theta, \phi)$ is known.

By considering the inverse square law, the irradiance on a scene point from a single pixel i can be modeled as

$$I_S = \frac{R_P(\theta_i, \phi_i)}{r_i^2}, \tag{3.2}$$

where r_i is the distance of the point to the pixel. If the scene point is located at $S = (S_x, S_y, S_z)^T$ and the pixel is centred at $P_C = (P_{Cx}, P_{Cy}, P_{Cz})^T$, then $r = |S - P_C|$. The angles θ_i and ϕ_i depend on the location of the scene point and can be calculated from the screen orientation.

Some assumptions are necessary for the above formulation:

- The radiance from the backlight $R_{unatten}$ is independent of the pixel location
- The directionality function $f(\theta, \phi)$ is independent of the pixel location
- The location of pixels with respect to the camera can be determined
- Pixels are point light sources

The directionality function and the relationship of pixel coordinates to the camera coordinate system are calibrated according to sections Section 3.4.3 and Section 3.4.2 respectively.

3.3.2 From Scene Radiance to Pixel Values

For the vision techniques implemented in this thesis, it is necessary to know the relationship between the scene radiance and the pixel values. The pixel values are read from the image files of the digital camera. Inferring the associated radiance from the scene is not as simple as a direct linear mapping. As Section 2.3.2 outlines, cameras do not all produce a linear mapping between the sensor irradiance and the pixel value.

In this thesis, it is assumed that the sensor irradiance is proportional to the scene radiance. Furthermore, the factor relating these two values is assumed to be constant over the entire image. For lenses with strong vignetting effects this assumption should not be made.

3.4 Calibration

As mentioned earlier, the computer vision methods implemented require knowing the screen position, the directionality function $f(\theta, \phi)$ and the radiometric response of the camera. The calibration procedures for screen position and directionality require that the camera is first calibrated geometrically and radiometrically. Geometric calibration of the camera is performed using the MATLAB toolbox [5]. Through this process, the intrinsic parameters of the camera are obtained as described in Section 2.3.1. Using these parameters, the toolbox provides means to map pixels to rays in the scene and scene points to pixels in the image.

3.4.1 Radiometric Camera Calibration

Radiometric camera calibration is performed to determine the mapping of image irradiance to pixel values. For the camera that was used for the experiments, the raw files contain the digitized values of the sensor output. With the camera software or API, one can convert the raw file using a *normal* or *linear* setting for the response curve. This process is termed *development* similar to the development of film negatives into prints. The linear development does not manipulate the data from the raw file other than converting from 12-bit to 8-bit representation. The normal development applies a response curve in the same way the camera does internally when storing images as a compressed JPEG file.

Using the HDR shop software [10], the response curves were recovered for both normal and linear development modes. Note that the *response curve* which is recovered combines all the responses of the camera and the development process, including the sensor response and the development response. Multiple curves were recovered for each mode and averaged. It was noted that the results from HDR shop varied depending on the set of input images. Significantly more variation was noted for the normal development mode. It is possible that this development mode does not apply exactly the same response curve to all images. In that case recovering a single curve from sets of images would only be an approximation of the camera's response.

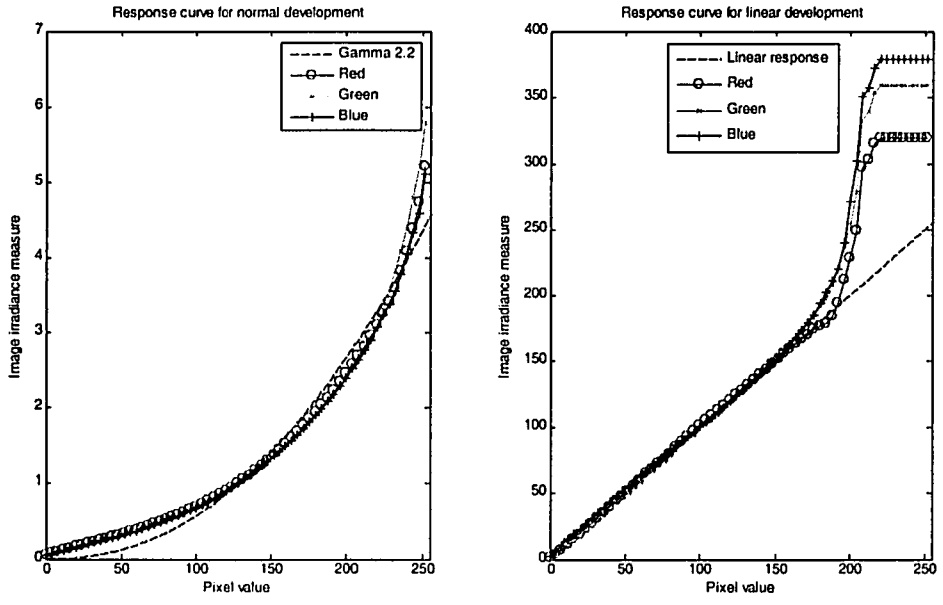

Figure 3.3: The average camera response functions for images developed using the "normal" and "linear" development methods. Note that the axes are swapped in comparison to those in Section 2.3.2

Figure 3.3 shows the averaged response functions for the two development modes. Note that the

y-axis is only an *image irradiance measure* which is proportional to the image irradiance, but with an unknown factor. Each individual curve is scaled to pass through the same point at a pixel value of 128 since an assumption needs to be made to relate their irradiance values to each other.

From the figure, it can be noted that the response under *normal* development is not exactly a gamma of 2.2 (or a gamma of 1/2.2 with the axes swapped). Camera manufacturers select a response curve they believe is suitable, so this deviation is expected. The response for the linear development setting shows several interesting features. One is that for pixel values above 160, the response becomes non-linear. This is expected due to sensor saturation. At over 210, all pixel values are mapped to the same irradiance value. This is a result of how the software develops images in linear mode. Even with saturated pixels present, the peak intensity in the linearly developed image does not reach 255. So the calibration software has no data to work with in this region.

Instead of having to correct for the camera response, all experiments are conducted using the linear development mode from raw files. It is ensured that pixel values are not higher than 160 so that the measurements remain in the linear response region of the sensor.

3.4.2 Screen Position Calibration

The goal of the screen position calibration is to obtain a 4×4 matrix that relates the pixel coordinates to 3D coordinates in the camera reference frame. It is assumed that the screen is flat and the pixels are square. If the screen was directly visible from the camera, the calibration could be accomplished by displaying a calibration pattern on the screen at a known position. But since the screen is not visible, the calibration is a little more challenging.

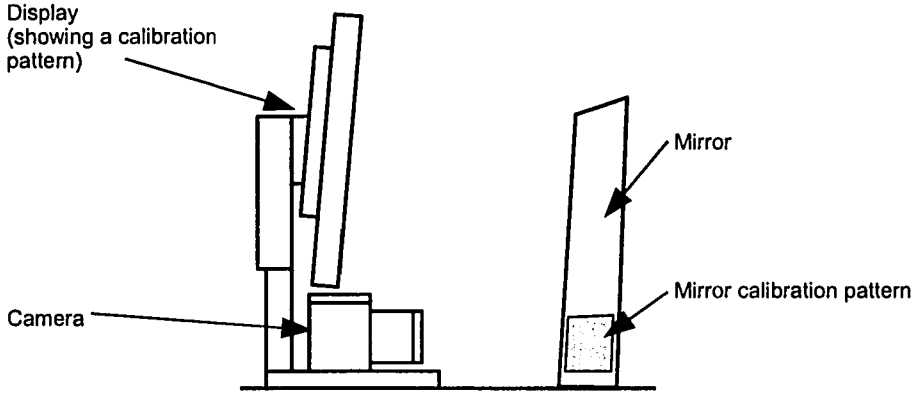

Figure 3.4: The screen position calibration setup.

To make the screen visible from the camera, a mirror is used in the developed calibration setup as shown in Figure 3.4. One calibration pattern is attached to the surface of the mirror and a second pattern is displayed on the screen. Images of the mirror are captured such that the screen calibration

pattern is visible as a reflection in the mirror. An example of such an image is provided in Figure 3.5.

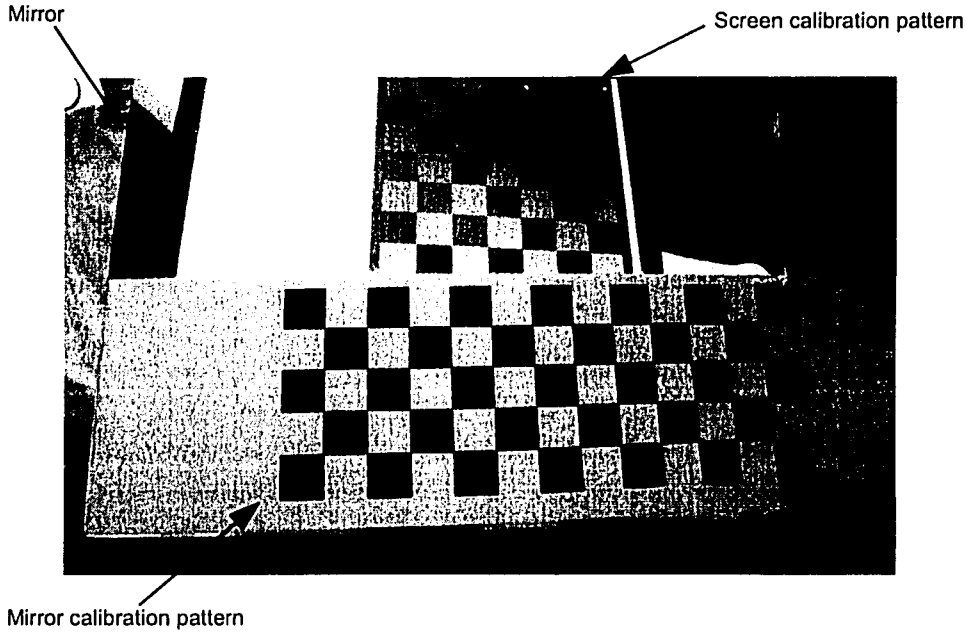

Figure 3.5: An example image used for the calibration of the screen position. Note that the mirror fills nearly the entire image except the left and bottom edges.

The following derivation assumes that points are represented by column vectors and are transformed from one reference frame to another by left multiplication. A transformation matrix from the reference frame A to the reference frame B is denoted by T_{BA} such that two consecutive transformations can be written as $T_{CB}T_{BA}$ to transform from the reference frame A to the reference frame C.

With a calibrated camera, the position and orientation of each of the calibration patterns can be determined. They are both 4×4 transformation matrices which can transform points in the reference frame of the calibration pattern to the camera coordinate system. The matrix T_{CM} is the *mirror transformation matrix*, and $T_{CS'}$ is the *mirrored screen transformation matrix*. Both are shown in Figure 3.6 with thick black arrows. A subscript S' is used to denote the mirrored screen while a subscript S refers to the real screen.

The derivation is broken down into two steps. The first step determines the matrix T_{CS} which relates the location of the real screen calibration pattern to the camera reference frame. The second step relates the screen pixel coordinates to the calibration pattern. With this, pixel coordinates can then be transformed to camera coordinates.

To obtain T_{CS}, it is necessary to determine the transformation matrix relating the mirrored screen

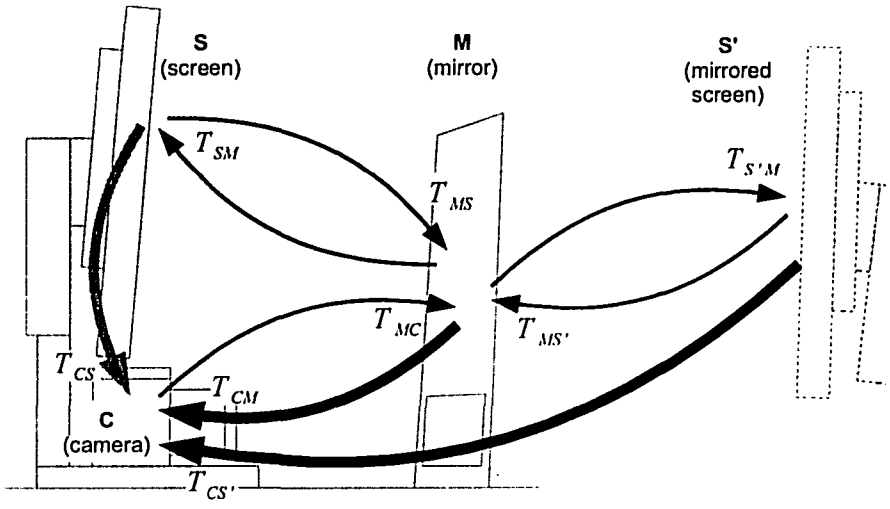

Figure 3.6: A diagram of the important transformation matrices between the four reference frames C, M, S and S'. The two known transformations $T_{CS'}$ and T_{CM} are highlighted with thick black arrows. The desired unknown transformation T_{CS} is highlighted with a thick gray arrow.

to the mirror. The mirrored screen matrix $T_{CS'}$ can be broken up as

$$T_{CS'} = T_{CM} T_{MS'}, \tag{3.3}$$

where T_{CM} is the aforementioned mirror matrix, and the $T_{MS'}$ matrix transforms from the mirrored screen reference frame to the mirror reference frame. By multiplying the previous equation by T_{CM}^{-1} one can obtain the equation

$$T_{MS'} = T_{CM}^{-1} T_{CS'}. \tag{3.4}$$

To get the real screen to mirror transformation matrix T_{MS}, the $T_{MS'}$ transformation is mirrored. The mirror surface is the $x - y$ plane in the local coordinate system of the mirror. So the mirroring can be accomplished by inverting the sign of the z component. This is easiest done by multiplying by

$$M = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{3.5}$$

Having defined M, the transformation of the real screen reference frame to the mirror is

$$T_{MS} = M T_{MS'} = M T_{CM}^{-1} T_{CS'}. \tag{3.6}$$

Now, the desired transformation matrix of the real screen to the camera reference frame can be defined as

$$T_{CS} = T_{CM} T_{MS} = T_{CM} M T_{CM}^{-1} T_{CS'}. \tag{3.7}$$

This equation allows one to transform from the reference frame of the screen calibration pattern to the reference frame of the camera. The reference frame of the calibration pattern is however not equivalent to the coordinates used for pixel positions. It is necessary to convert the pixel coordinates to physical dimensions by measuring the size of the pixels. Also, as Figure 3.7 shows, the origin of the calibration pattern coordinate system is not the same as that of the pixel coordinate system. So converting from pixel coordinates to calibration pattern coordinates involves both scaling and translation.

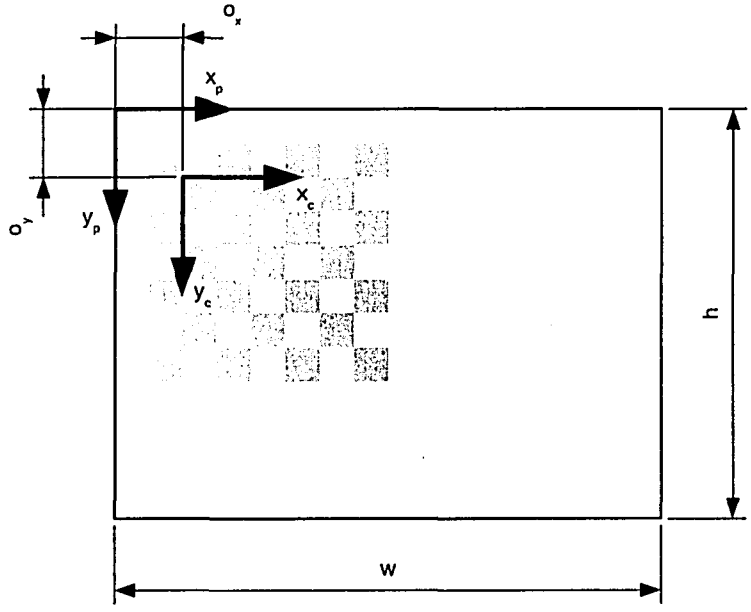

Figure 3.7: The coordinate systems of the screen and the calibration pattern displayed on the screen.

The translation between the two coordinate systems can be accomplished with the matrix

$$T_T = \begin{bmatrix} 1 & 0 & 0 & -o_x \\ 0 & 1 & 0 & -o_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3.8}$$

where o_x and o_y are the pixel coordinates of the calibration pattern origin. The scaling from pixel coordinates to real world coordinates is achieved by multiplying with the scaling matrix

$$T_S = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3.9}$$

where s_x is the ratio of the physical width of the display w to the number of pixels in the x direction, and s_y is the ratio of the display height h to the number of pixels in the y direction. Using the two

matrices T_T and T_S, a transformation matrix from pixel coordinates to camera coordinates can be constructed as

$$T_{CP} = T_{CS} T_S T_T. \tag{3.10}$$

Now, a pixel with coordinates P_P can be transformed to the camera reference frame with

$$P_C = T_{CP} P_P, \tag{3.11}$$

where P_C is the pixel's location in the camera reference frame. P_C and P_P are both 4x1 vectors in homogeneous coordinates. The z component of the P_P vector is zero since the pixels are located on the $x - y$ plane.

In the calibration setup a *first surface mirror* was used since it has advantages over the more commonly available *back silvered mirrors*. A back silvered mirror reflects light from both the front glass surface and from the back silvered surface. Light can even be reflected within the glass causing multiple reflections to occur. With a first surface mirror, only a single reflection of the light takes place, resulting in a clearer mirror image.

The accuracy of this calibration procedure is limited by the accuracy of the intrinsic camera calibration. For example an error in the calculated focal length will affect the extrinsic calibration of the two input matrices T_{CS} and T_{CM}. A significant amount of work would be required to determine the resulting error in the T_{CP} matrix given the error of the intrinsic parameters. However since the calibration accuracy affects the accuracy of the applications directly, further investigation is recommended for future work.

This section has shown how the location of the screen can be calibrated with respect to the camera using a mirror with an attached calibration pattern. The transformation matrices obtained in the calibration procedure are manipulated to obtain the matrix T_{CP}, allowing the transformation of pixel coordinates to camera coordinates.

3.4.3 Screen Directionality Calibration

This calibration step aims to determine the dependency of the display radiance on the viewing direction. In Section 3.3.1 this dependency is modeled as the directionality function over a sphere $f(\phi, \theta)$. A basic assumption made is that the directionality function is spatial invariant on the display.

The calibration procedure outlined here also assumes that the directionality function varies only with the angle from the screen normal ϕ (the colatitude). The angle about the screen normal θ (the azimuth) is assumed to not affect the radiance.

Under these assumptions the radiance can be measured by taking images of the screen from different angles. To accomplish this, the camera is directed at the centre of the screen, which is placed on a turntable. The axis of the turntable is in the same plane as the screen, and passes through the centre of the screen. A set of images is taken with the camera in full manual mode, to ensure that each of the images is exposed equally. The screen is rotated such that one image is taken

at each 10° increment from $\phi = -80°$ to $\phi = 80°$. An average is calculated from the measurements on the left and right sides.

The measured irradiance is then scaled such that the peak measurement becomes 1. This is possible since only a relative measurement is required. Linear interpolation is used to determine values of $f(\phi, 0)$ where it is not directly sampled. Figure 3.8 shows a plot of the calibrated directionality function.

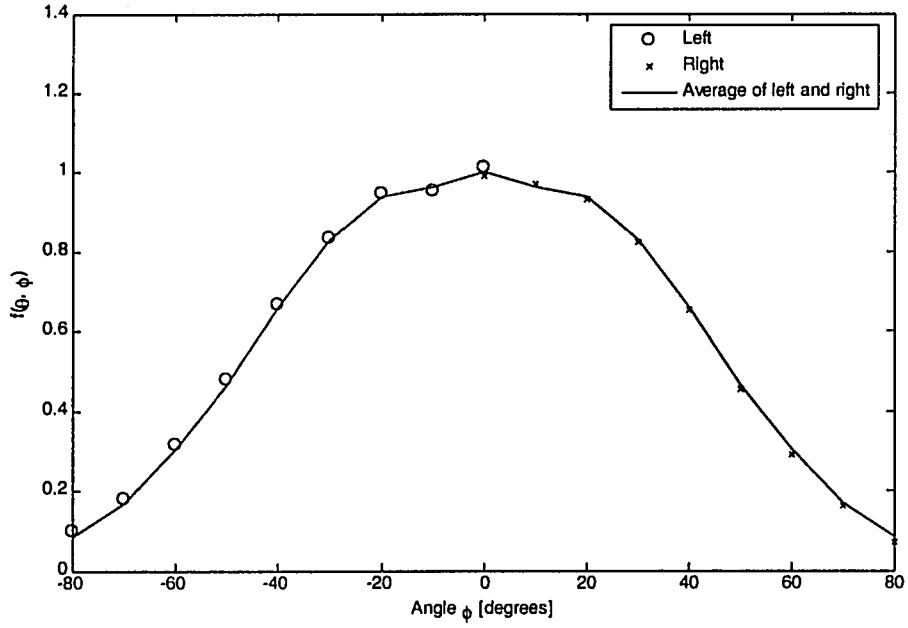

Figure 3.8: The directionality function $f(\phi, 0)$ with respect to ϕ. The function is assumed to be constant over 0.

3.5 Capturing Images

One of the main challenges in capturing images lit by an LCD screen is that the amount of light emitted by the screen is low in comparison to most other light sources. Camera sensors are limited in their sensitivity, and sensor noise becomes considerable in comparison to the magnitude of the measured irradiances.

Through experimentation, the following items were determined to have the most significant effect on increasing the measurement accuracy:

- **Downsampling the image** reduces the amount of noise while loosing spatial detail. Since sensor noise is typically pixel-independent, averaging multiple pixels is very effective for reducing noise. Naturally this comes at the cost of losing image detail, so an appropriate

balance needs to be determined.

- **Reducing ambient light** by enclosing the apparatus with cloth or cardboard minimizes the amount of light entering the camera. The camera can be set to longer exposure times without over-exposing the image.
- **Increasing the exposure time** reduces image noise due to an increased signal strength on the sensor. The drawback is the increased total capture time and the necessity to keep the captured object still during the capture process.
- **A large aperture** also increases the signal strength on the sensor by letting more light on the sensor. The downside is a reduced depth of field.

The specific parameter values depend primarily on the brightness of the display and the sensitivity of the camera. For the camera and display used in the experiments, it was necessary to enclose the setup. Images were exposed at the maximum exposure time of the camera which is 30 seconds. Acceptable noise levels could also be achieved with exposure times around 10 seconds but since minimizing the capture time was not considered crucial, the maximum exposure time was found most appropriate. The aperture was opened to its maximum of f/3.5 at 18mm focal length and f/5.6 for a focal length of 55mm. All images were downsampled by a factor ranging from 5 to 10 depending on the desired resolution for analysis. These parameters result in nearly unnoticeable image noise in the downsampled images. Depending on the application, more image noise may be tolerable, so a wider range of parameter values becomes acceptable.

3.6 Synthetic Image Generation

To allow controlled analysis of vision methods without the errors associated with real experiments, a synthetic image generation framework was implemented. It generates images lit by a simulated controlled lighting environment according to the model in Section 3.3. This includes the calibrated directionality function, inverse square law, an ideal pinhole camera corresponding to the calibrated real camera as well as the screen position and orientation from the calibrated setup.

The most significant factors that are controlled through this framework are:

- The **surface reflection** can be adjusted to an arbitrary BRDF.
- The **camera model** is an ideal pinhole model, causing the entire scene to be in focus.
- There is no effect of **sensor noise** on the generated images.
- The **light sources** are finite point sources.
- The **directionality function** of the sources correlates exactly to the calibrated model.
- The **screen position** is as calibrated from a real scene.

3.7 Summary

An apparatus for controlled illumination together with procedures necessary for its calibration has been presented. This setup can be used as a tool in the analysis and development of computer vision techniques as the following chapters demonstrate. Using the mathematical model presented, the contribution of individual screen pixels on the scene irradiance can be calculated. Through radiometric camera calibration, the mapping of pixel values to scene radiance values is established. The model requires that the screen pose is known and that the screen directionality function is available. These two requirements are met through the calibration procedures presented.

The main challenge identified is the low light radiance from the LCD screen. Due to the limited sensitivity of cameras, it is necessary to take measures ensuring the accuracy of the scene radiance measurements. Primarily, reducing the ambient lighting through enclosing the apparatus is found to be beneficial. Furthermore, downsampling of the captured images, use of long exposure times, and a wide camera aperture were determined to be effective in reducing the image noise and therefore increasing the radiance measurement accuracy.

With the synthetic image generation framework implemented, a simulation of the controlled lighting apparatus can be used to generate images without the image noise and other factors associated with capturing real images. This allows the analysis of algorithms in a controlled environment and provides a method of determining their performance under ideal conditions.

Chapter 4

Shape Recovery

To show how the controlled illumination setup can be used for shape recovery, and to test its applicability for this purpose, this chapter discusses the implementation of two methods. This entire chapter except for Section 4.6 focuses on the first method which is a *two-step* approach using photometric stereo to recover surface normals and a separate step to determine the depth from the recovered normals. The second shape recovery method operates on the same input images as the first. It recovers the depth directly in a *single step* and is hence termed the *direct depth recovery* method.

As mentioned in the background chapter, the original photometric stereo method is relatively old and well established. Its simplicity in comparison to the newer modified methods is the key reason for why it was implemented. It allows a clearer analysis of the strengths and weaknesses of the controlled illumination setup. The trade-off is a limited applicability to the surface types to which it can be applied.

Photometric stereo assumes that the light source illuminating each image is an infinitely distant point light source. With common raster displays, it is not possible to achieve exactly the same effect as a distant point source. So instead, a close approximation is used. Point sources are simulated by displaying small white squares on the screen. A single pixel most closely approximates a finite point source, however it is not bright enough to allow accurate measurements of its effects on the image. For this reason, a group of pixels need to be used to act as a single light source. Increasing the size of the square acting as light source increases the accuracy of the measured image irradiance contribution, but it also further deviates from the assumption that the light source is a distant point source. A square size of 50x50 pixels was selected as it was the minimum size that provided sufficient light.

The implementation uses a total of 6 squares. The number of sources and their positions were chosen with the following factors in mind:

- For the recovery of the surface normal and albedo at a point, at least three light sources need to contribute to the irradiance at that point. Certain areas of objects can not be lit by every light source due to self-shadowing. Hence, more than three sources are required for most objects.
- To recover the local surface normal at points lit by only three sources, the light vectors must

be non-coplanar as mentioned in Section 2.5.1. For sources on the screen this means that if any three sources are collinear, they alone are not sufficient to recover a surface normal.

- Increasing the angular separation between sources increases the numerical stability of the method if the angle is below 90°.
- Decreasing the number of sources decreases the total capture time.

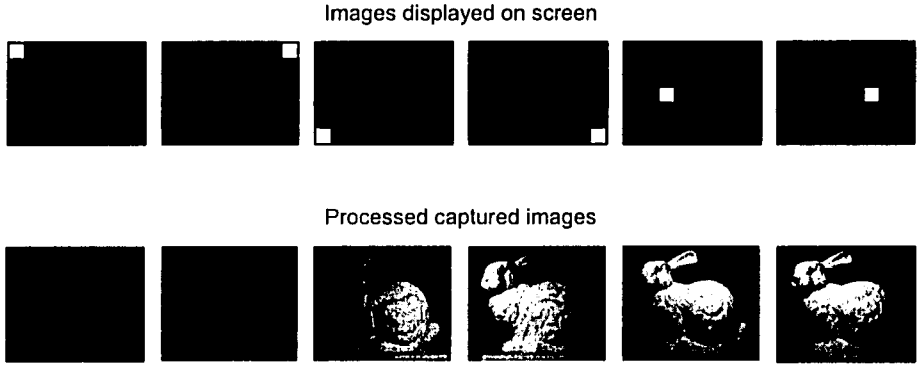

Figure 4.1: An example of the images displayed on the screen with their associated processed images captured by the camera. The size of the light sources is exaggerated for visibility. The processed images are generated by subtracting an image lit by only ambient and a "black" screen from the image lit by a light source.

Figure 4.1 shows an example set of images captured under associated lighting conditions. The processed images serve as the input for the photometric stereo method. One can note that the first two images are darker than the others. This is an effect of both the inverse square law and the directionality of the screen. The light sources from the top edge of the screen are more distant from the object than the bottom ones, and the angle from the screen normal is greater. Both factors reduce the irradiance on the surface.

Using square areas as light sources on a screen that is near the captured object, rather than using distant point sources introduces the following problems:

1. The inverse square law needs to be considered.
2. The light direction is not the same at every point on the object.
3. At certain points on the object, the light sources can be *partially occluded* through self-occlusion.

The first two issues are handled as described in the following section. Considering the partial occlusion of sources would add significant complexity to the shape recovery method. It is not explicitly considered in the developed method and should therefore be noted as a source of error in the results.

The shape recovery procedure requires calibration of the lighting apparatus prior to analysis. After calibration of the setup, the following steps are necessary:

1. Image capture
2. Region of interest (ROI) selection
3. Parameter selection
4. Image processing
5. Shape recovery

Steps 3, 4 and 5 are typically repeated to optimize the recovery results. The image processing step includes the downsampling of the captured images and subtraction of the image captured lit by the "black" screen.

The next sections discuss the first implemented photometric stereo method and the technique used to recover depth from surface orientation. Section 4.3 describes how the results can be improved through an iterative approach. An experimental analysis of the method on synthetic images is provided in Section 4.4. Analysis of the performance on real images is presented in Section 4.5. Finally, the direct depth recovery method is discussed in Section 4.6.

4.1 Photometric Stereo

Photometric stereo requires the surface irradiance to be known for each light source. However, with the inverse square law and directionality effects, the irradiance is dependent on the position of the surface. So the original photometric stereo formulation can not be applied unless some assumptions are made. Recall equation 3.2

$$I_S = \frac{R_P(\theta_i, \phi_i)}{r_i^2},$$

where I_S is the scene irradiance contribution of a single pixel at a distance r_i from the scene point. Substituting equation 3.1 for $R_P(\theta_i, \phi_i)$ gives

$$I_S = \frac{R_{unatten} a_i f(\theta_i, \phi_i)}{r_i^2}, \tag{4.1}$$

where $R_{unatten}$ is the unattenuated radiance from the backlight, a_i is the pixel attenuation factor, and $f(\theta, \phi)$ is the directionality function. The directionality function $f(\theta_i, \phi_i)$ is dependent on the direction from the pixel to the scene point.

To recover the surface normal, recall equation 2.11

$$\begin{bmatrix} R_1 \\ \vdots \\ R_n \end{bmatrix} = \begin{bmatrix} L_{1x} & L_{1y} & L_{1z} \\ \vdots & \vdots & \vdots \\ L_{nx} & L_{ny} & L_{nz} \end{bmatrix} \begin{bmatrix} N_x \\ N_y \\ N_z \end{bmatrix},$$

and its matrix formulation from equation 2.12

$$\vec{R} = \mathbf{L}\vec{N}.$$

Here the R values are the surface radiance values inferred from the images, $\vec{L}$ are the light source vectors, and $\vec{N}$ is the surface normal to be recovered. The magnitude of each $\vec{L}$ vector is the scene irradiance from the associated light source. This irradiance is the sum of all the pixel irradiance contributions from the square light source. To group them together as one light source, $\vec{L}$ is directed at the centre of the square as an approximation of the combined effect of all pixels.

The magnitude of $\vec{L}$ could be calculated as the sum of all individual I_S contributions of each pixel in the square for high accuracy. But since the variation of their values is small due to their proximity, only the contribution of the centre pixel is considered. Here it should be noted that only the relative irradiance values are of interest. So all constants such as $R_{unatten}$ and a_i can be safely disregarded. Due to the assumption that all pixels in the square area contribute equal irradiance and the fact that all light sources contain the same number of pixels, the relative irradiance measure can be written as

$$|\vec{L}_k| = \frac{f(\theta_i, \phi_i)}{r_i^2}, \tag{4.2}$$

where k indexes the light sources ranging from 1 to 6, and i is the index of the centre pixel for the associated light source. For a scene point $\vec{S}$ and a light source centre location $\vec{P}_k$, each light vector $\vec{L}_k$ can be calculated with

$$\vec{L}_k = \frac{f(\theta_i, \phi_i)}{r_i^2} \frac{\vec{P}_k - \vec{S}}{|\vec{P}_k - \vec{S}|}. \tag{4.3}$$

And since r_i is the distance from the scene point to the light source,

$$\vec{L}_k = f(\theta_i, \phi_i) \frac{\vec{P}_k - \vec{S}}{|\vec{P}_k - \vec{S}|^3}. \tag{4.4}$$

Now that $\vec{L}_k$ has been expressed, one can see how the light vector depends on the position of the scene point $\vec{S}$. The goal of shape recovery is to determine the location of all visible scene points, so $\vec{S}$ is unknown.

A simple method of finding the surface normal is by making an *initial estimate* of the position of $\vec{S}$. This approach is chosen due to its simplicity in comparison to the alternatives. The drawback is that the accuracy of the normals depends on the accuracy of the position estimate. All points are initially assumed to lie on a plane perpendicular to the camera axis at a distance z = DEPTH_ESTIMATE from the camera origin. Then, for each point, the light source vectors are calculated using equation 4.4.

With the light source vectors known, the surface normals are determined by applying a least-squares approach to equation 2.12 at every pixel location. A normalized residual r is obtained during this step which is calculated as

$$r = \frac{|\mathbf{L}\vec{N} - \vec{R}|}{|\vec{R}|}. \tag{4.5}$$

where $\vec{N}$ is the calculated normal vector. The residual serves as a measure for how well the normal vector fits the light vectors and radiance values.

Shadows are handled by thresholding the surface radiance values and omitting light sources that cause a radiance below the selected threshold value. A parameter SHADOW_THRESHOLD sets the pixel value below which a radiance measurement is discarded. For each pixel, the number of radiance measurements above the threshold is counted. If the number is above or equal to three, the photometric stereo algorithm is applied. Otherwise no normal can be recovered for the pixel in question. In the implementation, the shadow handling is accomplished by removing rows corresponding to radiance values below the threshold from the light source matrix and the surface radiance vector.

4.2 Depth from Surface Orientation

Two existing methods were implemented and did not provide satisfactory results, so a new method was developed. Horn's method for recovering depth from gradients [24] was implemented but it only converged slowly and showed problems of numerical instability for gradient maps with surface normals nearly perpendicular to the optical axis. Frankot & Chellappa's method [14] was also implemented. The main problem with this implementation was the lacking ability to handle depth discontinuities.

The new method considers the perspective projection of most cameras rather than using orthographic projection as assumed by the other methods implemented [24] [14]. It defines constraints relating the depth of two adjacent pixels based on the direction of their normals. One constraint per adjacent pair of pixels is defined and added to a large equation system. Then, adding a depth constraint for each independent region allows finding a solution using sparse matrix methods.

4.2.1 Basic Method

The method is based on defining a large set of constraints associating the depth at adjacent pixels. First, the constraint for a single adjacent pair of pixels P_1 and P_2 is derived. Figure 4.2 shows an exaggerated cross section of a surface intersected by two rays r_1 and r_2 passing through these pixels. The rays are defined through the normalized values x_{n1} and x_{n2}, calculated from their pixel coordinates. The surface is intersected by the rays at points $S_1 = (z_1, x_1)$ and $S_2 = (z_2, x_2)$. The values of x_{n1} and x_{n2} relate the points (z, x) on the ray such that

$$x_1 = x_{n1} z_1, \text{ and} \tag{4.6}$$

$$x_2 = x_{n2} z_2. \tag{4.7}$$

The unknown quantities to be determined are z_1 and z_2. The values of x_1 and x_2 are also unknown. By assuming that slope $(z_2 - z_1)/(x_2 - x_1)$ of the line connecting S_1 and S_2 is known, a relationship

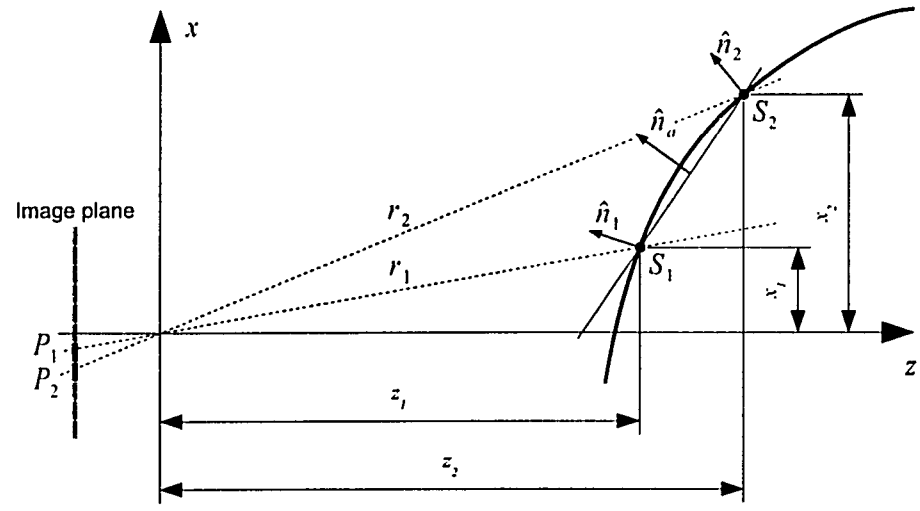

Figure 4.2: Cross section of a surface intersected by two rays r_1 and r_2 passing through adjacent pixels P_1 and P_2. Note that the angle between the rays is exaggerated for better illustration. The true angle between the rays is small for common image resolutions.

between z_1 and z_2 can be established. The slope is

$$\frac{z_2 - z_1}{x_2 - x_1} = \frac{-\hat{n}_{ax}}{\hat{n}_{az}}, \tag{4.8}$$

where $\hat{n}_{az}$ and $\hat{n}_{ax}$ are the components of the normal $\hat{n}_a$ to the line passing through S_1 and S_2. This normal is unknown, but it can be approximated by averaging the surface normals at the intersection points which are known. This assumes that the surface is smooth between the two intersection points. The error in this approximation decreases with decreasing angle between the two rays. The approximation used here is

$$\hat{n}_a \approx \frac{\hat{n}_1 + \hat{n}_2}{|\hat{n}_1 + \hat{n}_2|}. \tag{4.9}$$

Using this approximation for $\hat{n}_a$, equation 4.8 can be transformed to constrain z_1 and z_2 with the linear equation

$$(-n_{az} - n_{ax}x_{n1})z_1 + (n_{az} + n_{ax}x_{n2})z_2 = 0. \tag{4.10}$$

The coefficients of the z values can be written as $c_{i,j} = -n_{az} - n_{ax}x_{nj}$ and $c_{i,k} = n_{az} + n_{ax}x_{nk}$, where i indexes the individual pixel pairs, j indexes the first pixel, and k indexes the second pixel of the pair. This allows each constraint to be written as

$$c_{i,j}z_j + c_{i,k}z_k = 0. \tag{4.11}$$

This derivation examined two pixels adjacent in the x-direction. Naturally, an equivalent relationship holds for the y-direction. For pixels adjacent in the y-direction, the z coefficients are calculated

with $c_{i,j} = -n_{az} - n_{ay}y_{nj}$ and $c_{i,k} = n_{az} + n_{ay}y_{nk}$. The coefficients can be zero under one circumstance. When $\hat{n}_a$ is perpendicular to a ray, then the corresponding coefficient is 0.

A large linear system of equations can be constructed by listing the equations associated with all adjacent pixel pairs present in the image. For each equation, the z coefficients are entered in a large sparse coefficient matrix $\mathbf{H}$, such that the system can be expressed in matrix notation as the homogeneous system

$$\mathbf{H}\vec{z} = \vec{0}, \tag{4.12}$$

where $\vec{z}$ is a column vector containing all depth values. For an $m \times n$ image, there are $m(n-1)$ constraints in the x-direction (horizontal) and $(m-1)n$ constraints in the y-direction (vertical). So $\mathbf{H}$ is of size $(m(n-1)+(m-1)n) \times mn$, and the vector $\vec{z}$ is of size $mn \times 1$. This shows that even for relatively low resolutions such as 200×200, $\mathbf{H}$ is quite large at 79600×40000. It is also sparse, since every row contains at most two non-zero coefficients.

The rank of the coefficient matrix $\mathbf{H}$ for an image with mn pixels can be either $mn-1$ or mn assuming that all the coefficient values are non-zero. This property can be shown as follows. Assume the image pixels are represented by nodes, and the constraints form edges between pairs of nodes. Every edge corresponds to a row in $\mathbf{H}$. A tree that connects all nodes has $mn-1$ edges and corresponds to a set of rows, termed *tree rows* here. All these rows are linearly independent, so the rank of the matrix formed by all tree rows is $mn-1$. Adding an edge between two nodes of the tree forms a cycle. Consider the case where the row of $\mathbf{H}$ corresponding to an added edge is linearly independent of the other rows corresponding to edges that are part of the cycle. Then the rank of $\mathbf{H}$ is increased to mn. So the rank of $\mathbf{H}$ can only be $mn-1$ if every row that is not a tree row, is a linear combination of tree rows.

If the rank is $mn-1$, there are an infinite number of solutions. All solutions are related by a scalar factor λ such that if $\vec{z}$ is a solution, then so is $\lambda\vec{z}$. This is a result of the *depth ambiguity* inherent in perspective projection where an object at a certain distance from the camera can appear the same as a smaller object at a closer location. If $\mathbf{H}$ is full rank (the rank is mn), the only solution is the trivial solution $\vec{z} = \vec{0}$. In this case, an additional constraint is necessary to obtain a non-trivial solution. In the first case, where $\mathbf{H}$ is rank deficient, adding a constraint resolves the depth ambiguity problem. So in both cases, the addition of a constraint is beneficial.

Two options for constraining the solution were considered. The first was to constrain $|\vec{z}| = 1$ and find a solution that minimizes $\mathbf{H}\vec{z}$. This can be accomplished by performing an SVD on $\mathbf{H}$ such that $\mathbf{H} = \mathbf{U}\mathbf{D}\mathbf{V}^T$. The last column of $\mathbf{V}$ then corresponds to the minimizing vector $\vec{z}$. There was however difficulty in implementing this method due to the large size of $\mathbf{H}$. The second option, which was chosen for the implementation, is to specify one of the z values to be non-zero. The non-trivial solution $\vec{z}$ is obtained as the least squares solution to

$$\tilde{\mathbf{H}}\vec{z} = \vec{g}, \tag{4.13}$$

where $\bar{\mathbf{\Pi}}$ is a modified coefficient matrix, and $\vec{g}$ is a $(m(n-1)+(m-1)n+1) \times 1$ vector with all elements equal to zero, except the last element which is set to a desired depth value d. The modified coefficient matrix $\bar{\mathbf{\Pi}}$ is defined as

$$\bar{\mathbf{\Pi}} = \begin{bmatrix} \mathbf{\Pi} \\ \vec{e}_i \end{bmatrix} \tag{4.14}$$

where $\vec{e}_i$ is a row vector with all elements equal to zero except the element at position i which is equal to one. The index i determines the location at which the depth is constrained to the specified value d.

The structure of the coefficient matrix $\bar{\mathbf{\Pi}}$ is best shown by an example using a small image size. For a 3×3 image, there are 9 pixels in the image and a total of 12 constraints between adjacent pixels. After adding the depth constraint, $\bar{\mathbf{\Pi}}$ is a 13×9 matrix. Here, the first 6 rows of $\bar{\mathbf{\Pi}}$ are used for the vertical constraints, and rows 7 to 12 are used for the horizontal constraints. The last row corresponds to the additional depth constraint. Under these conditions the full equation system is written as

$$\begin{bmatrix}
c_{1,1} & c_{1,2} & 0 & \cdots & & & & & \\
0 & c_{2,2} & c_{2,3} & 0 & \cdots & & & & \\
 & \cdots & 0 & c_{3,4} & c_{3,5} & 0 & \cdots & & \\
 & & \cdots & 0 & c_{4,5} & c_{4,6} & 0 & \cdots & \\
 & & & & \cdots & 0 & c_{5,7} & c_{5,8} & 0 \\
 & & & & & \cdots & 0 & c_{6,8} & c_{6,9} \\
c_{7,1} & 0 & 0 & c_{7,4} & 0 & \cdots & & & \\
0 & c_{8,2} & 0 & 0 & c_{8,5} & 0 & \cdots & & \\
0 & 0 & c_{9,3} & 0 & 0 & c_{9,6} & 0 & \cdots & \\
 & \cdots & 0 & c_{10,4} & 0 & 0 & c_{10,7} & 0 & 0 \\
 & & \cdots & 0 & c_{11,5} & 0 & 0 & c_{11,8} & 0 \\
 & & & \cdots & 0 & c_{12,6} & 0 & 0 & c_{12,9} \\
1 & 0 & \cdots & & & & & \cdots & 0
\end{bmatrix}
\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \\ z_7 \\ z_8 \\ z_9 \end{bmatrix}
=
\begin{bmatrix} 0 \\ \vdots \\ \\ \\ \\ \\ \\ \\ \vdots \\ 0 \\ d \end{bmatrix},$$

where the z values are numbered in column-precedent order.

If some of the coefficients $c_{i,j}$ are zero, a unique solution may not be obtainable for the associated pixels. Since the depth of these pixels can not be determined reliably in this case, it is best to simply omit the corresponding z elements from $\vec{z}$, remove the associated columns from $\bar{\mathbf{\Pi}}$, and remove all rows from both $\bar{\mathbf{\Pi}}$ and $\vec{g}$ that form constraints on the pixels involved. The implementation for this thesis only prints a warning message when a coefficient is determined to be zero. In all the experiments, the problem was never encountered so it was not necessary to handle this case. However, for critical applications it is recommended to handle zero coefficients as described to avoid problems in recovering the depth.

This section implicitly assumed that the entire visible depth field is smooth since the depth between all adjacent pixel pairs is assumed to change smoothly. The assumption is only acceptable for a small class of depth fields. If depth field contains regions where the depth changes suddenly between two adjacent pixels, the discontinuity needs to be handled appropriately as the next section will discuss.

4.2.2 Considering Depth Discontinuities

As mentioned in Section 2.6, it is possible that there are discontinuities in the depth map where the normals on either side of the discontinuity do not provide any information about the relative depth over the edge.

The location of these edges in the depth field (the occluding boundaries of objects) can not be determined from the normal map alone without making some assumptions. For example, imagine a book placed flat on a desk viewed from the top. Assume all recovered surface normals of the book and desk are parallel to the viewing direction. At the occluding boundaries of the book, there is a depth discontinuity which can not be detected from the surface normal data alone. In addition, the normals provide no information on the distance between the book surface and the desk. The implementation for this thesis assumes that discontinuities in the depth are associated with rapid changes in the normal map. In the case of the book, the depth recovery would not capture the depth discontinuity unless the surface normals change at the book edges. The depth difference between the book and the desk is not recoverable.

For the assumption used, an edge detection method can be applied to the normal map to find edges of depth discontinuities. A threshold parameter COMPONENT_EDGE_THRES is introduced to adjust the sensitivity of the edge detection method. At the locations where the difference in the normals of two adjacent pixels is larger than the threshold, the associated constraint is removed from the $\tilde{\mathbf{H}}\vec{z} = \vec{g}$ system. By removing a constraint, the adjacent depth values are no longer directly constrained to each other.

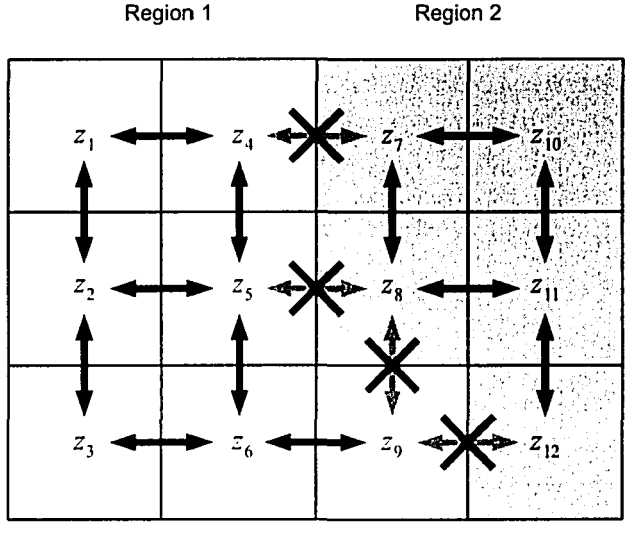

Figure 4.3: An example of a 3×4 image divided into two regions by a depth discontinuity. Four constraints are removed due to large differences in the normals of adjacent pixels.

As shown in Figure 4.3 it is possible that a region in the image is completely isolated from the rest of the image, so its depth values become independent of the other image regions. For an image with k regions, the minimum rank of $\bar{\mathbf{H}}$ then becomes $mn - k$. The depth ambiguity problem applies independently to each region, requiring the addition of a constraint on each of the regions for a non-trivial solution to be obtained. This is performed in a similar fashion to the single depth constraint in the previous section. The coefficient matrix $\bar{\mathbf{H}}$ is defined as

$$\bar{\mathbf{H}} = \begin{bmatrix} \mathbf{H} \\ \vec{e}_{i,1} \\ \vec{e}_{i,2} \\ \vec{e}_{i,3} \\ \vdots \\ \vec{e}_{i,n} \end{bmatrix} \tag{4.15}$$

where $\vec{e}_{i,j}$ is a row vector with all elements equal to zero except the element at position i which is equal to one. The index i identifies the pixel within a region at which the depth is constrained, and $j = 1 \ldots k$ indexes the image regions. The $\vec{g}$ vector is similarly modified to $[\vec{0}, d_1, d_2, d_3, \ldots d_k]^T$, where d_j is the depth value to which the point in region j is constrained. In the implementation, all depth values d_j are set to the DEPTH_ESTIMATE constant. The constrained pixels for each region are the first pixel of the region when scanned in a left-to-right top-down manner. An alternative would be to manually specify which point is constrained to manually specified depth values.

With the constraints described, it is possible to determine a unique solution using a least squares approach. It is important to keep in mind that the recovered depths of each image region are dependent on the depth assumptions made.

4.3 Iterative Estimation

Since the initial depth estimation can deviate far from the actual geometry of the object, the shape recovery results can be improved by repeating the photometric stereo and depth from surface orientation steps. The depth determined in the first step is used as depth estimate for the second step, increasing the accuracy of the recovered normals. In the implementation the constant N_PS_ITERATIONS is used to limit the number of iterations. As the experiments show, this process converges quickly to an accurate estimation of the depth.

4.4 Performance on Synthetic Images

To analyse the performance of the method, the intermediate results of the normal map are examined as well as the final depth map. For the synthetic images, a 3D model is available from which surface normals and depth can be accurately extracted at each pixel. The images are generated from these models as mentioned in Section 3.6.

Table 4.1: Results of the synthetic sphere shape recovery experiment

Iteration	Avg. normal error (degrees)	Avg. abs. depth error (mm)
1	0.4771	0.028
2	0.0065	0.012
3	0.0027	0.011
4	0.0026	0.011

The evaluation is performed by analysing the accuracy of the surface normals and depth using two error measures:

1. The **average normal error** is calculated as the average angle between the recovered and the true normals. It is measured in degrees.

2. The **average absolute depth error**, measured in millimeters, is calculated by averaging the absolute difference between the recovered and true depth at each pixel.

The error measures are calculated over the image region containing the centre pixel. This region is obtained from the segmentation method used for handling depth discontinuities. All the experimental parameters are included in Appendix B.

4.4.1 Sphere

A sphere with a radius of 7mm, centred at the point $(0, 0, 300)^T$ is rendered to an image of size 151 × 151. The depth estimate corresponds to the true depth at the centre pixel (293mm). The estimated depth map is scaled such that the centre depth equals the depth estimate, and therefore the true depth at that location.

The iterative estimation is applied for this experiment. As the results in Table 4.1 show, the accuracy improves with each iteration. After four iterations, further iterations show little improvement of the results. A depth profile comparing the true and estimated depths is provided in Figure 4.4.

4.4.2 Stanford Bunny

The Stanford bunny model is scaled to a size of approximately 180mm from its nose to its tail. The image size for the experiment was 168 × 200. The DEPTH_ESTIMATE value is set to 283mm corresponding to the depth of the centre point.

Similar to the synthetic sphere experiment, the accuracy of the results is improved by repeating the photometric stereo step after performing the initial shape recovery. Table 4.2 summarizes the results. A depth profile is provided in Figure 4.5.

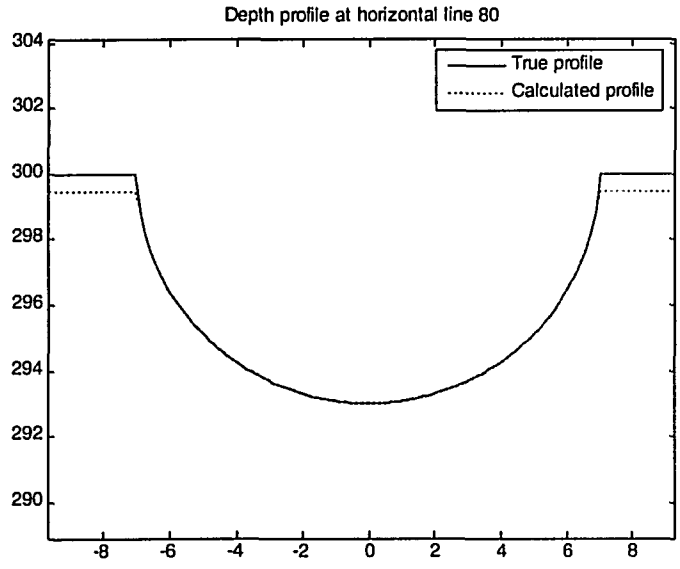

Figure 4.4: Comparison of the true and calculated profiles from the synthetic sphere experiment.

Table 4.2: Results of the synthetic Stanford bunny shape recovery experiment

Iteration	Avg. normal error (degrees)	Avg. abs. depth error (mm)
1	2.39	1.7
2	2.29	4.0
3	0.42	0.9
4	0.42	0.9

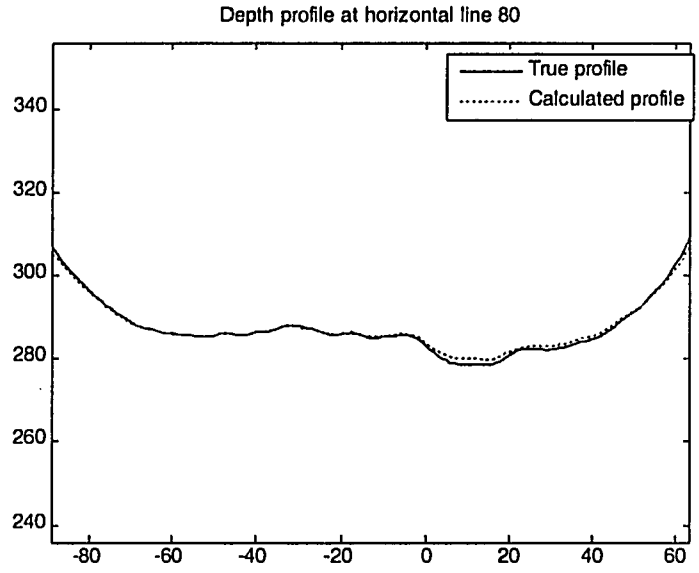

Figure 4.5: Comparison of the true and calculated profiles from the synthetic Stanford bunny model after 7 iterations.

4.5 Performance on Real Images

The same error measures as applied in the performance evaluation on synthetic images are applied here. The parameters used for each experiment are listed in Appendix B.

4.5.1 Sphere

The sphere used is a 22mm diameter ball taken from a computer mouse. Its surface appears nearly Lambertian, but no tests were conducted to confirm this. For this object, the ground truth model was positioned manually such that it matched the position of the sphere in the images captured. The image size is 51 × 64. The results of the photometric stereo step are displayed in Figure 4.6 and Figure 4.7. It can be seen that the normals in the center of the sphere do not point directly to the camera. The albedo is also not perfectly constant over the entire sphere surface. The main factors that are expected to contribute to these errors are the surface BRDF which might be non-Lambertian, the non-ideal light sources, and inaccuracies in the calibration of the screen directionality function and the screen position. The comparison of the results to the ground truth is provided in Figure 4.8. The recovered depth is shown in Figure 4.9, and Figure 4.10 shows a comparison of the recovered and true profiles.

After the first iteration, the average normal error is 17° and the average depth error is 1.8mm. These errors do not reduce in subsequent iterations.

During the analysis of the images, it was noted that the quality of results depend on the shadow

threshold. The shadow threshold should not affect the results in an ideal case where for example the BRDF is Lambertian, and all calibrations are exact. In this case, the shadow threshold affects whether 3, 4, 5, or all 6 light sources are used to determine the surface location at some points. Since the normal would be accurately determined by 3 sources in an ideal case, the addition of more sources should not affect the recovered normal. The two most likely causes for this effect are that the surface is not Lambertian or that the screen directionality calibration is inaccurate.

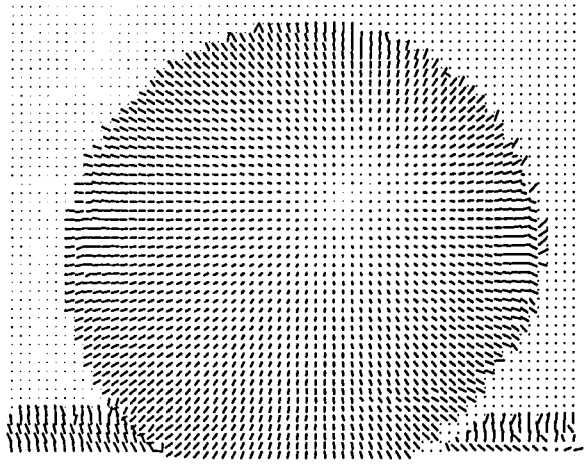

Figure 4.6: The recovered surface normals of the real sphere.

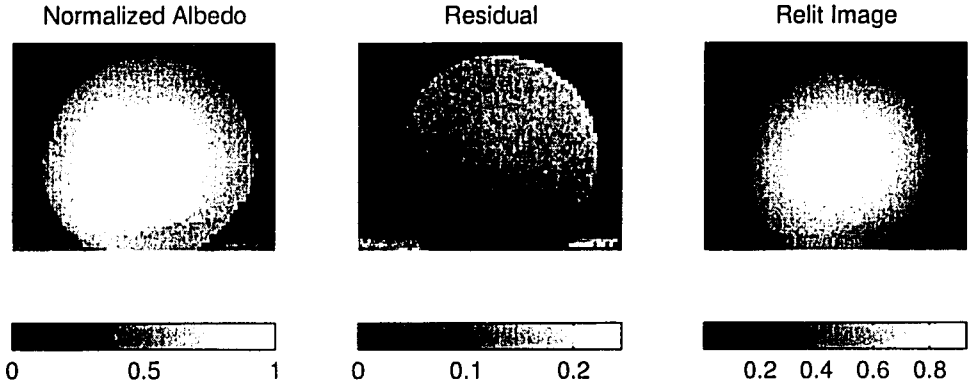

Figure 4.7: The recovered albedo of the real sphere, the residual values, and the relit sphere. The light used for relighting is a single distant point source located on the camera's optical axis.

4.5.2 Stanford Bunny

A plaster model of the Stanford bunny model was printed using a 3D printer. With this real object it is possible to compare the recovered depth to an accurate ground truth. The model was placed at a distance of 280mm from the camera. The ground truth model was positioned by minimizing

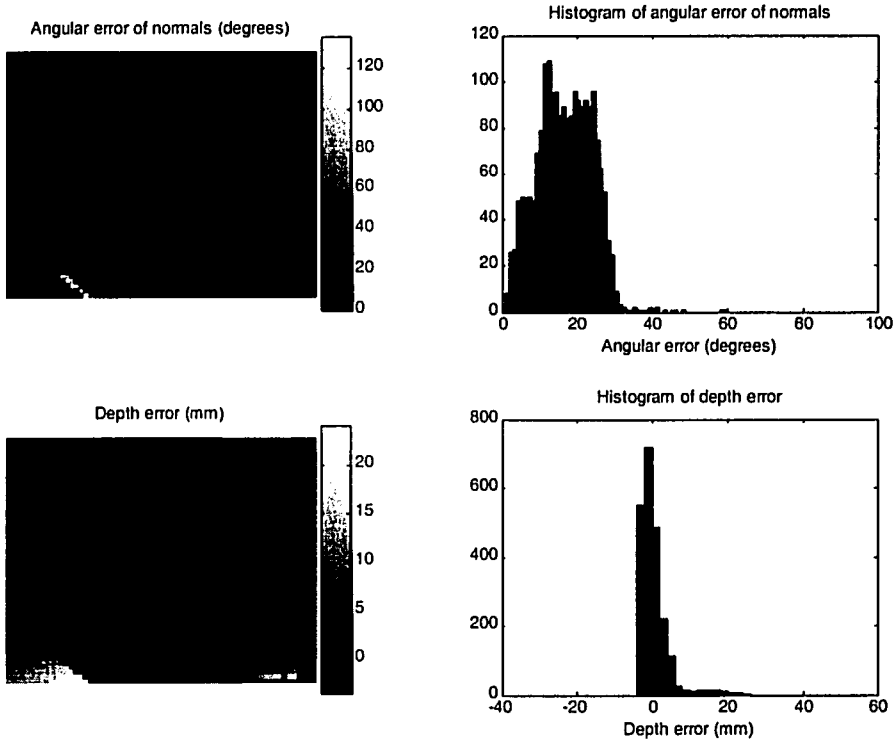

Figure 4.8: The results of the shape recovery of the real sphere.

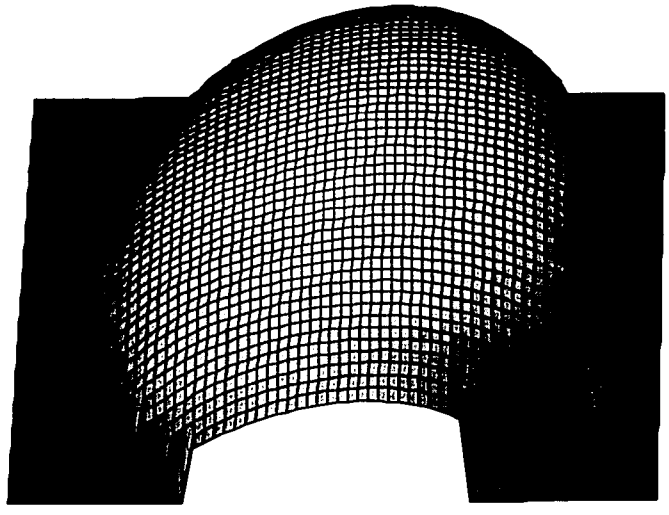

Figure 4.9: The depth map recovered for the real sphere experiment.

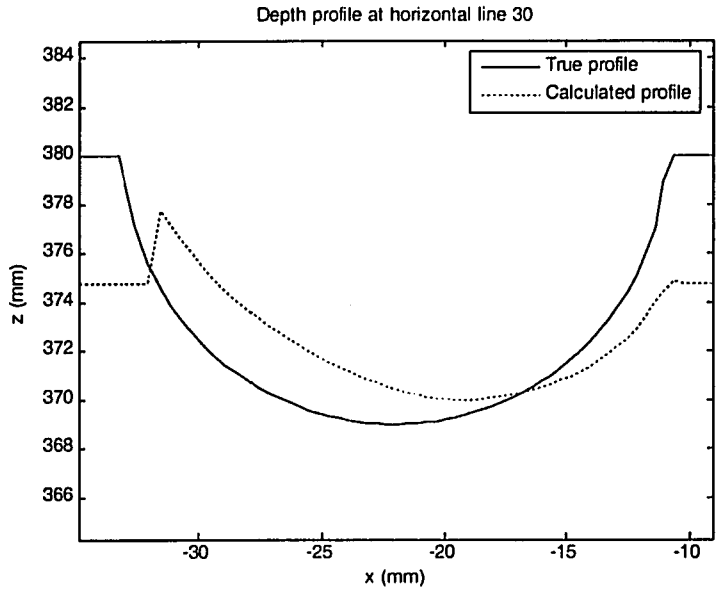

Figure 4.10: Comparison of the true and calculated profiles for the real sphere experiment.

the distance between a set of reprojected reference points on the model and the associated points manually selected in the captured images. An image size of 168 × 200 was used for the analysis. All other experimental parameters are included in Appendix B.

The results of the photometric stereo step are displayed in Figure 4.6 and Figure 4.7. It can be noted that although the albedo of the bunny model is constant, it is not recovered as such. The reasons for the errors in the results are expected to be the same as those for the real sphere experiment. The comparison of the results to the ground truth is shown in Figure 4.13. The angular error histogram shows that considerable errors are already present in the recovery of the surface normals. So the depth errors are likely just a propagation of these errors into the final result. The depth from surface orientation step is not expected to contribute much to the depth error.

The recovered depth is shown in Figure 4.14, and Figure 4.15 shows a comparison of the recovered and true profiles. After the first iteration, the average error in the normals is 21° and the average absolute depth error is 10mm. Further iterations do not improve the results. It can be noted though that the visual appearance of the relit recovered bunny model is quite good. This shows that the results can be used in applications where high accuracy is not required and visual appearance is more important.

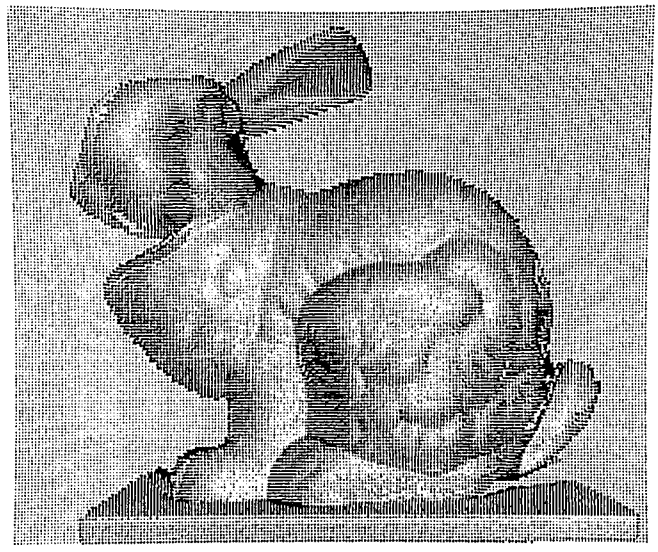

Figure 4.11: The recovered surface normals of the real Stanford bunny.

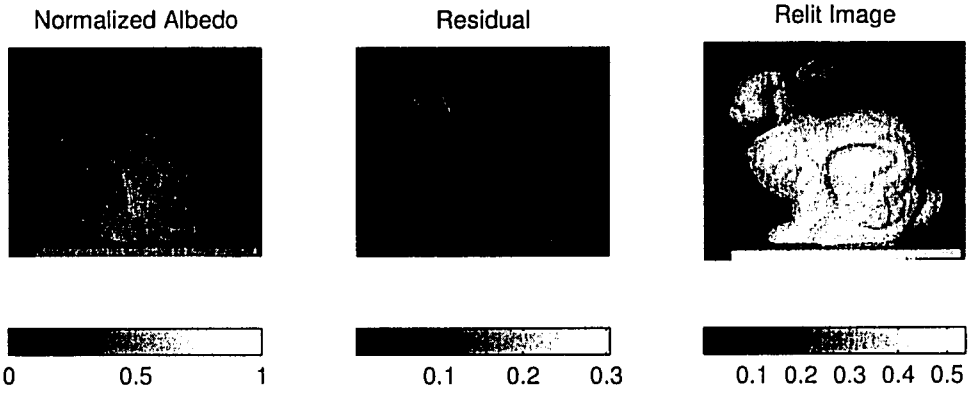

Figure 4.12: The recovered albedo of the real Stanford bunny, the residual values, and the relit bunny. The light used for relighting is a single distant point source located on the camera's optical axis.

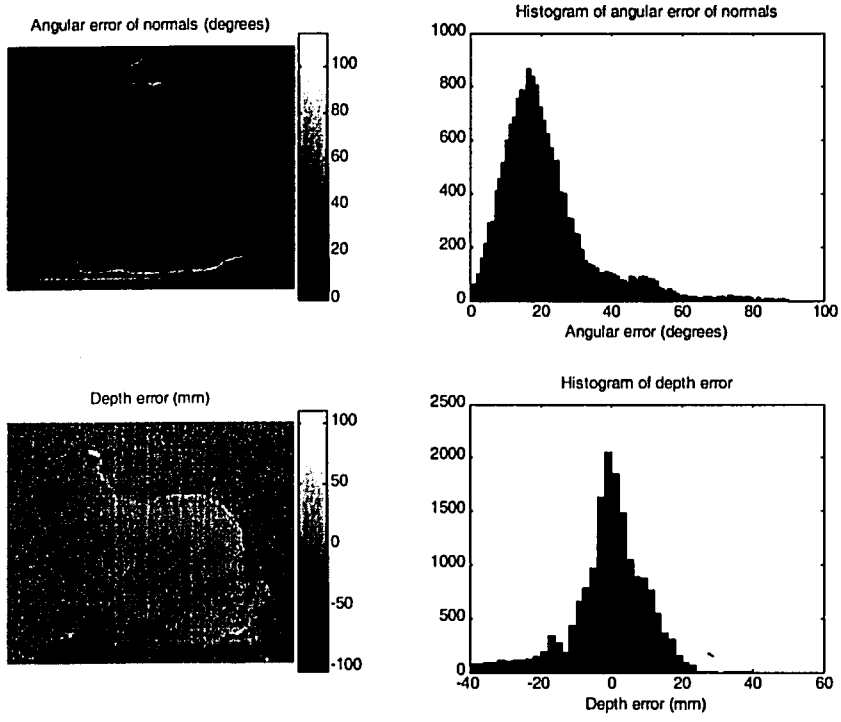

Figure 4.13: The results of the shape recovery of the real Stanford bunny.

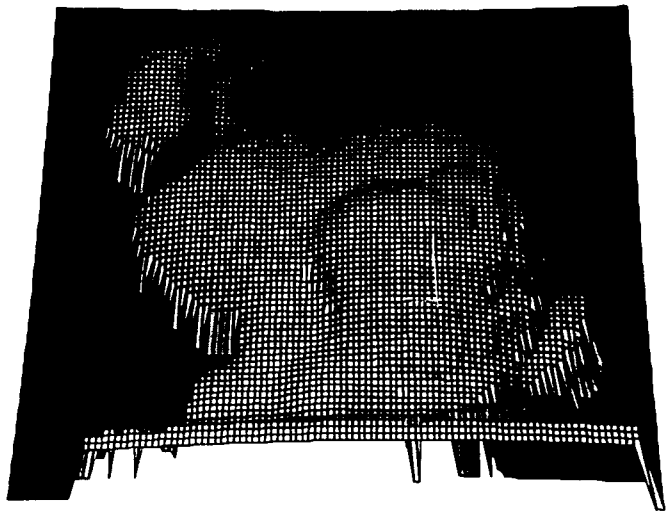

Figure 4.14: The depth map recovered for the real Stanford bunny experiment.

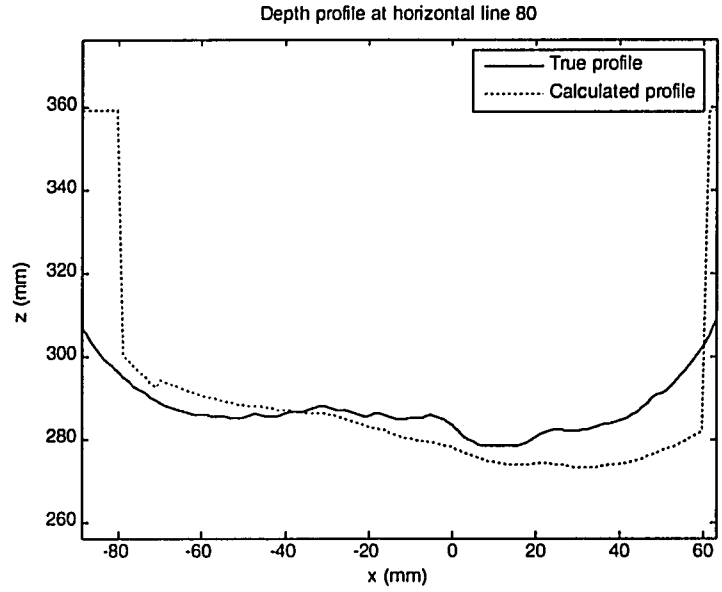

Figure 4.15: Comparison of the true and calculated profiles for the real Stanford bunny experiment.

4.6 Direct Depth Recovery

In addition to shape recovery using traditional photometric stereo, a new method of depth recovery was developed. It is shown here to work accurately on the set of synthetic Stanford bunny images. Unlike the previously examined method, this method determines the depth, surface normals and albedo all in a single step rather than breaking the process down into two steps.

The method is based on adjusting the depth at individual pixels such that the residual of the photometric stereo equation is minimized. This would not be possible with distant point sources and orthographic projection since under those assumptions a pixel's shading does not depend on its depth (assuming the pixel is not shadowed). But because the illumination model used in this thesis employs finite point sources and a screen directionality factor, the depth at a pixel does influence the pixel's shading.

Experiments were performed on synthetic and real images. The direct depth recovery only succeeded on synthetic images. It is likely that for accurate results to be obtained in real experiments, the setup would need to be calibrated with very high accuracy. For this, more accurate equipment might be necessary.

4.6.1 Theory

For the previous method, a depth estimate is used to be able to calculate light vectors for the photometric stereo equation (equation 2.11). In theory, given a setup where all assumptions are met, the

residual of this equation would be zero when the depth estimate is correct. If the depth estimate is incorrect, the light vectors become inaccurate and it is possible that there is no solution that meets all the constraints. In that case the residual is non-zero.

So the residual can be used as an indicator of how well the estimated depth fits the image data. The depth estimate can be adjusted at each point individually such that the residual value is minimized. Recall equation 2.11

$$\begin{bmatrix} R_1 \\ \vdots \\ R_n \end{bmatrix} = \begin{bmatrix} L_{1x} & L_{1y} & L_{1z} \\ \vdots & \vdots & \vdots \\ L_{nx} & L_{ny} & L_{nz} \end{bmatrix} \begin{bmatrix} N_x \\ N_y \\ N_z \end{bmatrix},$$

and its matrix formulation from equation 2.12

$$\vec{R} = \mathbf{L}\vec{N},$$

where $\vec{R}$ contains the surface radiance values, $\vec{L}$ are the light vectors which are grouped into the matrix $\mathbf{L}$, and $\vec{N}$ is the surface normal multiplied by the local surface albedo. The light source vectors $\vec{L}_k$ are determined by equation 4.4

$$\vec{L}_k = f(\theta_i, \phi_i)\frac{\vec{P}_k - \vec{S}}{|\vec{P}_k - \vec{S}|^3},$$

where $f(\theta_i, \phi_i)$ is the directionality function, $\vec{P}_k$ is the light source center, and $\vec{S}$ is the surface point. The depth estimate constrains the position of $\vec{S}$, so it can be clearly seen how the light vectors depend on this depth estimate. The residual r is determined as

$$r = |\mathbf{L}\vec{N} - \vec{R}|, \tag{4.16}$$

where $\vec{N}$ is the normal vector determined through least solution equation 2.12. Since the residual r depends on the location of the surface point $\vec{S}$, it can be written as $r(\vec{S})$. Under ideal conditions the residual is zero when $\vec{S}$ is accurate, so the location of the point can be determined by minimizing $r(\vec{S})$. Then for a single pixel the recovered surface point $\vec{S}_r$ can be written as

$$\vec{S}_r = \underset{\vec{S}}{\operatorname{argmin}}\, r(\vec{S}). \tag{4.17}$$

By performing this minimization for every pixel location, a depth map of the scene is recovered together with the surface normals and albedo as a byproduct.

In regions where traditional photometric stereo can not recover surface normals, this method can not recover depth. These regions include all areas that are lit by only 2 or fewer light sources, and areas where the surface albedo is zero.

4.6.2 Experiment

This experiment uses the same synthetic image data of the Stanford bunny as used in Section 4.4.2. A simplistic minimization approach was employed. The residual at each pixel was determined for

200 different depth values in increments of 2mm. Then, for each pixel the depth value associated with the smallest residual was recorded in the depth map.

As shown in Figure 4.16, the depth accuracy is good. The average absolute depth error is 2.5mm. This error could be further reduced by using a smaller increment for the depth.

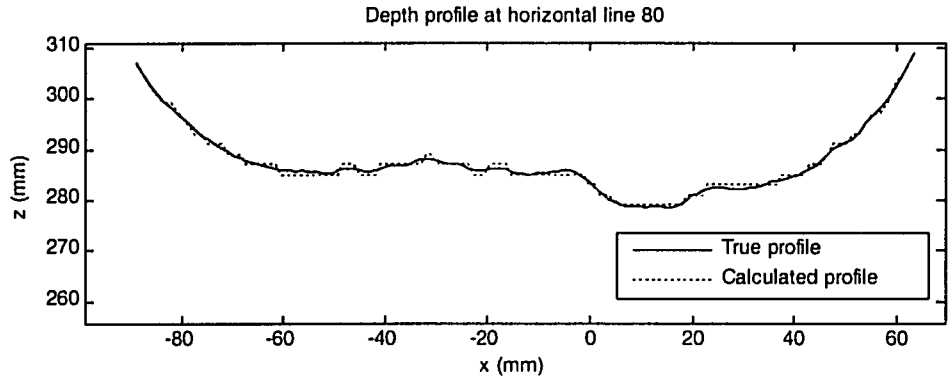

Figure 4.16: The depth profile for the direct depth recovery experiment on the synthetic Stanford bunny images.

The main benefits of this approach over the traditional photometric stereo method is that no depth estimate is required. Depth discontinuities also do not pose as much of an issue since no smoothness assumption needs to be made. The application of this approach to real images still requires future work. The recovered depth values are highly inaccurate with errors over 100mm. It is suspected that certain factors such as non-Lambertian BRDFs and inaccuracies in the calibration have a significant impact on location of the minimum of the residual. So, to obtain good results using direct depth recovery from real images, a more accurately calibrated setup would likely be required.

4.7 Summary

This chapter has demonstrated how the controlled illumination apparatus can be used for shape recovery. Highly accurate results are achieved on synthetic images, showing the potential of the setup under ideal conditions. Good results were also obtained for real images with the first depth recovery method.

A novel method for calculating depth from surface orientation has been presented. To the best knowledge of the author, it is the first method to use perspective projection for depth recovery. The high accuracy of the method is demonstrated by the results of the synthetic experiments.

The first depth recovery method which is based on photometric stereo, achieves good results on real images. The results do show noticeable errors in both the direction of the surface normals and the recovered depth though. Since the depth recovery method is the same for both synthetic and real experiments, and the synthetic results show little depth error, it can be concluded that the depth

error is mainly caused by the error in the normals. The differences between the real and synthetic images listed in Section 3.6 show the potential sources of error. Since the error in the normals is not random and pixel independent, image noise can be ruled out as the main source of error. More likely contributing factors are

- the potentially non-Lambertian surface reflection,
- the effects caused by non-ideal light sources,
- the inaccuracies in the screen directionality function, and
- the error associated with screen position calibration.

It is non-trivial to determine which of these factors are the primary sources of error. But by addressing each factor individually, the error in the normals is expected to be reduced. For example although the error associated with the screen position calibration would have a biasing effect on the normals as observed in the results, it is not necessarily the main contributing source of error.

In addition to the shape recovery method based on photometric stereo, a new direct depth recovery method was presented. Rather than dividing the process into a normal recovery step followed by a depth recovery step, the depth is recovered directly by minimizing the residual of the photometric stereo equation. The method is too sensitive to the errors associated with real images for depth to be recovered within a reasonable error. However for synthetic images the accuracy is high, as demonstrated in the experiment.

The general results are promising as initial results from a new apparatus. Improvements to the apparatus and the applied methods would likely better the results considerably. To obtain a comparison to previous methods, Woodham's work [46, 47] would be best suited since this thesis is based on his early photometric stereo papers. However these papers do not provide an experimental analysis of the approach, so no direct comparison can be made. Only a few later methods [21, 39, 41] include a quantitative comparison of experimental results on real scenes to a ground truth. Other methods [1, 2, 23, 34, 37] do not provide this type of comparison.

The methods that quantitatively evaluate results from real scenes provide an average angular error between the recovered and true surface normals. The ones discussed here achieve errors below 5°, while the average angular errors of the results presented in this chapter are 17° for the real sphere, and 21° for the Stanford bunny. Tagare and deFigueiredo's method for non-Lambertian surfaces [41] is examined on Lambertian and non-Lambertian spheres. They also study how photometric stereo performs when attempting to recover the normals of the non-Lambertian sphere using a Lambertian surface assumption. For this case they report a rms error of 19° and conclude that it is clearly inappropriate to use a Lambertian surface assumption on a non-Lambertian surface. When applying their non-Lambertian method the same images, the angular rms error is reduced to 2°. Hayakawa's method [21] recovers normals under a light source with arbitrary motion. He evaluates the normals

by measuring the angle between two planar surfaces on a milk carton. The recovered and true angles differ by less than 3°. Finally Solomon and Ikeuchi's results [39] show accuracies of 3° on a specular sphere. They use a bright bulb at over 2.6 meters distance from the scene, which generates more ideal lighting conditions than those achievable with a LCD display.

The lower angular errors can in part be explained by the more ideal lighting conditions and the ability of some methods to consider non-Lambertian BRDFs. To effectively reduce the error in the results, it would be greatly beneficial to use a method that does not assume Lambertian BRDFs. In addition, applying a method that employs extended light sources rather than distant point sources would provide great benefits.

Chapter 5

Evaluation of a Lighting Estimation Method

With a set of images generated under known lighting conditions, one can evaluate the performance of a lighting estimation method by comparing the recovered and true lighting conditions. Using a raster display for this purpose has an advantage over other real setups. The accuracy of the evaluation depends on the accuracy with which one can generate lighting conditions. For example, the accuracy with which a point light source can be positioned limits the accuracy with which a point source estimator can be evaluated. Assuming that the position of the screen is accurately calibrated with respect to the camera, all pixels are accurately aligned on a grid allowing fine control of light source positions. The screen also allows the display of complex distributions which becomes useful for the evaluation of distribution recovery methods. One limitation of a raster display in comparison to a light dome is that one can not control the entire hemisphere around the scene. This chapter shows that despite this limitation, a very useful evaluation can be performed.

As pointed out in Section 2.7, lighting estimation methods differ in the types of lighting conditions they recover. There are methods that recover single distant point sources, some that estimate multiple distant point sources, and others that recover entire illumination distributions. Due to the variety in recovered illumination types, it becomes necessary to develop multiple evaluation measures.

An implementation of the lighting estimation method proposed by Singh & Ahuja [38] is evaluated. Since the technique recovers an illumination distribution, it can also be used for the recovery of single distant point sources. The evaluation is performed using a set of six different lighting conditions, each with a single point source. Three evaluation measures are used in each of the experiments. Two of them use only the direction of the estimated point source, while the other evaluates the recovered distribution. The different experiments examine how modifications to the method and changes to the parameters affect the results. The evaluation effectively demonstrates how the setup can be employed to obtain performance measures of a lighting estimation technique.

5.1 Implementation

5.1.1 General Information

The implementation performs Singh & Ahuja's method on images of a sphere captured with the controlled lighting setup. Although the method can be applied to arbitrary known objects, the mapping of radiance values to the Gaussian sphere is easier to perform with this setup.

The method requires a square map $I(i,j)$ of the radiance as input. It produces an equally sized square map $\tilde{L}(i,j)$ of the estimated light distribution. To determine the input radiance map, the image captured by the camera needs to be remapped. This is accomplished by manually specifying the centre and diameter of the sphere in the image and assuming orthographic projection. To reduce the noise level in the resulting radiance map, the original image is first downsampled before remapping. The resolution of the input map determines the output resolution, so maximising this parameter is desirable. However the computational time for the approach is considerable, and increases dramatically with increasing resolutions. While considering these factors, the experiments were performed with resolutions of 32×32 and 64×64. An example of the original input image and remapped intensities is shown in Figure 5.1.

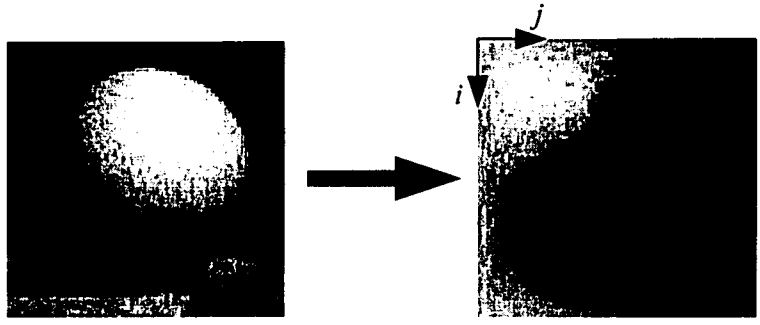

Figure 5.1: An example of the remapping of an image to the discretised Gaussian sphere map. The right side of the remapped data is black since there is not data available for the back-facing side of the sphere.

In order to have complete control over the light field, two images are taken and subtracted from each other. For the first image the screen is set to black. For the second, the desired lighting condition is displayed. The first image is subtracted from the second, resulting in an image that shows only the contributions of the screen illumination without any ambient lighting. In the experiments, the images displayed on the screen are the same square white boxes used for the shape recovery experiments. The squares are 50×50 pixels in size.

An initial guess for the output distribution is required. The experiments use $\tilde{L}(i,j) = 0$ for all i,j in all experiments. The intensity estimation is not evaluated, so the maximum amplitude

parameter A is set to 1 and the input image is scaled in intensity such that the maximum amplitude limit does not affect the results. The number of iterations was selected by observing the changes in the recovered distribution at regular intervals during the iteration process, and ensuring that the change due to successive iterations would be negligible. For resolutions of 32×32 the iteration limit was set to 1000 and for resolutions of 64×64, 3000 iterations were performed.

5.1.2 Handling Unknown Radiance Information

The original method assumes that the reflected radiance is known over the entire Gaussian sphere of surface orientations. However in the experiments not all that information is available. Assuming that the the images are taken using orthographic projection, radiance data is only available for half of the sphere. This is visible in Figure 5.1, where half of the remapped image is black.

A modification was necessary to handle this limitation on the input data. Since the original method uses the radiance value at each surface orientation as an individual constraint on the output, it is simple to omit the half of these constraints corresponding to the unknown input data. It is however important to note that with this modification, there are a reduced number of constraints on the output. This modification was used in all experiments.

5.1.3 Lighting Direction Constraint

Constraints can be added to the approach easily by applying additional projection operators in combination with the ones employed in the original method. For example, if the lighting distribution is known to be confined to a range of directions, then an additional constraint can be applied to ensure that lighting from all other directions is zero. Such a constraint is experimentally examined by confining the recovered distribution to the front facing side of the sphere. This should not be confused with the modification to handle unknown radiance information described in the previous section.

One experiment is performed without and another with this additional constraint. The expected effect of the constraint is that the recovered distribution is more accurate due to the reduction of possible solutions.

5.2 Evaluation Measures

There are many possible ways to evaluate the performance of a lighting estimation method. This is partially due to the variety of representations used for lighting conditions. Here, three evaluation measures are described. They are applicable for the evaluation of single distant point source recovery and distribution recovery. Since the Singh & Ahuja method recovers a distribution rather than distant point sources, its output is processed in order for it to be evaluated as a single point source estimator. The direction of the peak value in the recovered distribution is used as the direction to the point source.

To obtain geometric measures of the precision of recovery, it is necessary to know the location of the sphere. This is estimated from the size and projected location of the sphere in the image. Since the pose of the screen with respect to the camera is known, the pose of the screen with respect to the sphere can be determined. The sphere's reference frame is assumed to have its origin at the sphere's centre, with the z-axis pointing towards the camera. The y-axis is oriented such that it is on the camera's $y - z$ plane. Since the remapping of the radiance data to the Gaussian sphere assumes an orthogonal projection, there is a small error in the orientation of the recovered distribution with respect to this reference frame. The error is minimized by placing the sphere near the image centre, and using a long focal length.

The experiments use the following evaluation measures to determine the accuracy of the lighting estimation method:

- **Projection of recovered distribution to the screen:**
  A visual evaluation of the performance can be achieved by projecting the recovered lighting distribution onto the screen surface and comparing it to the true image displayed on the screen.
- **Angle between recovered and true light source direction:**
  The direction to the peak intensity of the recovered distribution and the direction to the centre of the square light source are compared by measuring the angle between them.
- **Pixel position of light source:**
  The true pixel position of the light source is compared to the intersection of the estimated source direction with the screen.

5.3 Experiments

All analysis was performed on the same image set of a sphere as used in shape recovery from real images. Figure 5.2 shows the sphere under the six different lighting conditions. Two factors that will influence the quality of the lighting estimation are visible in the images. One is the partial occlusion of the bottom part of the sphere by the base. The second is image noise present in all images.

The implementation is first evaluated at a distribution resolution of 32×32 without a direction constraint such that the unknown distribution spans the entire sphere of directions. A second evaluation is performed at the same resolution with the direction constraint confining the recovered distribution to the front facing side of the sphere. The final experiment performs the constrained estimation at a resolution of 64×64.

5.3.1 Without Direction Constraint (32×32 resolution)

Table 5.1 summarizes the results of this experiment. It is clear that the estimation of light source #4 fails since the angular error is 137°. The angular accuracy for the other light source directions ranges from 1° to 28°.

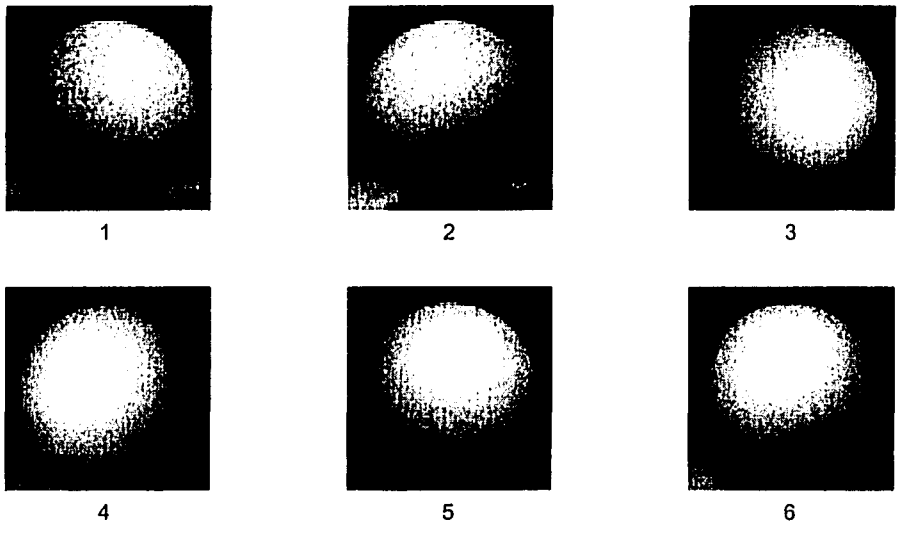

Figure 5.2: The six downsampled images used for evaluation of the Singh & Ahuja lighting estimation method. All images were scaled in brightness such that their peak intensities match. Since the effective light source radiance is smaller for images 1 and 2, their relative noise level is higher.

Table 5.1: Comparison of the recovered and true light source after 1000 iterations **without** the direction constraint at a distribution resolution of 32×32.

	Source#1	Source#2	Source#3	Source#4	Source#5	Source#6
Angular error	1°	3°	28°	137°	8°	17°
Recovered X	84	1194	400	Failed	249	249
Recovered Y	105	102	1725	Failed	506	506
True X	50	1100	50	Failed	500	800
True Y	50	50	900	Failed	500	500

A projection of the recovered distribution is shown in the second column of Figure 5.3. Due to the rough discretisation of the distribution, large quadrilateral regions appear in the projection. The recovered point source location corresponds to the *centre* of the brightest region.

5.3.2 With Direction Constraint (32×32 resolution)

The results in Table 5.2 as well as the projected distributions in the third column of Figure 5.3 show that the direction constraint improves the overall performance of the method. Due to the discretisation, the estimated directions of light sources 1, 2, 5, and 6 remain exactly the same. But the estimates of light sources 3 and 4 are improved such that their angular errors are 12° and 13° respectively.

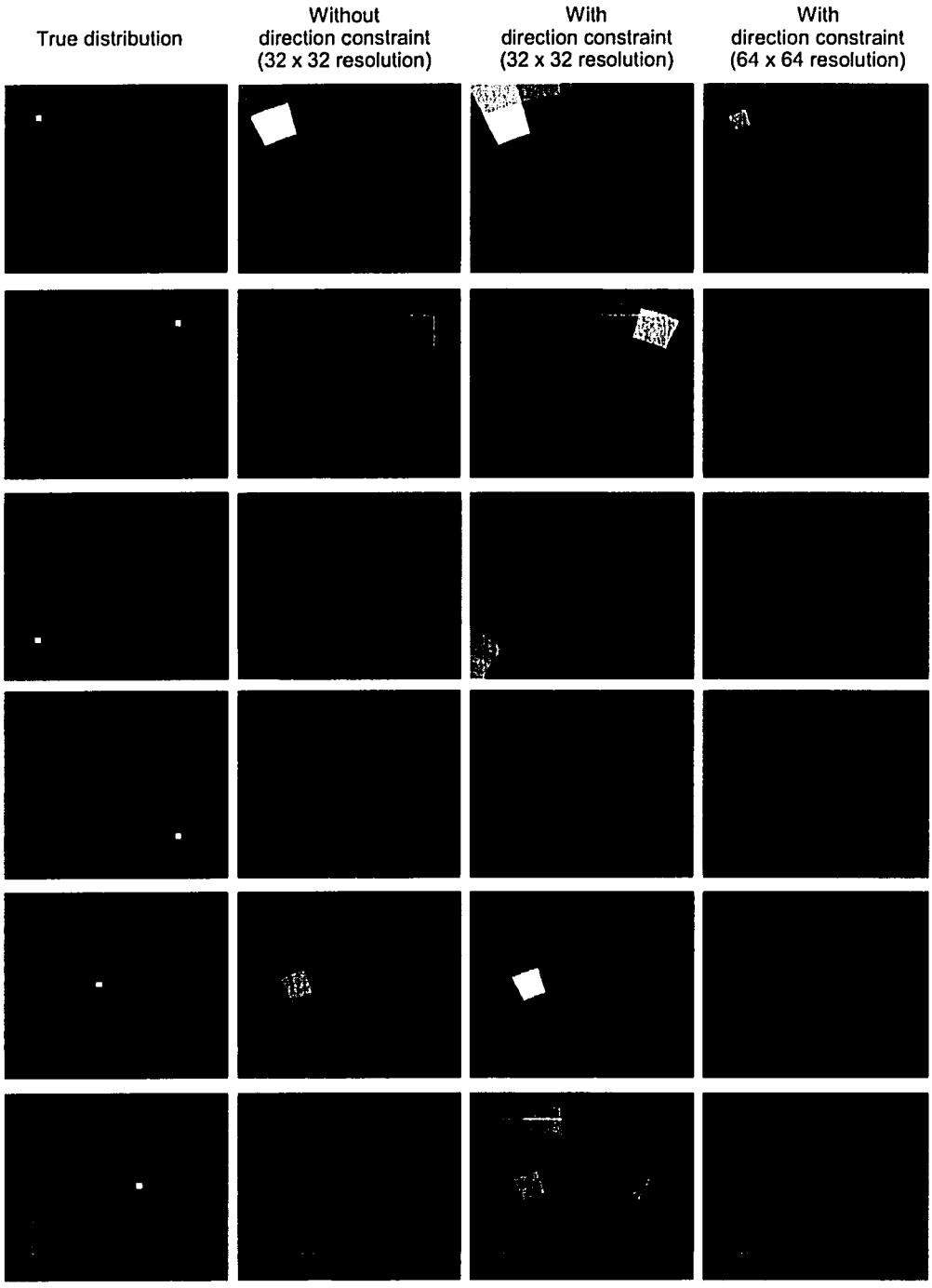

Figure 5.3: Comparison of the true lighting distributions to the recovered distributions projected on the screen. The screen boundary is highlighted as a rectangle in each of the images. Due to the discretisation of the recovered distribution, its projection on the screen appears as a grid of large quadrilaterals. The intensity of each quadrilateral represents the intensity of the recovered source located at its center.

Table 5.2: Comparison of the recovered and true light source after 1000 iterations **with** the direction constraint at a distribution resolution of 32×32.

	Source#1	Source#2	Source#3	Source#4	Source#5	Source#6
Angular error	1°	3°	12°	13°	8°	17°
Recovered X	84	1194	-364	1458	249	249
Recovered Y	105	101	1050	1121	506	506
True X	50	1100	50	1100	500	800
True Y	50	50	900	900	500	500

Table 5.3: Comparison of the recovered and true light source after 3000 iterations **with** the direction constraint at a distribution resolution of 64×64.

	Source#1	Source#2	Source#3	Source#4	Source#5	Source#6
Angular error	1°	3°	40°	43°	32°	19°
Recovered X	72	1205	1297	240	209	249
Recovered Y	78	74	1021	-556	1473	149
True X	50	1100	50	1100	500	800
True Y	50	50	900	900	500	500

5.3.3 With Direction Constraint (64×64 resolution)

For the cases examined, increasing the resolution to 64×64 does not improve any of the results significantly. In fact, the angular error is the same if not larger than with the previous experiment as can be seen in Table 5.3. The projected distribution results in the forth column of Figure 5.3 also show that increasing the resolution does not necessarily increase the accuracy of the estimates.

5.4 Summary

In this chapter a method was developed for evaluating the accuracy of lighting estimation methods. Three evaluation measures were developed that serve to examine the performance of such methods. Singh & Ahuja's technique was implemented, and three experiments were conducted to analyse its performance. The experimental results show that the method can be used even when only half of the radiance data from the Gaussian sphere of surface orientations is known. In addition, by constraining the distribution to the front facing side of the sphere, the results were improved such that the direction of single distant point sources were recovered within 1° to 17° angular error. It was found that increasing the resolution from 32×32 to 64×64 did not improve the point source recovery results and only slightly improved the distribution recovery.

This chapter demonstrates how the controlled lighting setup could be used for the experimental analysis of many methods. Through the use of the evaluation measures presented, lighting estimation techniques can be effectively compared to each other.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

6.1.1 Controlled Illumination

A framework for using raster display devices as controlled light sources has been proposed. An apparatus using an LCD screen was set up and calibrated. Calibration methods were proposed for the screen orientation and screen directionality function. They are not only applicable to the implemented setup, but also for other raster display devices. It was noted that the directionality of LCD screens is considerable and needs to be calibrated appropriately. A significant effort was put into ensuring a high accuracy in all calibration procedures. This is necessary since any errors in the controlled illumination setup directly affect the accuracy of the applications.

The main challenge discovered was the limited light emission from the LCD screen. Together with the unavoidable sensor noise and limited sensitivity of the camera, the low light levels from the screen made it necessary to enclose the apparatus with a black cloth to reduce the ambient light. Large aperture settings and long exposure times of more than 10 seconds were necessary to achieve good results. So the brightness of the screen and noise level of the camera form the greatest limitations on the accuracy of the results. But through the improvements suggested in Section 6.2.1 these challenges are addressed.

The applications to shape recovery and the evaluation of lighting estimation methods show that the developed framework is a useful tool with potential future use in multiple areas. Much of the potential is still untapped, so further investigation is greatly encouraged.

6.1.2 Shape Recovery

To demonstrate the applicability of the controlled illumination apparatus for shape recovery, two shape recovery techniques were implemented. For the first method, a novel technique for determining the surface depth from the surface normals under projective projection was developed.

The performance of the first shape recovery method was tested through experiments on synthetic and real scenes. In both cases the results were compared to ground truth data. For synthesized

images of a 180mm size model of the Stanford bunny, an average depth accuracy of 0.9mm was achieved. For real images of the model at the same scale, the average depth accuracy was 10mm. The effects of screen directionality and the inverse square law necessitated use of an initial depth estimate. For both of the results listed here, an accurate depth estimate value is assumed. Analysis of the recovered normals showed that the depth errors most likely originate from the photometric stereo step and not from the subsequent depth from surface orientation.

The direct depth recovery method was shown to work accurately on synthetic images. Though it has the advantage of not depending on an initial depth estimate, in its current state it is not robust enough to work on real images. The expected reason for this is that the location of the residual minimum is greatly affected by the inaccuracies associated with a real setup.

The methods are limited to application on Lambertian surfaces, so errors in the results can be partially attributed to not perfectly Lambertian surface BRDFs. Another cause for errors in the reconstruction is the inaccuracy in the directionality function calibration. Since the screen radiance depends greatly on the viewing direction, an inaccurate calibration of the directionality function is expected to have considerable effects on the shape recovery results. In addition, errors in the geometric calibration and the necessity of using non-point sources increase the difficulty in achieving highly accurate results. But by using methods that consider the specific features of the apparatus, great improvements are expected to be seen.

As initial results from a novel apparatus, the results are very promising. While the current methods do have the noted limitations, there are numerous possible modifications and extensions that would greatly increase the range of applications as Section 6.2.2 describes.

6.1.3 Evaluation of Lighting Estimation Methods

An experimental performance analysis of Singh & Ahuja's method was conducted, demonstrating the use of the controlled lighting setup for the experimental evaluation of lighting estimations methods. The setup makes it possible to evaluate these techniques using real scenes. Although evaluation using synthetic images allows more precise control of the lighting, the developed apparatus has the benefit of easily capturing complex scenes under controlled illumination.

With some modifications to Singh & Ahuja's original method, the direction of point sources was estimated within 1° to 17° angular error. Similar experiments can be performed on other methods, allowing comparison of different techniques to each other.

6.2 Future Work

6.2.1 Controlled Illumination

To further analyse the use of raster displays for controlled illumination it would be beneficial to perform experiments using raster displays other than the LCD monitor used in this thesis. Use of

an LCD projector for example would provide benefits of greater brightness and less dependence on the directionality. Especially if a Lambertian projection surface was employed, the directionality function could be completely eliminated from the model. In general, different display devices are expected to have different characteristics which may prove beneficial for specific applications.

The use of more sensitive cameras with less sensor noise could greatly improve the accuracy of any application using a controlled lighting setup. The results of the shape recovery method applied to synthetic images shows that there is great potential for accurate shape recovery when given high quality input images.

The screen directionality calibration could be improved to not rely on use of a turntable. It would be possible to develop a method that determines the screen directionality function based on a set of images of the screen take from arbitrary angles.

Finally, the necessity of enclosing the proposed apparatus to reduce ambient light is a considerable limitation. Brighter displays and more sensitive cameras would make it possible to perform experiments without the enclosure.

6.2.2 Shape Recovery

The ability to recover the shape of non-Lambertian surfaces is the most desirable enhancement to the current implementations. This could be accomplished by implementing the more recent photometric stereo methods outlined in Section 2.5.2. The use of raster displays for controlled illumination has the advantage over other controlled illumination methods in that a dense distribution of sources can be accurately controlled. For non-Lambertian BRDFs this dense distribution is expected to assist in accurately recovering both the surface orientation and BRDF simultaneously.

Some of the newer photometric stereo methods can recover scenes lit under general lighting conditions including extended light sources [2] [16]. Using large areas of the screen rather than small patches as light sources would provide the benefit of greater screen radiance. This would lead to shorter capture times and it could alleviate the necessity for an enclosure.

Using multiple screens around scene would increase the range of light directions and is expected to improve the accuracy of the shape recovery. Integrating the proposed shape recovery method with other methods such as stereo or shape from silhouette also has promising potential and deserves further examination.

The direct depth recovery method could be improved by trying to determine the main causes of poor performance on real images. Taking measures to reduce the effect of these factors, would allow recovery on real images which might come near to the high precision achieved on synthetic images.

6.2.3 Evaluation of Lighting Estimation Methods

More evaluation measures could be developed to assess the performance of multiple point source estimation and the estimation of extended sources. These measures could also include the evaluation

of the intensity estimation which most methods perform. Finally, a performance comparison of multiple methods would provide useful information for future research in field of lighting estimation.

Bibliography

[1] S. Barsky and M. Petrou. The 4-source photometric stereo technique for three-dimensional surfaces in the presence of highlights and shadows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(10):1239–1252, 2003.

[2] R. Basri and D.W. Jacobs. Photometric stereo with general, unknown lighting. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages II:374–381, 2001.

[3] P. Belhumeur, D. Kriegman, and A. Yuille. The bas-relief ambiguity. *Int. J. Comput. Vision*, 35(1):33–44, 1999.

[4] P.J. Besl and R.C. Jain. Three-dimensional object recognition. *ACM Computing Surveys*, 17(1):75–145, 1985.

[5] J.-Y. Bouguet. MATLAB camera calibration toolbox. http://www.vision.caltech.edu/bouguetj/calib_doc/. Viewed on May 26, 2005.

[6] J.-Y. Bouguet and P. Perona. 3d photography using shadows in dual-space geometry. *Int. J. Comput. Vision*, 35(2):129–149, 1999.

[7] L. M. Bregman. The method of successive projection for finding a common point on convex sets. *Doklady Akademia Nauk. SSSR*, 162(3):688–692, 1965.

[8] M.J. Brooks and B.K.P. Horn. Shape and source from shading. In *MIT AI Memo 820*, 1985.

[9] W. Chojnacki, M. J. Brooks, and D. Gibbins. Revisiting pentland's estimator of light source direction. *Journal of the Optical Society of America A*, 11(1):118–124, 1994.

[10] P. Debevec. HDR Shop. http://www.hdrshop.com/. Viewed June 8, 2004.

[11] P. Debevec, T. Hawkins, C. Tchou, H.-P. Duiker, W. Sarokin, and M. Sagar. Acquiring the reflectance field of a human face. In *Proc. SIGGRAPH*, pages 145–156. ACM Press/Addison-Wesley Publishing Co., 2000.

[12] P. Debevec and J. Malik. Recovering high dynamic range radiance maps from photographs. In *Proc. SIGGRAPH*, pages 369–378, 1997.

[13] P. Debevec, A. Wenger, C. Tchou, A. Gardner, J. Waese, and T. Hawkins. A lighting reproduction approach to live-action compositing. In *Proc. SIGGRAPH*, pages 547–556. ACM Press, 2002.

[14] 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.

[15] R. Furukawa, H. Kawasaki, K. Ikeuchi, and M. Sakauchi. Appearance based object modeling using texture database: acquisition, compression and rendering. In *Proc. Eurographics workshop on Rendering*, pages 257–266, Aire-la-Ville, Switzerland, Switzerland, 2002. Eurographics Association.

[16] A.S. Georghiades. Incorporating the torrance and sparrow model of reflectance in uncalibrated photometric stereo. In *Proc. IEEE Int. Conf. on Computer Vision*, pages 816–825, 2003.

[17] M.D. Grossberg and S.K. Nayar. Determining the camera response from images: What is knowable? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(11):1455–1467, 2003.

[18] M.D. Grossberg and S.K. Nayar. What is the space of camera response functions? In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, volume 2, pages 602–612, 2003.

[19] M.D. Grossberg and S.K. Nayar. Modeling the space of camera response functions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:1272 – 1282, October 2004.

[20] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049, first edition, 2000.

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

[22] J. Heikkila and O. Silven. A four-step camera calibration procedure with implicit image correction. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, page 1106, Washington, DC, USA, 1997. IEEE Computer Society.

[23] A. Hertzmann and S.M. Seitz. Example-based photometric stereo: Shape reconstruction with general, varying brdfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1254–1264, 2005. to be published.

[24] B.K.P. Horn. *Robot vision*. MIT Press, 1986.

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

[26] I. Horovitz and N. Kiryati. Depth from gradient fields and control points: bias correction in photometric stereo. *Image and Vision Computing*, 22(9):681–694, Aug. 2004.

[27] D. R. Hougen and N. Ahuja. Estimation of the light source distribution and its use in integrated shape recovery from stereo and shading. In *Proc. IEEE Int. Conf. on Computer Vision*, pages 148–155, 1993.

[28] K. Ikeuchi and B.K.P. Horn. Numerical shape from shading and occluding boundaries. *Artificial Intelligence*, 17(1-3):141–184, August 1981.

[29] H. Kawasaki, H. Aritaki, K. Ikeuchi, and M. Sakauchi. Image-based rendering for mixed reality. In *Proc. Int. Conf. on Image Processing*, volume 3, pages 939–942, 2001.

[30] C. Kolb, D. Mitchell, and P. Hanrahan. A realistic camera model for computer graphics. In *Proc. SIGGRAPH*, pages 317–324, New York, NY, USA, 1995. ACM Press.

[31] Y. Li, S. Lin, H. Lu, and H. Shum. Multiple-cue illumination estimation in textured scenes. In *Proc. IEEE Int. Conf. on Computer Vision*, pages 1366–1373, 2003.

[32] S. Magda, D.J. Kriegman, and T. Zickler P.N. Belhumeur. Beyond lambert: reconstructing surfaces with arbitrary brdfs. In *Proc. IEEE Int. Conf. on Computer Vision*, 2001.

[33] S. Mann and R.W. Picard. Being 'undigital' with digital cameras: Extending dynamic range by combining differently exposed pictures. Technical Report 323, M.I.T. Media Lab Perceptual Computing Section, Boston, Massachusetts, 1994. Also appears, IS&T's 48th annual conference, Cambridge, Massachusetts, May 1995.

[34] J.B. Mulligan and X.L.C. Brolly. Surface determination by photometric ranging. In *Conference on Computer Vision and Pattern Recognition Workshop*, 2004.

[35] A. P. Pentland. Finding the illuminant direction. *Journal of the Optical Society of America*, 72:448–455, 1982.

[36] A. Robles-Kelly and E.R. Hancock. A graph-spectral method for surface height recovery. *Pattern Recognition*, 38(8):1167–1186, Aug. 2005.

[37] K. Schlüns. Shading based 3d shape recovery in the presence of shadows. Technical report, University of Auckland, September 1997.

[38] M. Singh and N. Ahuja. Estimating light sources. In *Indian Conference on Computer Vision, Graphics and Image Processing*, pages 76–81, New Delhi, India, December 1998.

[39] F. Solomon and K. Ikeuchi. Extracting the shape and roughness of specular lobe objects using four light photometric stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(4):449–454, 1996.

[40] M. Stokes, M. Anderson, S. Chandrasekar, and R. Motta. A standard default color space for the internet - srgb. http://www.w3.org/Graphics/Color/sRGB, November 1996. Viewed June 8, 2005.

[41] H.D. Tagare and R.J.P. de Figueiredo. 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, February 1991.

[42] D. Terzopoulos. The computation of visible-surface representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(4):417–438, 1988.

[43] R. Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, 3:323–344, 1987.

[44] Y. Wang and D. Samaras. Estimation of multiple directional light sources for synthesis of augmented reality images. *Graph. Models*, 65(4):185–205, 2003.

[45] D. Weinshall. The shape and the direction of illumination from shading on occluding contours. *Artificial Intelligence Memo 1264*, 1990.

[46] R.J. Woodham. Photometric stereo. In *MIT AI Memo 479*, 1978.

[47] R.J. Woodham. Photometric method for determining surface orientation from multiple images. *OptEng*, 19(1):139–144, January 1980.

[48] R.J. Woodham. Photometric method for determining shape from shading. In *IU84*, pages 97–125, 1984.

[49] Y. Yang and A. Yuille. Source from shading. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 534–539, 1991.

[50] R. Zhang, P.-S. Tsai, J. E. Cryer, and M. Shah. Shape from shading: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(8):690–706, 1999.

[51] Y. Zhang and Y.-H. Yang. Multiple illuminant direction detection with application to image synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(8):915–920, 2001.

[52] J. Zhu and Y.-H. Yang. Frequency-based environment matting. In *Pacific Conf. on Computer Graphics and Applications*, 2004.

[53] D.E. Zongker, D.M. Werner, B. Curless, and D.H. Salesin. Environment matting and compositing. In *Proc. SIGGRAPH*, pages 205–214. ACM Press/Addison-Wesley Publishing Co., 1999.

Appendix A

Hardware Specifications

A.1 Display

The LCD monitor used for all experiements is a **NEC MultiSync LCD 1760NX**. The pertinent manufacturer's specifications as listed on the NEC Web site are:

```
Active Display Area:
- Horizontal: 13.3 inches / 33.79 cm
- Vertical: 10.6 inches / 27.04 cm
  (Dependent upon signal timing used)

Display Colors:
  16,194,277
  (Dependent upon display card used)

LCD Module:
- 17-inch (17.0'' viewable image size)
- active matrix
- thin film transistor (TFT)
- liquid crystal display (LCD)
- 0.264 mm dot pitch
- 260 cd/m2 white luminance - typical
- 450:1 contrast ratio - typical
- 16ms response time - typical

Recommended Resolution:
  1280 x 1024 @ 60 Hz

Tilt / Swivel:
  TILT: 35 deg (30 deg up / 5 deg down)
  SWIVEL: 140 deg (70 deg left/right)

VESA Hole Configuration Spec.
  100 x 100mm

Viewing Angle:
  Left/Right: 80 deg
  Up: 80 deg
  Down: 80 deg
  (5:1 measurement consideration)
```

A.2 Camera

The camera used for all experiments is the **Canon EOS 300D (Digital Rebel)**. The pertinent specifications as listed at the Web site http://www.dpreview.com/reviews/canoneos300d/ are:

```
Sensor
- 22.7 x 15.1 mm CMOS sensor
- RGB Color Filter Array
- Built-in fixed low-pass filter
- 6.5 million total pixels (3152 x 2068)
- 6.3 million effective pixels (3072 x 2048)
- 3:2 aspect ratio

Sensitivity
- Auto (100, 200 or 400)
- ISO 100
- ISO 200
- ISO 400
- ISO 800
- ISO 1600

Shutter
- Focal-plane shutter
- 30 - 1/4000 sec (0.3 EV steps)
- Flash X-Sync: 1/200 sec
- Bulb

File formats
- RAW (2048 x 1360 JPEG embedded)
- JPEG (EXIF 2.2)
```

A.3 Lens

A **Canon EF-S18-55mm f/3.5-5.6** zoom lens was used. Its features include:

```
Focal Length & Maximum Aperture
  18-55mm 1:3.5-5.6

Lens Construction
  11 elements in 9 groups

Diagonal Angle of View
  75 deg 20' - 27 deg 50'

Focus Adjustment
  Inner focusing system with Micro USM

Closest Focusing Distance
  0.28m / 0.92 ft. to infinity

Zoom System
  Rotating Type

Filter Size
  58mm

Max. Diameter x Length, Weight
  2.7" x 2.6", 6.07oz. / 69mm x 66.2mm, 190g
```

Appendix B

Experimental Parameters

DEPTH_ESTIMATE.....Initial depth estimate
SHADOW_THRES.....The shadow threshold
READ_RAW_DIRECT.....Set to 1 to read directly from raw file
LINEAR_CURVES.....Set to 1 if the camera curve is linear
WIDTH.....Desired pixel width of processed images
COMPONENT_EDGE_THRES.....Threshold for the individual edge images in x and y direction
EDGE_THRES.....Threshold for the combined edge images
N_PS_ITERATIONS.....Number of PS+depth iterations to make
HAS_GROUND_TRUTH.....Set to 1 if ground truth is available
DEPTH_TOLERANCE.....Depth estimate tolerance
PROFILE_COMP_LINE.....Location for the depth profile comparison
ESTIMATE_SCREEN_POS.....Flag to estimate screen position (not implemented)
IS_SYNTHETIC.....Flag to set if images are synthetic
SYNTH_WIDTH.....Width of synthetic images
SYNTH_HEIGHT.....Height of synthetic images
CX, CY, CZ, R.....Position and radius of synthetic sphere
N_CP_SQ_SIZE.....Screen calibration pattern square size
N_X_OFFSET.....Offset of calibration pattern in X
N_Y_OFFSET.....Offset of calibration pattern in Y
SCR_SIZE_MM.....Screen size in millimeters
SCR_SIZE_PIX.....Screen size in pixels

B.1 Synthetic Sphere

B.1.1 synth_sphere/session.m

```
DEPTH_ESTIMATE          = 293;
SHADOW_THRES            = eps;
%LINEAR_CURVES          = 1;
%WIDTH                  = 100;
COMPONENT_EDGE_THRES    = 0.1;
EDGE_THRES              = 0.2;
N_PS_ITERATIONS         = 4;
HAS_GROUND_TRUTH        = 1;
DEPTH_TOLERANCE         = 1.0e-5;

PROFILE_COMP_LINE       = 80;
```

```
ESTIMATE_SCREEN_POS     = 0;

% synthetic specific options
IS_SYNTHETIC     = 1;
SYNTH_WIDTH      = 3072;
SYNTH_HEIGHT     = 2048;
CX               = 0;%200
CY               = 0;
CZ               = 300;
R                = 7;

% Screen calibration pattern offset and size
N_CP_SQ_SIZE     = 100;
N_X_OFFSET       = 50;
N_Y_OFFSET       = 50;

% screen size
SCR_SIZE_MM = [337 269];
SCR_SIZE_PIX = [1280 1024];

% Region of interest
roi;
```

B.1.2 synth_sphere/roi.m

```
% ROI for centered sphere
ROI_top    =  1061 - 75;
ROI_bottom =  1061 + 75;
ROI_left   =  1513 - 75;
ROI_right  =  1513 + 75;
```

B.2 Synthetic Stanford Bunny

B.2.1 synth_bunny/session.m

```
DEPTH_ESTIMATE          = 283;
SHADOW_THRES            = 4;
LINEAR_CURVES           = 1;
WIDTH                   = 200;
COMPONENT_EDGE_THRES    = 0.08;
EDGE_THRES              = 0.2;
N_PS_ITERATIONS         = 10;
DEPTH_TOLERANCE         = 1e-5;
HAS_GROUND_TRUTH        = 1;
IS_SYNTHETIC            = 1;
PROFILE_COMP_LINE       = 80;
ESTIMATE_SCREEN_POS     = 0;

% Screen calibration pattern offset and size
N_CP_SQ_SIZE     = 100;
N_X_OFFSET       = 50;
N_Y_OFFSET       = 50;

% screen size
SCR_SIZE_MM      = [337 269];
SCR_SIZE_PIX     = [1280 1024];

% Region of interest
```

```
roi;

% Model position
t_model;
```

B.2.2 synth_bunny/roi.m

```
ROI_top    =    36;
ROI_bottom =  1457;
ROI_left   =   628;
ROI_right  =  2318;
```

B.3 Real Sphere

B.3.1 real_sphere/session.m

```
DEPTH_ESTIMATE          = 370;
SHADOW_THRES            = 3;
READ_RAW_DIRECT         = 1;
WIDTH                   = 64;
COMPONENT_EDGE_THRES    = 0.06;
EDGE_THRES              = 0.2;
N_PS_ITERATIONS         = 1;
DEPTH_TOLERANCE         = 1e-5;
HAS_GROUND_TRUTH        = 0;
IS_SYNTHETIC            = 0;
PROFILE_COMP_LINE       = 30;
ESTIMATE_SCREEN_POS     = 0;

% Screen calibration pattern offset and size
N_CP_SQ_SIZE    = 100;
N_X_OFFSET      = 50;
N_Y_OFFSET      = 50;

% screen size
SCR_SIZE_MM     = [337 269];
SCR_SIZE_PIX    = [1280 1024];

% Region of interest
roi;
```

B.3.2 real_sphere/roi.m

```
ROI_top    =   741;
ROI_bottom =  1131;
ROI_left   =  1128;
ROI_right  =  1614;
```

B.4 Real Stanford Bunny

B.4.1 real_bunny/session.m

```
DEPTH_ESTIMATE          = 280;
SHADOW_THRES            = 4;
READ_RAW_DIRECT         = 1;
WIDTH                   = 200;
COMPONENT_EDGE_THRES    = 0.06;
EDGE_THRES              = 0.2;
```

```
N_PS_ITERATIONS          = 1;
DEPTH_TOLERANCE          = 1e-5;
HAS_GROUND_TRUTH         = 1;
IS_SYNTHETIC             = 0;
PROFILE_COMP_LINE        = 80;
ESTIMATE_SCREEN_POS      = 0;

% Screen calibration pattern offset and size
N_CP_SQ_SIZE     = 100;
N_X_OFFSET       = 50;
N_Y_OFFSET       = 50;

% screen size
SCR_SIZE_MM      = [337 269];
SCR_SIZE_PIX     = [1280 1024];

% Region of interest
roi;

% Model position
t_model;
```

B.4.2 real_bunny/roi.m

```
ROI_top    =    36;
ROI_bottom =  1457;
ROI_left   =   628;
ROI_right  =  2318;
```