Depth recovery using a plenoptic camera.

Mohamed. Amtoun

University of Windsor

Follow this and additional works at: https://scholar.uwindsor.ca/etd

Recommended Citation
https://scholar.uwindsor.ca/etd/1643

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000 ext. 3208.
Depth Recovery Using a Plenoptic Camera

by

Mohamed Amroun

A Thesis
Submitted to the Faculty of Graduate Studies and Research
through the School of Computer Science
in Partial Fulfillment of the Requirements for
the Degree of Master of Science
at the University of Windsor

School of Computer Science
University of Windsor
Windsor, Ontario, Canada

April 2003
The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L’auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L’auteur conserve la propriété du droit d’auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.
Abstract

Depth recovery is a central problem in Computer Vision. Stereopsis is one of the most extensively investigated techniques amidst the approaches addressing it. Though sufficiently accurate, stereopsis is highly dependent on the success of point correspondence, an ill-posed problem that is known to suffer from a number of underlying difficulties. Single-lens stereo systems emerged recently as an effective means to thwart some of the problems of correspondence. The plenoptic camera in particular, has a special image formation geometry that claims to resolve the correspondence ambiguities and simplify depth recovery. However, to our knowledge, prior to this work, no study has ever been conducted on this system to experimentally assess its viability for depth extraction.

The research outlined in this thesis is twofold: the complete simulation process of this camera is presented, which delivers a physically-accurate rendering engine that faithfully accounts for its image formation geometry. Secondly, the depth recovery capabilities of the camera are closely examined; various configurations are tested against different types of scenes using the simulator, and a comparison framework with stereopsis is set up to outline the merits and the disadvantages of this camera. Degrading factors are studied and indicators are given as to how design parameters affect the performance of the system.
In the memory of Abdelilah Selmane...
Acknowledgements

I am primarily indebted to my advisor Dr. Boubakeur Boufama. His guidance and support were exemplary. I owe him a great debt of respect and gratitude for his supervision and personal qualities.

I also wish to thank Dr. Majid Ahmadi\(^1\) and Dr. Graham Jullien, the initial instigators of this thesis project, who co-supervised this work and provided the necessary support throughout. I hope they find my thesis useful to build upon for new research perspectives.

The project would not have been possible were it not for the financial support of Dalsa Inc.\(^2\) and Micronet R&D\(^3\). Many thanks to Dr. Gareth Ingram and Dr. Mike Miethig from Dalsa Inc. in Waterloo who did not hesitate to show their encouragement of this work from the very first day.

Many thanks go to Adlane Habed for very fruitful discussions regarding the issues that arose in the depth recovery part of this thesis.

Last yet not least, I wish to express my deepest gratitude to my first and most inspiring teachers, my parents Soumia Selmane and Ahmed Amtoun. Thank you.

\(^1\)Dr. Ahmadi supervised the simulation part of this work which was carried out jointly with Arash Razavi from the Electrical and Computer Engineering Department. His Master’s thesis is entitled Simulating the Plenoptic Camera.

\(^2\)www.dalsa.com

\(^3\)www.micronetrd.com
# Table of Contents

Abstract  iv

Dedication  v

Acknowledgements  vi

List of Tables  x

List of Figures  xi

1 Introduction  
   1.1 Description of the Plenoptic Camera  3
   1.2 Organization of the Thesis  3

2 Background  
   2.1 Geometry of Image Formation  5
      2.1.1 Pinhole imaging  5
      2.1.2 The thin lens  6
      2.1.3 The thick lens  7
   2.2 Real Camera Lenses  8
   2.3 Physically-Based Image Synthesis  8
      2.3.1 The rendering equation  10
      2.3.2 The bidirectional reflectance distribution function  12
2.4 Image Synthesis through Ray Tracing ........................................ 14

3 Passive Methods for Depth Recovery ........................................... 17
   3.1 Stereoscopic Vision ....................................................... 17
       3.1.1 The correspondence problem .................................. 19
       3.1.2 Single Lens Stereo Systems .................................. 20
   3.2 Other Passive Techniques ............................................. 23
       3.2.1 Structure from motion ........................................... 23
       3.2.2 Depth from focus ................................................. 25
       3.2.3 Depth from defocus ............................................. 27

4 A Simulation Framework for the Plenoptic Camera ....................... 29
   4.1 Solving Scene Illumination ............................................ 30
       4.1.1 Interpretation of the rendering equation .................. 31
       4.1.2 Monte Carlo global illumination algorithms ............... 32
       4.1.3 Path tracing with next event estimation using the modified Phong model .... 33
   4.2 Simulating the Plenoptic Camera ..................................... 39
       4.2.1 The pixel equation .............................................. 39
       4.2.2 Approximation of the pixel equation ......................... 40
       4.2.3 Thick lens approximation ..................................... 42
       4.2.4 Tracing rays through a thick lens ............................ 44
       4.2.5 Results ......................................................... 45

5 Depth Recovery Using a Plenoptic Camera ................................ 48
   5.1 Single-Lens Stereo ..................................................... 48
       5.1.1 Expression of depth ............................................ 50
   5.2 Disparity Analysis ..................................................... 51
   5.3 Experimental Results ................................................. 53
   5.4 Analysis and Discussion .............................................. 63
6 Conclusions

6.1 Potential Applications ........................................ 67
6.2 Future Work ..................................................... 67

Appendix A - Monte Carlo Integration ....................... 69
A.1 Overview of Monte Carlo Techniques ..................... 69
A.2 Variance Reduction Techniques ............................. 70
   A.2.1 Importance sampling ................................... 70
   A.2.2 Stratified sampling ..................................... 71

Bibliography ..................................................... 72

Vita Auctoris ................................................... 79
List of Tables

5.1 Lenticular sheet specifications retained for the experiments. ................. 55
5.2 Table recapitulating the outcome of the tests. ................................. 65
# List of Figures

1.1 Plenoptic imaging. .................................................. 4

2.1 Pinhole imaging. .................................................... 5

2.2 Image formation with a thin lens. ............................... 7

2.3 Image formation with a thick lens. .............................. 7

2.4 Snapshots of two different lenses taken from our lens visualization program. The row in italics refers to the aperture stop (not depicted in the diagrams) ........ 9

2.5 The radiance $L_r$ emitted by $x$ within a direction $\omega'$ is obtained by multiplying the BRDF of the surface $x$ belongs with the incoming radiance ($L_i$) within the illumination hemisphere surrounding point $x$ and integrating over all solid angles of the hemisphere. .................................................. 11

2.6 Illustration of the components used in the second version of the rendering equation. ................................. 13

2.7 Tracing rays using the pinhole model. ........................... 15

3.1 A normal and a general stereo configuration. ..................... 18

3.2 Triangulation. Point $P$ is projected at two different positions in the right and the left image. Disparity $d = x_r - x_l$ is linearly dependent on the distance $D$ of $P$ from the optical centers. .................................................. 18

3.3 Teoh and Zhang's catadioptric camera system. The center mirror rotates to two distinct positions. Each position being responsible for the imaging of the scene in one half of the image (left or right). ................................. 21
3.4 Nishimoto and Shirai's single-lens stereo system. The knob is used to rotate the
glass plate, so that the optical axis is shifted slightly.......................... 21
3.5 Goshtaby and Gruver camera system. \( C_1 \) and \( C_r \) are the optical centers of the two
virtual views................................................................. 22
3.6 Nene & Nayar's Catadioptric camera systems........................................... 23
3.7 The bi-prism single lens stereo camera..................................................... 24
3.8 Depth from Focus. Illustration of the components used in the expression of the
circle of confusion diameter....................................................... 26
4.1 The rendering diagram........................................................................ 30
4.2 Interpretation of the expansion into a Neumann series of the rendering equation: the
case of two reflections........................................................................ 32
4.3 Stratified sampling................................................................................ 40
4.4 N-rooks sampling.................................................................................. 41
4.5 Rays traced through a 50\text{nm} Double Gauss lens.................................. 42
4.6 Diagram of a 50\text{mm} Double-Gauss lens depicting the method used in determining
the second principal plane................................................................. 43
4.7 Tracing rays through a thick lens........................................................... 45
4.8 320 \times 200 scene rendered using a 50\text{mm} Double-Gauss lens system and 625 sam-
plies per pixel. The lens is focused on infinity, but objects are acceptably sharp due
to their distance from the camera. The farthest object is 10\text{m} away from the camera. 45
4.9 Same scene as in figure 4.8 rendered using a 100\text{mm} lens system (U.S. patent #
1,160,148) and 625 samples per pixel. Notice the zooming effect due to a higher
focal length. The objects are slightly blurred because the focus is on infinity...... 46
4.10 300 \times 300 plenoptic image of the scene in figures 4.8 and 4.9 rendered using a
50\text{mm} Double-Gauss Lens and 1225 samples per pixel. The viewpoint resolution
(nb. of subpixels) is 3 \times 3 and the spatial resolution (nb. of lenticules) is 100 \times 100. 46
4.11 Sub-images extracted from plenoptic image in 4.10. If the images along the horizontal or vertical dimension are put in an animation from left to right, the scene will appear to be moving along that dimension from right to left, since the scene is entirely located before the plane in focus (infinity).

5.1 Single lens stereo with a plenoptic camera.

5.2 Baseline between two different sub-images.

5.3 Close-range scene (CLOSERANGE1).

5.4 Sub-images extracted from the CLOSERANGE1 scene.

5.5 Disparity map of the CLOSERANGE1 scene calculated with a patch size of 11×11.

Highest disparity value: 4 pixels.

5.6 Depth plot corresponding to the CLOSERANGE1 scene.

5.7 Medium-range scene (MEDRANGE).

5.8 Sub-images extracted from the MEDRANGE scene.

5.9 Disparity map of the MEDRANGE scene calculated with a patch size of 11×11.

Highest disparity value: 2 pixels.

5.10 Depth plot corresponding to the MEDRANGE scene.

5.11 The CLOSERANGE2 scene.

5.12 Sub-images extracted from the CLOSERANGE2 scene.

5.13 Disparity map of the CLOSERANGE2 scene calculated with a patch size of 11×11.

Highest disparity value: 3 pixels.

5.14 Depth plot corresponding to the CLOSERANGE2 scene.

5.15 Depth plot obtained on a 5×5 set of sub-images representing the MEDRANGE scene.

5.16 Disparity map obtained from the analysis of a 3×3 set of sub-images from the CLOSERANGE2 scene, each with a resolution of 200×200. Highest disparity: 3 pixels.

5.17 Stereo pair of 100×100 pixels obtained with a baseline of 2 mm.

5.18 Stereo pair of 200×200 pixels obtained with a baseline of 2 mm.
5.19 Disparity map obtained from the analysis of a stereo pair with a medium and a large baseline (5 mm and 8 mm respectively). The images were 100 × 100 each. Point matching was carried out using ZNCC. *Highest disparity value: 5 pixels.*

5.20 Viewpoint aliasing with the plenoptic camera. The image of the cubes is supposed to appear exactly the same as in *image*22.
Chapter 1

Introduction

Since its early days, one of the main objectives of photography was to bring a faithful representation of the surrounding world to the viewer. The techniques used in photography were developed over time, but the representation remained a mere two-dimensional one. In fact, The 3D to 2D imaging process is a non-invertible one and as such, conventional photography cannot capture the depth component necessary to upgrade to the 3D representation needed by a multitude of industrial applications. However, it can be achieved in a passive fashion by extracting additional visual cues from 2D images, or through active techniques which perceive depth through the direction of some form of energy on target scenes. While active techniques have a better overall accuracy, passive techniques drew more research efforts because of their practicality and applicability in a variety of environments without the need for expensive equipment. Actually, a considerable amount of information can be inferred from 2D images, but this requires the combination of several images of the same scene, that either have been taken from different viewpoints or with different camera settings.

In stereo vision [DA89], one of the most investigated techniques for depth recovery, the depth of any point can be recovered in a straightforward manner when the matches between the two images have been established, that is, once every point of an image has been assigned a correspondent in a second viewpoint image. The difficulty in stereo however, does not lie in depth calculation but rather in determining the correspondences between points. Several algorithms addressing the correspondence problem were devised, most of which produce successful results to some extent,
but their success is highly dependent on the type of scene they are applied to. This makes the correspondence problem an ill-posed one. We can name a number of problems that undermine the accuracy of correspondence, and consequently lead to erroneous or inaccurate depth estimates:

1. when the intensity differences between two stereo images are large, it is unlikely that any algorithm may yield sufficiently accurate depth estimates

2. the absence of high frequency information such as edges and texture from the image biases the correspondence process

3. the inevitable geometric differences between two spatially offset images can sometimes complicate the search and make the process error-prone

4. a small baseline affects the accuracy of disparity, and when it is large, the chances that occlusion occur become higher, albeit the accuracy of depth estimates is enhanced

Research only managed to overcome some of the difficulties mentioned above. Furthermore, knowing that some of these difficulties stem from the use of two different cameras (intensity and geometrical differences), some researchers geared the bulk of their efforts towards the design of cameras called single-lens stereo cameras, which use the same camera lens to produce stereo pairs as if taken by spatially offset cameras. We particularly found that the plenoptic camera [AW92] presents at least two major advantages with regard to its counterparts: (1) it simultaneously takes several snapshots of the observed scene along the horizontal as well as the vertical dimensions, (2) the displacement between virtual sub-views takes the form of pure translation, which greatly simplifies the correspondence search.

After its invention in 1991, the plenoptic camera remained a research-domain device whose usefulness has never been assessed. In fact no experimental data has ever been released in support of its validity for solving the depth recovery problem. The lack of such information was the main motivation of this work, which has the following contributions: in the first part, a camera simulator is detailed which provides a testing framework for experimenting different types of design parameters against different types of scenes using the plenoptic camera. The second part is concerned with the study and analysis of this unique camera system from a depth perception standpoint.
1.1 Description of the Plenoptic Camera

The plenoptic\textsuperscript{1} camera was invented in 1991 [Ade91]. It belongs to the single-lens stereo system category of cameras. In a plenoptic camera, an image is formed similarly to conventional monocular cameras, with the difference that a \textit{lenticular sheet} is placed between the lens system and the sensor grid. This lenticular sheet (or array) consists of a battery of small lens elements termed \textit{lenticules}, which transform light directed through the aperture of the camera lens (henceforth referred to as the main lens) before it reaches the sensor grid. Every lenticule captures a chunk of the light directed through the aperture of the main lens, and images it onto the group of underlying pixels (termed \textit{macropixel}). Figure 1.1(a) shows a side view of the plenoptic camera.

In a regular camera, light coming from different directions is summed and averaged as it hits a pixel regardless of its angle of incidence. A lenticular sheet placed between the lens and the sensor grid, allows the plenoptic camera to keep track of the structure of light impinging on its sensor grid.

Similarly to stereo, the 3D information that can be extracted from the sub-images takes the form of \textit{parallax} and hence may be exploited to extract depth. Furthermore, the sub-images in question provide us with both virtual and horizontal parallax. The number of macropixels defines the spatial resolution of the subimages, while the number of subpixels defines the number of subimages to be extracted from the "raw" plenoptic image (figure 1.1(b)); for example, a $500 \times 500$ plenoptic image obtained using a $50 \times 50$ lenticular sheet, will result in a $10 \times 10$ grid of subimages each with a resolution of $50 \times 50$ pixels.

1.2 Organization of the Thesis

In order to lay the ground for a full understanding of the simulation process exposed further in this thesis, we begin in chapter 2 by exposing the necessary geometrical and mathematical background, namely, computer graphics techniques for realistic image synthesis, and details on image formation in real-world cameras as well as simplified lens models. Chapter 3 provides an overview of the most common passive depth recovery techniques in computer vision and a survey of single

\textsuperscript{1}plenoptic is derived from \textit{plenus} meaning complete
lens stereo cameras that were devised recently. In Chapter 4, we expose the techniques used to simulate the plenoptic camera along with the outcome of the simulator: images representing what would be viewed by the plenoptic camera in a synthetic environment. Finally, chapter 5 outlines the depth recovery part of this work which builds upon results obtained from the simulator; details are given as to the viability of the plenoptic camera for depth recovery purposes, its strengths and its weaknesses.
Chapter 2

Background

2.1 Geometry of Image Formation

This section reviews the fundamentals of image formation, which will be used in subsequent parts of this dissertation. In particular, details about the geometry and the radiometry of image formation will be presented.

2.1.1 Pinhole imaging

The pinhole camera model is a simple approximation of image formation within real world cameras. In this model, the camera is reduced to a pinhole which is a minute point, through which each point from the scene projects onto a single point on the image. Considering figure 2.1, let $p(x, y, z)$
be the projection of point \( P(X, Y, Z) \) on the image plane. The image plane being separated from the pinhole (or center of projection) \( O \) by a focal length \( f \), we have \( z = f \) and:

\[
\frac{x}{X} = \frac{y}{Y} = \frac{f}{Z}
\]

hence,

\[
\begin{align*}
  x &= f \frac{X}{Z} \\
  y &= f \frac{Y}{Z}
\end{align*}
\]  \hfill (2.1)

Since every scene point is imaged onto a single point on the image plane, the images obtained are in clear focus regardless of their distance from the camera.

2.1.2 The thin lens

A more realistic model for image formation can be obtained by a thin lens approximation. In a thin lens, the image of a point depends on the its distance to the camera. The thin lens model is governed by lens law which writes:

\[
\frac{1}{f} = \frac{1}{D} + \frac{1}{v}
\]  \hfill (2.2)

where \( f \) is the focal length of the camera, \( D \) is the distance between the lens and the observed object point and \( v \) is the distance between the sensor plane and the lens (also called focal distance). Lens law suggests that only one single plane can be in focus at a time, therefore chances are some parts of the image will appear on several points\(^1\) on the image forming a blur circle (see figure 2.2). The thin lens can be considered as a generalization of the pinhole camera since the same formula presented earlier holds for the present model, with the only difference that due to a larger aperture, a scene point is imaged from light coming from different directions rather than a single direction. As can be seen on figure 2.2, the light that emanates from a scene point is converged by the lens towards a single point on a plane whose distance from the lens \( v \) is determined by lens law. This also tells us that points at infinity will appear in focus at distance \( v = f \) from the lens.

\(^1\)unless the scene consists of one single plane
CHAPTER 2. BACKGROUND

![Image of image formation with a thin lens]

Figure 2.2: Image formation with a thin lens.

2.1.3 The thick lens

Although the thin lens offers a more concise modelling of image formation compared to the pinhole model, it is not suitable for lenses whose thickness is not negligible. In this case, the thick lens model (figure 2.3) should be used. A thick lens is characterized by two principal planes $PP$ and $PP'$ between which light is translated before being converged. $PP$ and $PP'$ are separated by the lens' effective thickness $t$. Similarly to the thin lens, the perspective projection equations hold for the thick lens, and lens law is applicable here too with the difference that $D$ and $v$ are calculated from $PP$ and $PP'$ respectively, as illustrated on figure 2.3.

![Image of image formation with a thick lens]

Figure 2.3: Image formation with a thick lens.
2.2 Real Camera Lenses

Real camera lenses are made of a plurality of simple lenses that generally have a planar or spherical surfaces. These surfaces are arranged in a way that reduces some of the unwanted artifacts that characterize simples lenses such as aberration and stigmatism. Modelling the behavior of a lens system through a lens model such as the thin lens or thick lens models provides a decent approximation for applications where precision is not a must. These models also suppose ideal image formation, and are not suitable for such designs as wide angle lenses. Physical modelling on the other hand remains the best way to simulate the behavior of a real lens system. Actually, lens designers have been using physical modelling through paraxial ray tracing for centuries and it is still the most efficient way to account for all the effects exhibited by lenses. Some examples of lenses along with their parameters are illustrated in figure 2.4; each row refers to the parameter set of a single lens sub-element. \textit{Radius}: the radius of curvature of the lens sub-element. \textit{IOR}: the index of refraction which characterizes the material of the lens sub-element. \textit{Diameter}: the diameter of the lens sub-element. \textit{Width}: the distance that separates two consecutive sub-elements; the width of last sub-element refers to the distance that separates it from the image plane. The aperture stop (diaphragm which regulates the amount of light let through the lens system) is represented with a zero radius curvature. In most optics books, only the formulae that govern paraxial ray tracing (i.e. two dimensional) through lens systems are presented. However, those formulae turn out to be insufficient for the context of our simulation, since they cannot be applied to non-paraxial rays. A simple and accurate way of tracing rays through lenses will be outlined in the simulation chapter.

2.3 Physically-Based Image Synthesis

While the first section reviewed the fundamentals of image formation geometry, this section provides an overview of the fundamental concepts of image formation from a radiometric perspective.

An image forms as a result of light reflected by objects striking a photosensitive surface such as a film plane or an array of digital image sensors. Light can be perceived as a dense bundle of

\footnote{Using only rays which are parallel to the optical axis}
Figure 2.4: Snapshots of two different lenses taken from our lens visualization program. The row in italics refers to the aperture stop (not depicted in the diagrams).
CHAPTER 2. BACKGROUND

Infinitesimal rays that carry a charge of energy in the form of photons.

Digital photography is based on the photoelectric effect, which stipulates that when light rays (photons) strike a photosensitive surface, a resulting electrical charge (electrons) is generated. This electrical charge, which is proportional to light brightness, is internally converted by the digital camera to an analog signal. The latter is then converted via a frame grabber to a digital representation consisting of picture elements (pixels) each of which encodes the color of the smallest elements of the image. Depending on the type of filters used in the digital camera, the pixels can be simple gray level values (ranging between 0 and 255) or a structure of data representing a color. Color is most commonly represented using the RGB (Red, Green and Blue) triplet representation (also known as the tristimulus values).

2.3.1 The rendering equation

To study light behavior in a scene of objects illuminated by one or a plurality of light sources, we start by a brief definition of the notion of radiance. Radiance is a physical quantity that describes the amount of energy (photons) hitting an infinitesimal patch of an object in space. Radiance is defined for a position \( x \) and a direction \( \omega \) and is denoted \( L(x, \omega) \).

Objects can emit light into space either directly if they are light sources, or indirectly if they reflect light received from one or several light sources. This leads us to the notion of irradiance, which defines the amount of light received by a point \( x \). The irradiance is simply obtained by integrating over a hemisphere centered at point \( x \) termed the illumination hemisphere.

When light is received from a direction within the illumination hemisphere, some of it is absorbed while the rest is reflected depending on the properties of the surface. Therefore by knowing the surface properties, the radiance of point \( x \) can also be calculated. This reflection phenomenon is modelled by the following equation:

\[
L_r(x, \omega_r) = \int_{H} \rho_{rd}(x, \omega, \omega') L_i(x, \omega) \cos \theta d\omega \tag{2.3}
\]

where \( L_i \) is the incoming radiance, \( L_r \) is the reflected radiance and \( \rho_{rd}(x, \omega, \omega') \) is the Bi-directional
reflectance distribution function (BRDF), which defines the probability that light coming from direction \( \omega \) reflects over point \( x \) in direction \( \omega' \) (see figure 2.5). The BRDF is a function that describes the reflectivity characteristics of a given surface. Although the BRDF depends on the wavelength, we hereby limit our study to a single monochromatic wavelength for the sake of simplicity, and will relax this restriction when color is considered.

![Diagram](image)

Figure 2.5: The radiance \( L_r \) emitted by \( x \) within a direction \( \omega' \) is obtained by multiplying the BRDF of the surface \( x \) belongs with the incoming radiance \( (L_i) \) within the illumination hemisphere surrounding point \( x \) and integrating over all solid angles of the hemisphere.

By introducing the emitted radiance \( L_e \), we can also account for light emitting objects (light sources), hence, the radiance of point \( x \) becomes:

\[
L(x, \omega) = L_e(x, \omega) + L_r(x, \omega) \tag{2.4}
\]

Supposing light travels in a vacuum, radiance remains constant along its travelling direction, therefore, the incident radiance coming from point \( x' \) along direction \( \omega \) over point \( x \) is equal to the radiance of \( x' \), that is

\[
L_i(x, \omega) = L_i(x', -\omega) = L_i(h(x, -\omega), -\omega)
\]

where \( h(x, -\omega) \) determines the closest point from \( x \) in direction \( -\omega \). This leads us to what is known
as the rendering equation:

\[
L(x, \omega) = L_e(x, \omega) + \int_B L(h(x, -\omega'), \omega') \cdot \rho_{bd}(x, \omega, \omega') \cdot \cos \theta' \cdot d\omega'
\]  

(2.5)

The rendering equation formalizes the global illumination problem and provides a framework for solving it. It exists in two forms in the computer graphics literature: the first form (as defined in Eq. 2.5) is due to Immel [ICG86] and is defined as the integral over all solid angles in the hemisphere, while another form due to Kajiya [Kaj86], is written as an integral over all surfaces in the scene (see figure 2.5):

\[
L(x, \omega) = L^s(x, \omega) + \int_A \rho_{bd}(x, \omega, \omega') L(x', \omega') G(x, x') V(x, x') dA'
\]  

(2.6)

where \( V(x, x') \) is a visibility term that takes 0 if points \( x \) and \( x' \) are occluded by a non transparent object and 1 otherwise. \( G(x, x') \) is a geometry term defined as follows

\[
G(x, x') = G(x', x) = \frac{\cos \theta \cos \theta'}{|x - x'|^2}.
\]

The rendering equation exists under other forms in the computer graphics literature which do not differ much from the two equations defined above. Regardless of how it may be defined, the rendering equation is an integral equation in which the unknown \( L \) appears on both sides, and inside the integral. Equations of this type are called Fredholm equations of the second kind [Atk76]. They have no analytical solution but a formal solution can be written for them.

2.3.2 The bidirectional reflectance distribution function

The BRDF defines a mathematically and physically correct model for the reflective properties of a surface. At their simplest level, surfaces can be diffuse, in which case incident light is reflected evenly over all directions within the hemisphere (for e.g. cotton cloth), or specular, in which case, light is reflected symmetrically with respect to the normal at the point of reflection (for e.g. a mirror, polished metal, etc.). However, most surfaces are modelled as a combination of both components.
Figure 2.6: Illustration of the components used in the second version of the rendering equation.

(diffuse and specular).

Given the BRDF of a certain surface, we are able to determine the intensity of light reflected at each direction within the hemisphere surrounding each point of that surface. A BRDF can be represented using an analytical function or as a set of raw data; in the first case, determining the outcome of the function for a given set of parameters (point, incident direction and reflected direction) is straightforward, however, in the second case, this is done using interpolation.

A renderer makes use of a reflectance model. The reflectance model defines a set of BRDFs, usually in a simple way, however, does not guarantee their correctness. In global illumination algorithms, a physically accurate reflectance model is necessary for achieving convergence. Probably the most commonly used model in computer graphics is the Phong model [Pho75]. Other more or less sophisticated models include [Bli77] [Kaj85][CT82]. In the category of physically correct models, we mainly find He’s model [HTSG91] and Lafontaine’s modified Phong model [LW97].

For a reflectance model to be successfully used in a global illumination algorithm, it must be physically plausible. In other words, the following conditions must be satisfied:

- The direction of light may be interchanged:

\[ \rho_{bd}(x, \omega, \omega') = \rho_{bd}(x, \omega', \omega). \]

This is called the Hemholtz reciprocity constraint.
CHAPTER 2: BACKGROUND

- To ensure conservation of energy, the total reflected radiance has to be less or equal to 1, that is:

\[ \int_{U} \rho_{bd}(x, \omega, \omega') \cos \theta' d\omega' \leq 1. \]

2.4 Image Synthesis through Ray Tracing

The goal of image synthesis is to produce realistic looking images reflecting what is seen by a measuring device (camera) (We henceforth consider the case of a digital camera since it is of interest for us). Image synthesis algorithms render a virtual scene based on its description, by calculating the intensity generated at each pixel of the observing measuring device. This is made possible using a wide variety of techniques, each of which has its advantages and limitations. We limit our overview here to a small subset of these techniques: those based on the ray paradigm. A ray describes the trajectory of a photon travelling in space before reaching the eye. The concept of ray tracing is particularly suited for our goal (simulating a real world camera model), since we can follow all light paths to the camera and measure their impact on the sensors.

*Ray tracing* is an image synthesis technique that uses rays to model light behavior in a virtual scene, and its eventual effect on the observing camera. The first variant was devised by Appel [App68], well before the rendering equation was introduced in the computer graphics area as a unifying framework for all physically accurate image synthesis techniques. Ray tracing is a very efficient rendering technique, however, it is computationally extensive.

The capabilities of a ray tracing algorithm can be described using a notation based on regular expressions, which was introduced by Heckbert in [Hec91]. Any type of ray tracing scheme can be expressed in the form of a regular expression according to the type of light interactions it supports. We distinguish the following components:

- **E**: the eye
- **L**: the light source
- **D**: a diffuse surface
• $S$: a specular surface

The generic shape of a simple ray tracer is outlined in Algorithm 1. Figure 2.7 illustrates the way a ray is traced using a pinhole camera model. Using this camera model, perfectly focused images are produced and thus, many of the artifacts that are produced by real world camera models are ignored. Other more elaborate tracing schemes that account for real camera artifacts are exposed later in the context of global illumination algorithms. Note that the SampleRay() and Trace() functions are specific to the rendering algorithm used. Image synthesis solutions based on the ray paradigm can

alg. 1 A generic ray tracing algorithm.

\[
\text{for each pixel } p \text{ do} \\
\quad \text{intensity } \leftarrow 0 \\
\quad \text{for } i = 1 \text{ to } n\text{samples do} \\
\quad \quad \text{ray } \leftarrow \text{SampleRay}(p) \\
\quad \quad \text{sampleIntensity } \leftarrow \text{Trace}(\text{ray}) \\
\quad \quad \text{intensity } \leftarrow \text{intensity } + (\text{sampleIntensity}/n\text{samples}) \\
\quad \text{end for} \\
\quad \text{StorePixelValue}(p, \text{intensity}) \\
\text{end for}
\]

Figure 2.7: Tracing rays using the pinhole model.

be classified into three distinct categories:

• Appel’s ray tracing: is a local illumination\(^2\) technique that traces paths of the form $EDL$.

\(^2\)as opposed to global illumination which will be detailed later
This method simplifies the problem by tracing the first ray only (the primary ray), and determining shading by sending a ray to the light source.

- **Whitted-style ray tracing**: or recursive ray tracing traces paths of the form $E[S^r]DL$. Before the rendering equation was introduced under its two known forms in 1986 by Immel [ICG86] and Kajiya [Kaj86], a rendering technique known as ray tracing was presented in 1980 by Whitted [Whi80]. It uses an improved version of the Phong [Pho75] illumination model for calculating light reflection. While it provides support for ideal reflection and refraction, it is not considered a global illumination technique, mainly because the only global component that is calculated is the specular one, as for the diffuse component, it is directly sampled from the light source by sending shadow rays. As such, many lighting interaction situations are not handled.

- **Approaches based on a solution to the rendering equation**: these are the global illumination techniques that follow a $E[D^r]S^rL$ rendering scheme. That is they account for all types of light interactions. They calculate the exact behavior of light in a scene, and thus produce physically accurate images. The first technique in this category was distributed ray tracing [CPC84], which is often regarded as an extension of Whitted-style ray tracing. Other techniques include: path tracing, bidirectional path tracing and Metropolis path tracing, all of which are direct approximations of the rendering equation. Unlike local illumination algorithms which are deterministic (i.e. trajectory of rays is predictable to some extent), global illumination techniques are probabilistic; starting from a point on the pixel-grid, they attempt to reduce the ray space (of infinite size) to a finite set of rays that includes only those rays whose contribution is most important to the lighting of the scene, using stochastic techniques.
Chapter 3

Passive Methods for Depth Recovery

Depth (or range) is one of the most researched subjects in the field of computer vision. In this chapter, we are interested in passive methods which operate on 2D images to extract visual cues that indicate the depth of objects. Passive techniques fall into two distinct categories: one in which the parallax cue is exploited, and another which operates on the blur cue. This chapter reviews four of the most commonly used techniques in the area of computer vision, namely, stereopsis, structure from motion, depth from focus and depth from defocus. The mode of operation of each method as well as its strengths and weaknesses are hereby outlined. In the framework of the presentation of stereo, we also review single-lens stereo cameras which, like the plenoptic camera, exploit the parallax cue and provide a geometrical solution to some of the problems of stereopsis.

3.1 Stereoscopic Vision

Stereopsis or stereo vision uses information from two different views of a scene to recover the depth of visible objects. Stereopsis is based on the notion of disparity, which marks the displacement of a point in one image with respect to its correspondent in the second image. In a normal stereo setting (i.e. two cameras having collinear X axes and parallel Y and Z axes) as opposed to a general stereo configuration where the movement between the two cameras is an arbitrary one. Both configurations are illustrated in figure 3.1, where $O_r$ and $O_l$ represent the center of projection of the right and left
camera. In the general stereo configuration, the epipolar line is the intersection of the epipolar plane (formed by point $P$ and the optical centers $O_r$ and $O_l$) with the image plane. In a normal setting, the epipolar line in the second image is simply the horizontal scanline which bears point $p$. The depth $D$ of point $P$ can be recovered through triangulation (figure 3.2):

$$D = \frac{fB}{d}$$

(3.1)

where $d = x_r - x_l$ is the disparity between the projection of point $P$ in the left and right images, and

![figure 3.1](image)

(a) Normal Configuration  (b) General Configuration

**Figure 3.1:** A normal and a general stereo configuration.

![figure 3.2](image)

**Figure 3.2:** Triangulation. Point $P$ is projected at two different position in the right and the left image. Disparity $d = x_r - x_l$ is linearly dependent on the distance $D$ of $P$ from the optical centers.

$B$ is the baseline of the stereo system (i.e. the distance between the two centers of projection of the images). Therefore, once disparity is calculated, depth can be recovered in a straightforward manner.
using 3.1. However the major difficulty of stereopsis lies in finding the disparities between the images, this is what is referred to as the correspondence or matching problem [WAH92][ZDFL94].

3.1.1 The correspondence problem

Matching a pair of points is usually achieved using correlation techniques which operate on pixel intensities to establish pixel-wise correspondences. Among the most commonly used techniques, we find ZNCC (Zero Mean Normalized Cross-Correlation) and SSD (Sum of Square Differences). The likelihood of a match is calculated over a window surrounding the pixel of interest by applying a similarity measure criterion; in the case of ZNCC, the similarity measure criterion between a window of size $W$ around $x$ in image $f$ and another window around $x + dx$ in image $g$ writes:

$$ZNCC(x, x + dx) = \frac{\sum_{\Delta \in W} (f(x + \Delta) - \bar{f}(x)) (g(x + dx + \Delta) - \bar{g}(x + dx))}{\sqrt{\sum_{\Delta \in W} (f(x + \Delta) - \bar{f}(x))^2 \sum_{\Delta \in W} (g(x + dx + \Delta) - \bar{g}(x + dx))^2}}$$

where $\bar{f}$ and $\bar{g}$ refer to the mean value of the selected window resp. in $f$ and $g$, $\Delta = (i, j)$ and $W = \{(i, j) | -n \leq i \leq n \text{and} -m \leq j \leq m\}$ (i.e. the window has a size of $n \times m$ pixels).

The SSD similarity criterion writes:

$$SSD(x, x + dx) = \frac{\sum_{\Delta \in W} (f(x + \Delta) - g(x + dx + \Delta))^2}{nm}$$

The search of a point correspondent is performed along a line called the epipolar line (figure 3.1) and consists in maximizing the value of the similarity criterion when using ZNCC and in minimizing it in the case of SSD. The epipolar line is found using the epipolar constraint [Har97] which stipulates that the correspondent of a point lies on a line in the second image whose equation is in function of the relative orientation parameters between the two views\(^1\). The correspondence search is highly simplified if the relative positions of the two camera are known. For instance, in a normal stereo setting, the epipolar line on which the search is performed is the horizontal scanline on which the point lies. In a general stereo setting, we either rectify the images [LZ99] so that the two planes

\(^1\)which are encapsulated in a structure called the fundamental matrix
become axis aligned or we find the epipolar line through feature matching. It is worth noting that the epipolar constraint can be reinforced by other constraints that restrain the correspondence search space even more and reduce the matching complexity.

Stereopsis is based on the pinhole camera model and supposes the knowledge of the camera parameters. When these are not known, the problem requires the calibration of the cameras either explicitly using a pattern [FT87] or from a sequence of images (usually three and more) [Fus00]. A detailed review on stereopsis can be found in [DA89].

Stereopsis is probably one of the most commonly used passive range recovery methods. However, it is highly dependent on the validity of the matching process which is an ill-posed problem. Moreover, when the baseline is large, the occlusion problem sets in and makes the correspondence search more ambiguous. The best results are obtained with medium size baselines which are large enough to yield a good accuracy and small enough to minimize the impact of occlusion.

3.1.2 Single Lens Stereo Systems

Single-lens camera systems are a viable solution that suppresses some of the problems of conventional stereo: geometric and intensity differences that undermine the correspondence problem. It is worth noting that the originality of those systems merely lies in the fact that they suppress the need for post-processing operations such as rectification or intensity homogenization between stereo views; in fact, they are expected to yield better results with existing correspondence techniques. In what follows, a brief overview of these systems is presented.

Teoh and Zhang’s system

Teoh and Zhang’s system [TZ84] is one the first single lens stereo systems ever proposed for the stereo problem. It is based on a camera system observing a rotative mirror placed in front of the camera lens and two fixed mirrors on the sides of the camera (figure 3.3). The rotative mirror shifts to two different rotational positions to reflect incoming light from either one of the fixed side-mirrors resulting in a left and right image with parallel optical axes. Since the images are not captured simultaneously, the camera requires that the scene be static throughout the imaging
process.

Figure 3.3: Teoh and Zhang’s catadioptric camera system. The center mirror rotates to two distinct positions. Each position being responsible for the imaging of the scene in one half of the image (left or right).

Nishimoto and Shirai’s system

Nishimoto and Shirai obtain a stereo pair from a single lens by placing a rotative plate of glass in front of the camera (figure 3.4). This glass plate can be rotated so that it slightly deviates the optical axis to two different positions, hence, allowing the same scene to be imaged from two virtual viewpoints. This camera system requires the scene to remain static while the two images are captured due to the movement of the glass plate. The resulting stereo images have a small disparity, so correspondence is simplified and occlusion artifacts are minimized.

Figure 3.4: Nishimoto and Shirai’s single-lens stereo system. The knob is used to rotate the glass plate, so that the optical axis is shifted slightly
**Chapter 3. Passive Methods for Depth Recovery**

**Goshtaby and Gruver's system**

In this camera system [GG93] (figure 3.5), two mirrors are placed next to one another and are separated by an angle $\alpha$ to simulate the imaging process of a virtual stereo pair with two converging optical axes. This camera system presents an extra advantage with respect to its counterparts: the field of view of the camera can be adjusted by modifying the angle that separates the two mirrors. The angle between the two camera being known, the projection center can be deduced and the images formed can easily be rectified to simplify the correspondence search.

![Goshtaby and Gruver camera system](image)

Figure 3.5: Goshtaby and Gruver camera system. $C_l$ and $C_r$ are the optical centers of the two virtual views.

**Nene and Nayar's stereo systems**

Nene and Nayar [NN88] propose 4 different catadioptric systems that allow the capture of two different images with single lens using different types of mirrors: planar, hyperboloidal, ellipsoidal and paraboloidal. As illustrated in figure 3.6, the mirrors in each camera provide two effective viewpoints $v$ and $v'$, which are used to project point $M$ onto $m$ and $m'$ on the image plane. Point $C$ represents the actual optical center of the perspective camera used. In the paraboloidal configuration, an orthographic camera is used instead. The mirrors are arranged in such a way that the field of view is wide.

The system is a generalization of Goshtaby's camera that supports a wider range of mirror shapes. The authors derive an expression of the epipolar constraint for each type of camera to solve

---

*systems that combine refractive and reflective surfaces (a lens and a mirror for instance*
the correspondence problem. The main advantage of the single-lens catadioptric systems is the large field of view that can be obtained, which allows for a more accurate depth perception to take place.

![Diagram of catadioptric camera systems](image)

Figure 3.6: Nene & Nayar's Catadioptric camera systems.

The bi-prism stereo camera

Similarly to its counterparts, the bi-prism stereo camera system [LKR99] delivers stereo images using a single lens in a single shot. Unlike the previous camera systems which delivers images of a general stereo setting, the bi-prism stereo system delivers axis aligned images, therefore, corresponding points fall on the same scanline, which simplifies the correspondence search. As illustrated in figure 3.7, \( m_l \) and \( m_r \) are the respective projections of virtual points \( X_l \) and \( X_r \) onto the image plane. The angle \( \alpha \) and the index of refraction of the bi-prism intervene in the determination of the deviation \( \delta \), which in turn determines the size of the baseline, and consequently, the amount of disparity.

### 3.2 Other Passive Techniques

#### 3.2.1 Structure from motion

In many settings, an observed scene may not be static and some objects in it may be subject to rigid motion\(^3\). We refer to the methods that infer range information from motion as structure from motion (SFM) techniques. There are different configurations in which SFM can be applied; either

\(^3\)All the points of the scene have the same rotation and translation parameters with respect to the observing camera
CHAPTER 3. PASSIVE METHODS FOR DEPTH RECOVERY

![Diagram of a bi-prism single lens stereo camera.]

Figure 3.7: The bi-prism single lens stereo camera.

the camera is moving and the scene is static, the scene is in motion and the camera is static or a combination of both.

If we are capturing images that are sampled at a very high rate, the sequence thus obtained represents a series of images where both the camera and the scene may be moving. Although the relative rigid motion between the scene and the camera is not a unique one\(^4\), the depth of a given point in the scene can be recovered if its motion parameters are estimated. These parameters are encompassed in a 2D vector called the motion field, denoted \( v \) henceforth which represents the projection of the velocity vector \( V \) onto the image plane. \( v \) can be estimated either through differential techniques or feature matching. In the former set of techniques motion field is approximated\(^5\) with optical flow, a vector corresponding to the apparent movement of the brightness pattern of the image, and subject to the brightness constancy constraint\[^HS81\]:

\[
(\nabla I)^T v = -I_t
\]

where \( I_t \) is the partial derivative of point \( I(x, y, t) \), \( t \) being the time at which the frame was taken, and \( \nabla I \) is the spatial image gradient. Using the brightness constancy constraint, \( v \) can be recovered

\(^4\)Not all the points are subject to the same relative motion
\(^5\)In some cases, optical flow may not approximate properly the motion field
using linear least squares [LK81]. After the optical flow is estimated, structure (depth $Z$) can be recovered [DA92] in a straightforward manner using:

$$
v = \begin{pmatrix} \frac{e_y}{f} & -f & \frac{e_z}{f} & y \\ f & \frac{e_x}{f} & -\frac{e_y}{f} & -x \end{pmatrix} R + \frac{1}{Z} \begin{pmatrix} -f & 0 & x \\ 0 & -f & y \end{pmatrix} T
$$

where $T$ and $R$ resp. are the translational and rotational components of camera motion and $f$ is the focal length of the camera.

Unlike optical flow estimation which results in a dense estimation of motion field, feature matching techniques using such correlation methods as SSD and ZNCC, as in stereo, to set up a sparse map of feature correspondences based on information gathered from only two frames. Using the Kalman filter [MKS89] feature points can be tracked over a large sequence of images. The advantage of the feature tracking technique lies in its independency from motion parameters; actually, the rigid motion between the scene and the frames needs not be known. A method that uses an orthographic camera model was introduced in [TK93], in which the translational component of the orthographic projection is eliminated, and the motion parameters as well as depth are extracted simultaneously up to an arbitrary rotation only from feature points.

It is worth noting that the optical flow and the feature tracking methods both constrain the motion between the scene and the camera to be unique. In the case of multiple motions within the scene [Adi85], the problem requires the segmentation of the scene into regions with homogeneous motion[BBAM97][WA94], which complicates the depth inference process. Variations of the two motion analysis paradigms presented above can be found in [Zha95] and more recent research is presented in [Har01][DSTT00].

### 3.2.2 Depth from focus

*Depth from focus* (DFF) operates on the principle that the amount of defocus (or blur) is a function of range. From lens law (equation 2.2), one can deduce that depth $D$ of a point can be recovered if one were to know the camera parameters (focal length and focal distance) and when the point is in focus. The latter condition is particularly hard to verify automatically and is the central issue in the
research that deals with DFF.

It can be shown that a point in defocus is imaged onto a circular region on the image called the blur circle or circle of confusion whose diameter \( d \) writes:

\[
d = \frac{Rf}{D - f} \left( 1 - \frac{D}{D_0} \right)
\]

where \( R \) is the diameter of the lens. \( d \) is positive if \( D < D_0 \) and negative if \( D > D_0 \), \( D_0 \) being the distance to the focal plane (figure 3.8).

![Figure 3.8: Depth from Focus. Illustration of the components used in the expression of the circle of confusion diameter](image)

DFF operates by varying \( f \) or \( v \) or a combination of both [Sub88] to bring a surface in the scene to a focus. Once the values of \( f \) and \( s \) are found, depth can be calculated in a straightforward manner using the lens formula. In practice, this requires the acquisition and processing of a large number of images (more than 10); the difficulty lies in finding the set of camera parameters when focus is achieved. The point spread function (PSF) of a camera is usually approximated with a 2D Gaussian:

\[
g(x, y) = \frac{1}{2\pi \sigma^2} e^{-\frac{1}{2} \left( \frac{x^2 + y^2}{\sigma^2} \right)}
\]  

(3.2)
where $\sigma$ is the standard deviation of the Gaussian given by $\sigma = kd$, $k$ being a characteristic parameter of the camera that can be determined with a calibration step [Sub88].

Since blurring can be achieved with a mere low pass filter, the amount of blur can be recovered by convolving the image with a high-pass operator designed to measure the high frequency components in the image (for e.g. the Laplacian [Kro87]). The operator is applied on the neighborhood of a given pixel for each image of the sequence, then depth is recovered using the parameters of the frame that exhibits a maximum sharpness measurement according to the operator used. A detailed review on research on DFF can be found in [Jar83] and more recent research can be found in [NN94][SC95][JG94].

Range from focus involves the use of a single camera only, and unlike stereo, it does not deal with the correspondence problem. However, the acquisition process is a major burden, since a large sequence is needed (10 or more). This involves the mechanical movement of the internal camera components, therefore the scene must remain stationary throughout the entire process.

### 3.2.3 Depth from defocus

Similarly to DFF, *Depth from Defocus* (DFD) is a monocural technique which is based on the principle that the amount of defocus is a function of lens parameters and scene depth, hence, if it can be recovered then the distance to objects in the scene can be estimated [Pen82]. The blur parameter $\sigma$ can be written as:

$$D = \frac{vf}{v - f - \sigma f_{ap}}$$

(3.3)

where $f$ is the focal length, $f_{ap}$ is the f-number of the lens, $D$ is the depth of the observed point and $v$ is the focal distance (see figure 3.8).

DFD methods do not need to perform any search to find the perfect focused image in an acquired sequence, rather, they operate a minimum of two images [Sub88] taken with different camera parameters (usually apertures are changed, but other parameters such as the focal distance or focal length can be changed as well) to recover the amount of defocus in either one of the images and
consequently apply equation 3.3 to obtain the absolute depth value. The PSF of the camera is usually approximated with a Gaussian, and hence the defocus parameter \( \sigma \) (see equation 3.2) can be recovered by inverse filtering.

It was shown [Sub88] that the blur parameters \( \sigma_1 \) and \( \sigma_2 \) of two defocused images are related by the following relation:

\[
\sigma_1 = \alpha \sigma_2 + \beta
\]

where \( \alpha \) and \( \beta \) are constants related to the camera settings.

To recover \( \sigma \) Pentland [Pen87] deconvolves the images in the frequency domain to obtain \( \sigma \), but his method presupposes that one of the two images is acquired with a very small aperture (pinhole aperture). However, more recent techniques do not impose any constraint on the aperture size of the images and use a variety of spatial and frequency filters [Sub88][EL93][XS93] [SS93]. A state of the art review on DFD can be found in [CR99].
Chapter 4

A Simulation Framework for the Plenoptic Camera

Simulation is a very effective means for studying the behavior of a system. It is ideal for hardware testing, engenders low development costs and allows for a multitude of manipulations to take place in a system before it is manufactured. R.E. Shannon defines simulation as:

...the process of designing a model of a real system and conducting experiments with this model for the purpose either of understanding the behaviour of the system or of evaluating various strategies (within the limits imposed by a criterion or set of criteria) for the operation of the system.

Owing to its prohibitive computational complexity, it is only recently that simulation of camera systems through realistic image synthesis emerged as an effective means to realistically test vision systems [KMH95][Par97].

There are two inherent aspects to image formation, namely, the path light follows before reaching the sensor sheet (scene illumination) and the measurement of the intensity of light striking the image plane (the imaging process) (see diagram 4.1). Thus, implementing a simulator that mimics the behavior of the plenoptic camera system requires that both aspects of image formation be accounted for.
A simulated system operates in an environment that has to be set up in a proper manner, to allow it to operate fully. The hosting environment of a plenoptic camera is its surrounding world which serves as an input. Providing a real-world scene as input to a virtual camera can be achieved through a 3D scene description based on objects and lighting specifications. In other words, the plenoptic camera simulator will be implemented as a renderer that, given a set of objects and light sources, produces synthesized plenoptic images according to the specifications of the camera. Therefore, the implementation of the simulator suggests that the following two subproblems be solved:

- Solving scene illumination by modelling light interaction between objects in the observed scene (the global illumination problem)
- Calculating the intensity of light impinging on the camera sensors

Details on how to solve these subproblems and their related issues are discussed throughout this chapter.

![Diagram](image.png)

**Figure 4.1: The rendering diagram.**

### 4.1 Solving Scene Illumination

Given a scene description consisting of a set of objects and light sources, scene illumination calculation is carried out by applying a lighting model to the surfaces of all visible objects from within the capturing device (camera). In applications that directly deliver images for user appreciation, the illusion of life-like images is easily achieved by applying simple shading models that are fast enough to be used even in real-time applications. However, obtaining physically accurate images
requires a correct calculation of all light interactions between objects and light sources in a scene. This is known as the global illumination problem.

Global illumination refers to the measurement of the intensity of light that is emitted/received from/to a surface in the scene; an object, can be both a receptor and an emitter of light when its material have reflective or refractive properties. Several approaches addressing the global illumination problem were devised in the computer graphics field in an attempt to provide efficient techniques for achieving realism and physical correctness. One of the best rendering techniques for solving the global illumination problem are Monte Carlo (MC)[Hal70] techniques. In what follows, MC-based image synthesis techniques will be briefly exposed in the light of what has been announced earlier in chapter 2 and will constitute the backbone of the plenoptic camera simulator.

4.1.1 Interpretation of the rendering equation

The rendering equation cannot be solved analytically, however, a formal solution based on operators can be written for it. An operator is a mathematical entity that maps between functions. Let \( \mathcal{R} \) be a reflective operator that takes a radiance function and integrates its product with the BRDF in the scene:

\[
(\mathcal{R}L)(x, \omega) = \int \rho_{bd}(x, \omega, \omega') \cos \theta' d\omega'.
\]  
(4.1)

Hence,

\[
L = L^e + \mathcal{R}L \Rightarrow L = (1 - \mathcal{R})^{-1}L^e.
\]  
(4.2)

The operator \( \mathcal{R} \) transforms radiance function \( L(x, \omega) \) into radiance after one reflection. Using iteration, we can expand Equation 4.2 as follows:

\[
L_0 = L^e
\]
\[
L_1 = L^e + \mathcal{R}L_0
\]

\[...
\]
\[
L_n = L^e + \mathcal{R}L_{n-1} = \sum_{i=0}^{n} \mathcal{R}^n L^e.
\]
The infinite sum of the equation above is called a Neumann series and writes:

\[ L = \sum_{n=0}^{\infty} R^n L^e. \]  

(4.3)

Figure 4.2: Interpretation of the expansion into a Neumann series of the rendering equation: the case of two reflections.

Let us consider the Neumann series of the rendering equation at term 2 (i.e. \( I_g(x, \omega) \)), we have in this case:

\[ L = L^e_{\text{emission}} + R L^e_{\text{first reflection}} + R^2 L^e_{\text{second reflection}} \]  

(4.4)

In this case \( (n = 2) \), we constrained light to reflect over exactly two surfaces in directions \( \omega_2, \omega_1 \) and \( \omega_0 \) respectively (see figure 4.2). The contribution of each reflection is added to the irradiance of point \( (p, \omega_0, \omega_1, \omega_2) \).

4.1.2 Monte Carlo global illumination algorithms

Solving the rendering equation through MC (Monte Carlo) techniques employs the properties of random numbers to estimate the integrals of the Neumann Series (See appendix A). This amounts to the estimation of the paths rays follow before hitting the image plane. Applications of MC techniques for the solution of the rendering equation include: distributed ray tracing [CPC84], path
tracing [Kaj86], Bi-directional path tracing [LW93] and Metropolis light transport [Vea97]. These methods have the particularity of being view-dependent, are unbiased and converge to a physically correct result without any simplifying assumptions, as opposed to another set of techniques that use simplifications at the expense of accuracy such as radiosity [GTGB84], photon map [JC95] or ray bundles [Las00]. These methods also significantly reduce the time complexity [Las02], however, since it is vital for us to achieve accuracy, we will retain the first category of methods. It is worth noting that the algorithms mentioned here are gathering-walk algorithms, that is algorithms where rays are started from the eye, as opposed to shooting-walk algorithms where rays are started from light sources\(^1\).

All of the four unbiased methods mentioned above yield correct results. Furthermore, with the exception of distributed ray tracing, the other methods are very similar in their operation mode. Path tracing was devised as a direct implementation of the rendering equation, while bi-directional path tracing and metropolis light transport are improved variants that obviate some of the problems that arise in scenes with poor lighting or caustics\(^2\). Since there is no need for the support of complex scenes, Bi-directional path tracing and Metropolis light transport can be ruled out.

Distributed ray tracing operates by shooting randomly sampled rays at the pixel grid (called primary rays), and recursively spawning a random set of child rays at each intersected surface. In path tracing, instead of spawning several rays at each surface, a single random ray is generated according to the BRDF of the intersected surface, forming a path that goes from the pixel to the light source. Consequently, path tracing is considerably faster since it ignores those rays that do not contribute notably to the lighting in the scene without introducing any bias.

### 4.1.3 Path tracing with next event estimation using the modified Phong model

Path tracing maps to the multi-dimensional definition of the rendering equation described in section 4.1.1. In this method, a path represents a set of vertices linked with randomly formed edges; at

\(^1\) with the exception of bi-directional path tracing which belongs to both categories, because rays are started from both light sources and the camera

\(^2\) concentrated light refraction that produces extremely bright spots on a surface. An example is light refraction through a magnifying glass
each intersection, only one ray is spawned whose direction is sampled according to the BRDF of the surface using importance sampling. The latter ensures that the direction is generated where the transmission is large (See Appendix A).

Since its first introduction in [Kaj86], path tracing knew considerable improvements, mainly with regard to the sampling patterns that are used in its implementation. One of the most recent and efficient variants of path tracing is path tracing with next event estimation (PTNEE) [La96] which we will be using to solve the global illumination problem in target scenes. We will also be using the modified Phong reflectance model proposed in [LW97], which is a physically plausible model that supports diffuse and specular surfaces. In what follows, PTNEE will be detailed along with the modified Phong model from a mathematical standpoint and the corresponding algorithm will be given.

Mathematical expression of PTNEE

PTNEE considers light sources and objects as two distinct entities, therefore we can get rid of the light emitting $L^e$ term in equation 4.1. The rendering equation can be split into a direct lighting term which accounts for light source illumination, and an indirect lighting component that accounts for light that bounces from other visible objects:

$$L(x,\omega) = L_D(x,\omega) + L_I(x,\omega)$$

Where

$$L_D(x,\omega) = \int_A \rho_{bd}(x,\omega,\omega') L^e(x',\omega') G(x,x')V(x,x')dA'$$ \hspace{1cm} (4.5)

And

$$L_I(x,\omega) = \int_A \rho_{bd}(x,\omega,\omega') L(x',\omega') G(x,x')V(x,x')dA'$$

In both $L_D$ and $L_I$, we need Monte Carlo techniques to estimate the direction to follow at each object intersection according to the BRDF of the surface. PTNEE estimates the new direction to follow by selecting a random direction according to the Modified Phong Model through importance sampling.
The modified Phong reflectance model

The BRDF of a given surface using The Modified Phong reflectance model is described by:

\[
\rho_{bd}(x, \omega, \omega') = \frac{k_d}{\pi} + k_s \frac{n + \frac{2}{2\pi} \cos^n \alpha}{2\pi}
\]

(4.6)

where \( k_d \) denotes the diffuse coefficient, \( k_s \) is the amount of specularity, and \( n \) is the specular exponent. \( \alpha \) is the angle between the perfect specular direction (ideal mirror reflection). This reflectance model is physically plausible as per the definition given in the background chapter as long as \( k_d + k_s \leq 1 \). According to [LW97], the best way to choose a direction using this BRDF is to sample the diffuse and the specular components separately. To carry out importance sampling on the components, a proper PDF has to be chosen. For the diffuse part, the following cosine distribution is used:

\[
p_d(\omega) = \frac{\cos \theta}{\pi}
\]

where \( \theta \) is the angle that the incoming direction \( \omega \) forms with the surface normal. To sample a direction according to this PDF we use two random number \( \zeta_1 \) and \( \zeta_2 \) to define the polar and the azimuthal angles of the direction

\[
(\alpha, \phi) = (\arccos \sqrt{\zeta_1}, 2\pi \zeta_2)
\]

which translates into the following Cartesian representation:

\[
(x, y, z) = (\sqrt{1 - \zeta_1} \cos(2\pi \zeta_2), \sqrt{1 - \zeta_1} \sin(2\pi \zeta_2), \sqrt{\zeta_1})
\]

(4.7)

For the specular part, we use the following PDF:

\[
p_s(\omega) = \frac{n + 1}{2\pi} \cos^n \alpha
\]
and the corresponding direction is defined by:

\[ (\alpha, \phi) = (\arccos \frac{1}{\zeta_1^{n+1}}, 2\pi \zeta_2) \]

which translate into the following Cartesian representation:

\[ (x, y, z) = \left( \sqrt{1 - \zeta_1^{\frac{2}{n+1}}} \cos(2\pi \zeta_2), \sqrt{1 - \zeta_1^{\frac{2}{n+1}}} \sin(2\pi \zeta_2), \zeta_1^{\frac{1}{n+1}} \right) \] (4.8)

Shooting a specular or diffuse ray is done probabilistically; in the following section, we present how either type of ray is selected.

The corresponding algorithm

Since each light source diffuses light evenly in all directions, the BRDF used in the expression of \( L_D \) can be expressed in terms of the diffuse component only, and \( I^f(x, \omega) \) can be replaced with a constant intensity term \( I_s \) which does not depend on the direction. Hence,

\[ L_D(x, \omega) = \int_A V(x, x') \frac{k_d}{\pi} I_s G(x, x') dA' \]

where \( G \) and \( V \) denote the geometry term and the visibility term respectively, that are used in the rendering equation (defined by eq. 2.6).

Performing importance sampling on \( L_D \) with the proper PDF yields the following:

\[ L_D(x, \omega) = \frac{I_s k_d}{N\pi} \sum_{i=1}^{N} V(x, x'_i) \frac{G(x, x'_i)}{p_l(x'_i)} \] (4.9)

where \( N \) is the number of samples used in the estimate, and \( p_l \) is the probability of choosing a point on the light source. The PDF to be used depends on the shape of the light source. The issue of light source sampling is beyond the scope of this dissertation. We refer the reader to [SWZ96] for a set of Monte Carlo techniques for light source sampling.

According to section 4.1.1, the rendering equation can be estimated using a random walk that
extends from the pixel to the light source. At each object intersection, a new random direction is sampled within the hemisphere according to a probability density function (PDF) \( p \) to form a path. By using eq. 4.2 and eq. 4.9, the following estimate can be written for the rendering equation:

\[
L(x_0, \omega_0) = \sum_{i=1}^{n} L_{D}(x_i, \omega_i)(\prod_{j=0}^{i-1} \frac{\rho_{bd}(x, \omega_{i}, \omega_{i-1}) \cos \omega_{x_i}}{p(\omega_{x_i})p_{dir}}) \tag{4.10}
\]

We weight each path with \( p_{dir} \) which is equal to \( \varphi_d = \frac{k_d}{k_d + k_s} \) if a diffuse ray is cast, and to \( \varphi_s = \frac{k_s}{k_d + k_s} \) if we cast a specular ray. Deciding to shoot either type of ray is done probabilistically, by generating a canonical random number\(^3\) \( \zeta \) and casting a diffuse ray if \( 0 \leq \zeta < \varphi_d \) and a specular ray otherwise. This decision criterion holds when using the raw version of path tracer, that is, the one in which a path is to be stopped when it hits a light source. However, in practice this may require the generation of extremely long path, parts of which can be discarded without biasing the estimate. In fact, a MC technique named Russian roulette can be used to alleviate this shortcoming; by examining the recursive formulation of the rendering equation and the contractive nature of the reflection operation, it was concluded [Laf96] that the path may be stopped before hitting a light source without biasing the path. The expression of Russian roulette for a path tracer stipulates that at each intersection, we generate a random number \( \zeta \) according to which we either cast a diffuse ray, a specular ray or stop the path. A diffuse ray is cast when:

\[
0 \leq \zeta < \varphi_d
\]

A specular ray is cast when

\[
0 \leq \zeta < \varphi_d + \varphi_s
\]

Or the path is stopped if:

\[
\varphi_d + \varphi_s \leq \zeta < 1
\]

Note that indirect lighting contribution is taken into account only if the path hits a light source.

\(^3\)in the interval \([0, 1)\)
otherwise, it is only the direct lighting component that is used.

In Algorithm 2, we define the Trace() function used in algorithm 1 (chapter 2). The algorithm calculates the irradiance (or color) returned by a ray, by recursively tracing it through the scene and calculating its contribution at each intersection.

Alg. 2 Trace(ray)
\begin{algorithm}
\textbf{Require:} \quad \text{cumulativeColor} = 0 \text{ and } \text{transportTerm} = 1

1: \quad \text{if} \ \text{Intersect(ray, intersectionInfo)} \ \text{then}
2: \quad \quad \text{if} \ \text{IsLightSource(intersectionInfo)} \ \text{then}
3: \quad \quad \quad \text{colorContribution} \leftarrow \text{transportTerm} \ast \text{GetColor(intersectionInfo)}
4: \quad \quad \quad \text{//stop the path and return the resulting color}
5: \quad \quad \quad \text{return cumulativeColor} + \text{colorContribution}
6: \quad \quad \text{else}
7: \quad \quad \quad \text{//calculate direct lighting}
8: \quad \quad \quad \text{directColor} \leftarrow \text{CalculateDirect(intersectionInfo)} \ \text{//using eq. 4.9}
9: \quad \quad \quad \quad \text{$\zeta \leftarrow \text{random()}$}
10: \quad \quad \quad \text{if $\zeta < \varphi_d$ then}
11: \quad \quad \quad \quad \text{newRay} \leftarrow \text{SampleDiffuseDir(intersectionInfo)} \ \text{//using eq. 4.7}
12: \quad \quad \quad \quad \text{transportTerm} = \text{DiffuseTransportTerm(intersectionInfo)}
13: \quad \quad \quad \text{else if } \text{$\zeta < \varphi_d + \varphi_s$ then}
14: \quad \quad \quad \quad \text{newRay} \leftarrow \text{SampleSpecularDir(intersectionInfo)} \ \text{//using eq.4.8}
15: \quad \quad \quad \quad \text{transportTerm} = \text{SpecularTransportTerm(intersectionInfo)}
16: \quad \quad \quad \text{else}
17: \quad \quad \quad \quad \text{//termiate path}
18: \quad \quad \quad \quad \text{return cumulativeColor}
19: \quad \quad \quad \text{end if}
20: \quad \quad \quad \text{cumulativeColor} + \ast \text{transportTerm} \ast \text{directColor}
21: \quad \quad \quad \text{return Trace(newray)}
22: \quad \text{end if}
23: \quad \text{else}
24: \quad \quad \text{//No intersection}
25: \quad \text{return cumulativeColor}
26: \quad \text{end if}
\end{algorithm}

Functions \text{SpecularTransportTerm()} \text{ and } \text{DiffuseTransportTerm()}, \text{ which are referred to in the path tracing algorithm, calculate term } \frac{\omega_d \omega_i \omega_i \omega_j}{p(\omega_i)p(\omega_j)} \text{ in 4.10, whose value depends on whether we are casting a diffuse or a specular ray. Function Intersect()} \text{ returns the closest intersection point with an object in the scene.}
Note that we have only considered a single monochromatic wavelength so far. This would only produce gray-level images. To upgrade to a color representation, we calculate the color for each RGB component separately at the following steps of the path tracing algorithm: 3, 5, 8, 12, 15 and 20.

4.2 Simulating the Plenoptic Camera

4.2.1 The pixel equation

To be able to image a scene using a measuring device, we introduce the measurement equation (also known as the pixel equation). The measurement equation describes the response of a pixel belonging to the image plane in terms of the radiance that impinges on its surface. The measurement equation provides a framework for modelling any measuring device (a camera for instance). The intensity $m$ generated at a pixel is given by:

$$m = \int_\mathcal{P} \int_\mathcal{A} L(x, T(x, x')) G(x, x') dA(x) dA(x') \quad (4.11)$$

$L$ denotes the radiance emitted by the scene, $T$ encapsulates the geometric transformation that a sampled ray undergoes as it crosses the camera system, while $G$ is a geometry term equal to $\frac{\cos \theta \cos \theta'}{|x - x'|^2}$ that scales the radiance transported between $x$ and $x'$. Simulating the plenoptic camera amounts to the estimation of the different components that make up the measurement equation for which a discrete estimate should be determined. Similarly to the rendering equation we will use Monte Carlo sampling techniques to estimate $m$.

In equation 4.11, sampling intervenes at both the pixel level as well as the aperture. The latter in a plenoptic camera, corresponds to the aperture of the lenticule facing the sampled pixel. Previous research led by Shirley [Shi00] and Mitchell [Mit96] showed that a MC technique called stratified sampling yields the best results for pixel and aperture sampling. The estimate therefore writes:

$$m = \frac{1}{N_s} \sum_{i=0}^{N_s} L(x_i, T(x_i, x'_i)) \frac{\cos \theta \cos \theta'}{|x - x'|^2} \quad (4.12)$$
where \( N_s \) denotes the number of samples.

### 4.2.2 Approximation of the pixel equation

Equation 4.12 stipulates that in order to form an image within the plenoptic camera, we initiate rays from the pixel grid lying behind the lenticular sheet (sampling \( N_s \) points on the pixel area), trace them through the lenticule facing each one of them (estimating \( T \)), and determine their intersections with the main lens, as well as the exit point of the rays as they leave the lens system.

At this point, the role of the camera ends, and the rays that made it outside the camera framework will be principal rays that will initiate the global illumination calculation (estimating \( L \)) within the observed scene. The following steps detail the entire process:

**Step 1.** Sample points on the pixel grid using stratified sampling. This consists in subdividing the pixel area into strata that cover completely the pixel area, selecting a point in the middle of each subpixel and jittering it by adding random noise to it (see figure 4.3). In fact, although rectangular shapes are the most common form of strata, any other shape can be used [Mit96]. The procedure is yet simple and involves only the use of a pseudo-random number generator,

Figure 4.3: Stratified sampling.

but the sampling scheme it yields reduces considerably the variance of the estimate [Vea97].

**Step 2.** Sample points on the lenticule facing the pixel. The lenticular sheet lying in front of the sensor grid is made of small spherically shaped lenses that are arranged adjacent to one another. These lenticules are cut in a hexagonal or rectangular fashion to avoid any gaps resulting from their arrangement side by side. However, a rectangular arrangement is the most preferred form as advised by the inventors of the camera. Therefore, the same
sampling scheme (stratified) is used here as well. More specifically, we sample points from the rectangular area at the external edge of the lenticule.

**Step 3.** Link sample points on the pixel area with their lenticule counterparts to form initial rays. Linking a pixel sample \( S_p \) with a lenticule sample of the same order (i.e. \( S_l \)) would yield biased results. Instead, we use the *N-rooks sampling* technique [Shi00] where samples are linked randomly as illustrated in figure 4.4.

![Diagram of N-rooks sampling](image)

**Figure 4.4:** N-rooks sampling.

**Step 4.** Refract rays as they cross the outer surface of the lenticules through which they have been traced using Snell’s law (refer to the next step for a mathematical modelling of the refraction phenomenon).

**Step 5.** Successively intersect rays with each surface of the main lens (rays that hit the aperture stop or do not reach the last element because of vignetting, are discarded). To trace a ray through a lens system, we intersect it with the different surfaces that make up the lens. These are mainly spherical or planar, therefore, the intersection operations are straightforward. At each point of intersection, traced rays have to be refracted according to Snell’s law. The latter stipulates that a refracted ray lies in the plane defined by the incident ray and the normal to the point of incidence. The relation between the angle of incidence and the angle of refraction is given by:

\[
\frac{\sin \theta}{\sin \theta'} = \frac{r_i}{r_f}
\]

where \( r_i \) is the index of refraction of the incident ray and \( \eta \) is the index of refraction of the
transmitted ray. Therefore, given the direction \( d \) of the incident ray, the new direction \( d' \) of the transmitted ray can be deduced [Gla89]:

\[
d' = \frac{r_t(d - n(d \cdot n))}{r_t} - n \sqrt{1 - \frac{r_t^2(1 - (d \cdot n)^2)}{r_i^2}}
\]

where the root term has to be positive, otherwise refraction does not occur. The dot product of the incident vector and the normal to the point of incidence \((d \cdot n)\) should be negative, otherwise, the sign of the normal vector is to be reversed. An example of the multiple refractions rays undergo inside a lens system are depicted by figure 4.5.

![Figure 4.5: Rays traced through a 50mm Double Gauss lens.](image)

### 4.2.3 Thick lens approximation

For the majority of lenses\(^4\), we can obtain a thick lens approximation [KMH95] that can be used instead of the real lens system. This approximation is useful in two respects: (1) the intersection operations are reduced. (2) Focusing can be performed in a simple and efficient manner as will be outlined later in this section. To find a thick lens approximation for a real lens, we determine the corresponding principal planes that characterize the thick lens. In fact, focusing a camera lens at a certain distance requires that its subelements are internally moved inside the objective, while the objective itself remains stationary. This movement is not the same for all lenses and is generally undocumented in the majority of lens specifications. However, we can use a thick lens approximation

\(^4\)abberation-limited lenses
as a substitute, and focus it using lens law (equation 2.2).

In what follows, we detail the method used for finding the principal planes of a lens then outline the technique used for tracing rays through a thick lens.

![Diagram of a 50mm Double-Gauss lens depicting the method used in determining the second principal plane.](image)

Figure 4.6: Diagram of a 50mm Double-Gauss lens depicting the method used in determining the second principal plane.

**The Principal planes**

There are two principal planes that characterize a system of lens elements. A paraxial ray traced from the backplane of the lens system enters in a direction $d_0$ and undergoes several refractions before emerging from the foremost element with a direction $d_n$. Intersecting directions $d_0$ and $d_n$ yields the position of the first principal plane $P$. The second principal plane $P'$ is obtained in the same fashion by tracing a ray from the foremost element and intersecting it with the emerging direction at the rearmost element. Figure 4.6 depicts the two principal planes of a 50mm Double-Gauss lens which were obtained as follows: a paraxial ray is traced from the front of the lens (direction $d_0$). After few refractions, the ray emerges from the rearmost element with direction $d_n$. The intersection of the rays with directions $d_0$ and $d_n$ (in dashed lines) gives the position of $P'$. Note that $P'$ comes before $P$ for this lens.

To improve the accuracy of the estimated positions of $P$ and $P'$, we generally trace few paraxial rays and retain the mean value as the final position. A quick and efficient way to verify the validity
of these estimates consists in calculating the distance of $P$ and $P'$ from the respective focal points\(^5\). When the positions are correct, they are both equal to the focal length of the lens (in this case 50mm).

**Focusing**

To focus a thick lens at a distance $D$, we use lens law which is defined by:

$$v = \frac{fD}{D - f}$$  \hspace{1cm} (4.13)

where $v$ is the focal distance, which is calculated from principal plane $PP'$, $f$ is the focal length of the camera and $D$ is the distance to the observed point, that is the distance at which we are focusing. The latter is calculated from principal plane $PP$ (see figure 2.3). Note that we only move the principal planes, while keeping the image plane stationary.

### 4.2.4 Tracing rays through a thick lens

A ray to be traced through a thick lens is formed as follows: we select a sample point on the pixel and link it with a sample point on the last element of the approximated real lens. This sample ray is first intersected with a virtual plane located at distance $D$ (i.e. the focal plane) from principal plane $P$, and its intersection point is memorized. The ray is then intersected with principal plane $P'$. From that point it is translated to principal plane $P$, then extended to the initial intersection point with the focal plane (the one we memorized). Figure 4.7 illustrates the tracing technique. Note that the sampled ray has to be intersected with the aperture stop too. Depending on the position of the aperture stop with respect to the principal planes, This intersection operation is performed either before the ray reaches the $P$, between $P$ and $P'$ or after $P'$.

---

\(^5\)The focal point is the point of intersection of all paraxial rays with the optical axis after being transformed by the optics of the lens. The rear focal point is obtained by tracing paraxial rays from the foremost lens element and conversely the front focal point is obtained by shooting paraxial rays from the rearmost lens element.
4.2.5 Results

Figures 4.8 and 4.9 represent images generated by a 50 mm and a 100 mm lenses (resp.) mounted on a regular camera. In figure 4.10, a 300 × 300 plenoptic image was generated using the same lens as in 4.8 and a lenticular configuration of 100 × 100 lenticules each subtended by a macropixel of 3 × 3 pixels. The resulting extracted sub-images are shown in 4.11.

Figure 4.8: 320 × 200 scene rendered using a 50 mm Double-Gauss lens system and 625 samples per pixel. The lens is focused on infinity, but objects are acceptably sharp due to their distance from the camera. The farthest object is 10 m away from the camera.
Figure 4.9: Same scene as in figure 4.8 rendered using a 100 mm lens system (U.S. patent # 1,160,148) and 625 samples per pixel. Notice the zooming effect due to a higher focal length. The objects are slightly blurred because the focus is on infinity.

Figure 4.10: 300 × 300 plenoptic image of the scene in figures 4.8 and 4.9 rendered using a 50 mm Double-Gauss Lens and 1225 samples per pixel. The viewpoint resolution (nb. of subpixels) is 3 × 3 and the spatial resolution (nb. of lenticules) is 100 × 100.
Figure 4.11: Sub-images extracted from plenoptic image in 4.10. If the images along the horizontal or vertical dimension are put in an animation from left to right, the scene will appear to be moving along that dimension from right to left, since the scene is entirely located before the plane in focus (infinity).
Chapter 5

Depth Recovery Using a Plenoptic Camera

5.1 Single-Lens Stereo

Once the plenoptic image is acquired, it is broken down into a set of sub-images as described in the previous chapter. This set of sub-images is analyzed to produce a dense\(^1\) depth map from displacement estimates. Estimating image displacement (or disparity) is a central issue in range recovery techniques that operate on the parallax cue (SFM and stereo). In what follows, we present the depth formulation for the plenoptic camera. First let us formulate an expression of depth using a plenoptic camera; considering figure 5.1, let

\(^1\)In the presence of regions that lack high frequency features such as texture and bumps, the depth map may not be dense, which requires the resort to interpolation techniques. This is an inherent problem with all depth recovery techniques.
$f$ the focal length of the lens system
$F$ the distance between the image plane and the lens
$d$ the distance to a point in the scene from the lens
$h$ displacement of point’s image
$b$ displacement of aperture between (baseline)
$s$ distance to plane conjugate to plane in focus
$D$ distance to plane in focus
$e$ distance to conjugate plane beyond image plane

II and II' represent the plane at focus and its conjugate plane (resp.). The diagram also depicts how a sub-image is formed, where $b$ (the sub-image baseline) outlines the displacement between a sub-image formed by the center sub-aperture and the one imaged by the upper sub-aperture.

From the same illustration (figure 5.1), we can deduce (using similar triangles) that:

$$\frac{e}{s} = \frac{h}{b}$$

knowing that $e = s - F$ we get

$$\frac{1}{s} = \frac{1}{F}(1 - \frac{h}{b})$$

the lens equation

$$\frac{1}{f} = \frac{1}{s} - \frac{1}{d}$$

this yields,

$$\frac{1}{d} = \frac{1}{f} - \frac{1}{F}(1 - \frac{h}{b}) \tag{5.1}$$

or

$$\frac{1}{d} = \frac{h}{b} \left( \frac{1}{f} - \frac{1}{D} \right) + \frac{1}{D} \tag{5.2}$$

Single-lens stereo operates on a pair of images from two different viewpoints separated by baseline $b$. The images on which displacement analysis is performed are chosen adjacent to one another. However, this is not a requirement, and we may select any pair of images along one dimension to
CHAPTER 5. DEPTH RECOVERY USING A PLENOPTIC CAMERA

Figure 5.1: Single lens stereo with a plenoptic camera.

Perform displacement on.

Let $A$ be the diameter of the full lens aperture. In the case of two adjacent sub-images, $b$ takes the same value regardless of the orientation:

$$ b = \frac{A}{\text{nb. of subpixels}}. $$

In the general case, the baseline between two images $i$ and $j$ is given by:

$$ b_{ij} = (j - i) \frac{A}{\text{nb. of subpixels}} $$

For an array of $n \times n$ sub-images we have as many as $2n(n - 1)$ pairs of stereo images to estimate disparity from.

5.1.1 Expression of depth

The outcome of the plenoptic camera hence formed is subdivided into a number of sub images corresponding to the number of subpixels $N_6$ per macropixel, and whose spatial resolution along a dimension is equal to the number of lenticules (or macropixels) $N_l$ along the same dimension.

\footnote{Since the number of subpixels (hence the number of sub-views) is the same in both the vertical and horizontal dimensions}
Figure 5.2: Baseline between two different sub-images.

Considering figure 5.1, and assuming the following camera parameters are known: the focal length $f$, the distance at which camera is focused $D$ and the aperture number of the camera $A_n$, the depth $d$ of a point observed by two adjacent sub-images is expressed by:

$$\frac{1}{d} = \frac{h}{b} \left( \frac{1}{f} - \frac{1}{D} \right) + \frac{1}{D} \quad (5.3)$$

where $h$ is the displacement, and $b = \frac{f}{NA_n}$ is the baseline between two sub-views. The central problem in a plenoptic camera consists in finding the amount of displacement $h$. Given the fact that the baseline of sub-images is limited by the size of the aperture, inter-image baseline is very small.

### 5.2 Disparity Analysis

Lucas and Kanade proposed in [LK81] an iterative least squares technique for the estimation of $h$ that was later used by the camera inventors [AW92]. Let $I_1$ and $I_2$ be the respective intensity functions of the left and right images of a stereo pair. An error function can be written over a small patch of the image:

$$E(h) = \sum_x (I_1(x+h) - I_2(x))^2$$
Applying Taylor expansion on $I_1(x + h)$, and truncating it to retain only the linear terms (function and its first order derivative) yields:

$$I_1(x + h) = I_1(x) + hI_1'(x)$$

to minimize the error on $h$, we differentiate $E$ with respect to $h$ and set it to zero:

$$0 = \frac{\partial E}{\partial h} \approx \frac{\partial}{\partial h} \sum_x [I_1(x) + hI_1(x) - I_2(x)]^2$$

$$= \sum_x 2I_1'(x)(I_1(x) + hI_1'(x) - I_2(x))$$

hence

$$h \approx \frac{\sum_x I_1'(x)(I_2(x) - I_1(x))}{\sum_x I_1'(x)^2} \tag{5.4}$$

$I_1'$ is the spatial derivative of image 1 along the dimension on which the displacement analysis is performed. The spatial derivative can be approximated using a simple pixel difference. In our experiments, we did not find other derivative masks to yield a much different result. Each estimate is weighted with a confidence estimate:

$$w = \sum_x I_1'(x)^2$$

Due to a very small displacement, we need not resort to the iterative form of equation 5.4 which is outlined in [LK81]; retaining the linear terms of the Taylor expansion of $I_1(x + h)$ yields a good approximation when $h$ is sufficiently small, which is the case.

For $n \times n$ sub-images, we can have up to $2n(n - 1)$ stereo pairs in both dimensions altogether, to which eq. 5.4 can be applied. Hence, an estimate of $h$ for all pairs of adjacent images can be
CHAPTER 5. DEPTH RECOVERY USING A PLENOPTIC CAMERA

expressed as follows:

\[ h(x) = \frac{\sum_{i=1}^{n-1} \sum_{j=1}^{n} w_{i,i+1}^j(x) h_{i,i+1}^j(x)}{\sum_{i=1}^{n-1} \sum_{j=1}^{n} w_{i,i+1}^j(x) + \sum_{i=1}^{n} \sum_{j=1}^{n-1} w_{i}^{j+1}(x)} + \frac{\sum_{i=1}^{n-1} \sum_{j=1}^{n} w_{i,i+1}^j(x) h_{i,i+1}^j(x)}{\sum_{i=1}^{n-1} \sum_{j=1}^{n} w_{i,i+1}^j(x) + \sum_{i=1}^{n-1} \sum_{j=1}^{n} w_{i}^{j+1}(x)} \]  

(5.5)

where \( h_{i,i+1}^j \) is an estimate of two horizontally adjacent sub-images, and \( h_{i}^{j+1} \) is an estimate from two vertically adjacent sub-images (the same notation applies for the weights). Eq. 5.5 gives a weighted average of all adjacent stereo pairs of the sub-image set.

5.3 Experimental Results

The experiments conducted on the plenoptic camera simulator were geared towards determining the quality of depth estimates. As such, they were organized as follows. Three different scenes were used: in the first set of experiments, a close-range scene was imaged (scene CLOSERANGE1), where the target is 25 cm away from the camera objective and the depth offset between the different levels equals 5 cm. In all the experiments, the depth levels are in the form of three superimposed cubes on which were placed several small boxes (a "wedding cake"). These small boxes act as a 3D texture which provides the necessary high frequency information for depth extraction.\(^3\)

The second set of experiments was carried out on a medium-range scene where the target is located 2 m away from the camera (scene MEDRANGE). The depth levels are offset by a distance of 30 cm.

The third set of experiments was conducted on another close-range scene with similar specification as CLOSERANGE1 (scene CLOSERANGE2), but with a relatively small distance separating its cubes (3 cm).

The choice of the lenticular array parameters was made based on what is available in the market.\(^3\)as mentioned earlier, Stereo algorithms rely heavily on high frequency information such as texture, edges, corners, etc.
We used the specifications of a high-resolution lenticular sheet from Micro Lens Technology Inc\(^4\). With this lenticular sheet, we were able to obtain an acceptable quality of images even with low sampling rates. The parameters turned out to be also compatible with those used by the inventors of the camera in [AW92]. We also empirically tested some other lenticular configurations, however, as of the writing of this thesis, the simulations conducted with the Micro Lens lenticular yielded the best results. Table 5.1 outlines these parameters.

For each set of experiments, the plenoptic image of the scene is presented (in this order) along with its sub-images, a disparity map where displacement values are encoded in gray level (i.e. the more disparity we have, the brighter the color), as well as a depth plot. Note that the planes at focus appear in black on the disparity map since they remain stationary in all the set of plenoptic sub-images (zero disparity). The tests are organized as follows:

1. *depth recovery test on scene CLOSERANGE1* : figures 5.3, 5.4, 5.13 and 5.6.

2. *depth recovery test on scene MEDRANGE* : figures 5.7, 5.8, 5.9 and 5.10.


We also performed a study on how changing the imaging parameters affects the quality of depth, and compared the outcome with conventional stereopsis. The following outline the tests conducted and their corresponding illustrations:

1. *the effect of increasing viewpoint resolution* : figure 5.15.

2. *the effect of increasing sub-image resolution* : figure 5.16

3. *comparison with conventional stereo* : figures 5.17, 5.18 and 5.19.

For the calculation of disparity, we used a patch size of $11 \times 11$. Choosing a patch size is dependent upon the type of scene used. Increasing it imposes a smoothing constraint that blurs the sharp details in the images. In a highly textured scene for example, we can further increase its value without really affecting the accuracy of disparity estimates. In the opposite case, a patch size of 5 to 9 is sufficient.

\(^4\)www.microlens.com
Table 5.1: Lenticular sheet specifications retained for the experiments.

<table>
<thead>
<tr>
<th>Specification</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Period of the lenticular (diameter)</td>
<td>0.282222 mm</td>
</tr>
<tr>
<td>Curvature radius of lenticules</td>
<td>0.165 mm</td>
</tr>
<tr>
<td>Lenticule Gauge (thickness)</td>
<td>0.71 mm</td>
</tr>
<tr>
<td>Refraction index</td>
<td>1.6</td>
</tr>
</tbody>
</table>

Figure 5.3: Close-range scene (CLOSERANGE1).
Figure 5.4: Sub-images extracted from the CLOSERANGE1 scene.

Figure 5.5: Disparity map of the CLOSERANGE1 scene calculated with a patch size of 11 × 11. *Highest disparity value:* 4 pixels.
Figure 5.6: Depth plot corresponding to the CLOSERANGE1 scene.

Figure 5.7: Medium-range scene (MEDRANGE).
Figure 5.8: Sub-images extracted from the MEDRANGE scene.

Figure 5.9: Disparity map of the MEDRANGE scene calculated with a patch size of $11 \times 11$.  
Highest disparity value: 2 pixels.
CHAPTER 5. DEPTH RECOVERY USING A PLENOPTIC CAMERA

Figure 5.10: Depth plot corresponding to the MEDRANGE scene.

Figure 5.11: The CLOSERANGE2 scene.
Figure 5.12: Sub-images extracted from the CLOSERANGE2 scene.

Figure 5.13: Disparity map of the CLOSERANGE2 scene calculated with a patch size of $11 \times 11$. 
*Highest disparity value:* 3 pixels.
CHAPTER 5. DEPTH RECOVERY USING A PLENOPTIC CAMERA

Figure 5.14: Depth plot corresponding to the CLOSERANGE2 scene.

Figure 5.15: Depth plot obtained on a 5 × 5 set of sub-images representing the MEDRANGE scene.

Figure 5.16: Disparity map obtained from the analysis of a 3 × 3 set of sub-images from the CLOSERANGE2 scene, each with a resolution of 200 × 200. Highest disparity: 3 pixels.
Figure 5.17: Stereo pair of 100 × 100 pixels obtained with a baseline of 2 mm.

Figure 5.18: Stereo pair of 200 × 200 pixels obtained with a baseline of 2 mm.

Figure 5.19: Disparity map obtained from the analysis of a stereo pair with a medium and a large baseline (5 mm and 8 mm respectively). The images were 100 × 100 each. Point matching was carried out using ZNCC. Highest disparity value: 5 pixels.
5.4 Analysis and Discussion

We summarize the results obtained with the plenoptic camera in table 5.2. In all the simulations that were performed, the objective was to evaluate the depth extraction quality. This qualitative analysis unveils, as expected, that depth can indeed be extracted due to the parallax the camera provides in both the horizontal and vertical dimensions. However, the results are only qualitatively correct and depth resolution is poor (see table 5.2 for the number of recovered levels for each configuration).

For CLOSERANGE1 (image in figure 5.3), due to the fact that the out-of-focus planes are blurred, and although we have enough parallax to resolve those depth levels (4 pixels), only two depth levels were recovered and with many irregularities on each recovered surface (as can be seen by the depth plot in figure 5.6). In MEDRANGE (image in figure 5.7), three levels of depth were recovered. A maximum disparity value of 2 pixels was enough to recover the three depth levels without many outliers (figure 5.10). This is mainly due to the fact that the depth levels that are beyond the focal plane (second cube) have been imaged with a sharp focus, hence the displacement analysis was not biased by the blur which prevented the recovery of the other depth levels in CLOSERANGE1 (see figure 5.6). In CLOSERANGE2 (image in figure 5.11) the scene had almost the same specifications as CLOSERANGE1 (image in figure 5.3), yet the outcome is better, given the fact that there is no blurring in the out-of-focus planes. Nevertheless, fewer depth levels were recovered. This is explained by the fact that the depth difference (3 cm) is so small between the out-of-focus planes that they have been resolved with a single disparity value, whereas in MEDRANGE (see depth plot in figure 5.10, the difference in depth is large enough (30 cm) to gain an extra pixel of parallax.

Contrary to initial assumptions, increasing the number of viewpoints does not improve the accuracy of depth recovery. This can be noted when comparing the depth plots of MEDRANGE at a viewpoint resolution of 3 × 3 and 5 × 5 (figures 5.15 and 5.10 resp.). In the former, more outliers appeared on the depth plot and only two depth levels were recovered. This can be explained by the fact that the amount of parallax was divided among 5 views instead of 3, hence the one-pixel parallax level has disappeared. On the other hand, increasing sub-image resolution has yielded a decent result (disparity map in figure 5.16), as three depth levels were recovered instead of two for
the 100 × 100 versions. This was an expected result given that parallax is recorded on a higher number of pixels.

The comparison with stereo has yielded the following interesting facts about the plenoptic camera. By comparing the disparity maps obtained in figure 5.13 and that obtained using a stereo pair of 100 × 100 (figure 5.17), we can see that both techniques yielded similar results; only two levels were recovered. At a resolution of 200 × 200 (figure 5.18), the plenoptic camera could resolve another depth level (figure 5.16), in contrast with the stereo pair. Note that with relatively larger baselines, the disparity ambiguities can be clearly identified in the center of the disparity maps of the stereo pairs (figure 5.19). Those levels are supposed to have a brighter color.

A close look at the outcome of these tests has shown that the plenoptic camera has an enormous potential if the viewpoint aliasing problem (figure 5.20) is solved effectively as it undermines the displacement estimation. We have applied a low pass filter on the raw plenoptic image, however, aliasing was still present. This is an issue that was already pointed out in [AW92], which may be subject to future research.
### Table 5.2: Table recapitulating the outcome of the tests.

<table>
<thead>
<tr>
<th>Scene</th>
<th>Viewpoint Resolution</th>
<th>Sub-image Resolution</th>
<th>Minimum Depth</th>
<th>Maximum Depth</th>
<th>Focus Distance</th>
<th>Levels recovered</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLOSERANGE1</td>
<td>$3 \times 3$</td>
<td>100</td>
<td>250</td>
<td>400</td>
<td>250</td>
<td>2</td>
</tr>
<tr>
<td>CLOSERANGE1</td>
<td>$3 \times 3$</td>
<td>150</td>
<td>250</td>
<td>400</td>
<td>250</td>
<td>2</td>
</tr>
<tr>
<td>CLOSERANGE2</td>
<td>$3 \times 3$</td>
<td>100</td>
<td>310</td>
<td>400</td>
<td>310</td>
<td>2</td>
</tr>
<tr>
<td>CLOSERANGE2</td>
<td>$3 \times 3$</td>
<td>200</td>
<td>310</td>
<td>400</td>
<td>310</td>
<td>3</td>
</tr>
<tr>
<td>MEDRANGE</td>
<td>$3 \times 3$</td>
<td>100</td>
<td>2000</td>
<td>2900</td>
<td>2000</td>
<td>3</td>
</tr>
<tr>
<td>MEDRANGE</td>
<td>$5 \times 5$</td>
<td>100</td>
<td>2000</td>
<td>2900</td>
<td>2000</td>
<td>2</td>
</tr>
</tbody>
</table>

Figure 5.20: Viewpoint aliasing with the plenoptic camera. The image of the cubes is supposed to appear exactly the same as in `image22`. 
Chapter 6

Conclusions

This thesis has presented a study of a single-lens stereo system whose viability for the depth recovery problem has never been assessed prior to this work. The contributions of this work can be summarized as follows: (1) The depth capabilities of the plenoptic camera were evaluated in a variety of scenes, with different configurations, then compared with conventional stereo. (2) A physically accurate camera simulator was written to carry out tests on the plenoptic camera. This system is not confined to the plenoptic camera only, as it also supports regular cameras and approximative lens models (i.e. pinhole, thin lens and thick lens). It can be tailored to support other camera designs as well with few changes. (3) Indicators were given as to what key design parameters affect the performance. This is particularly useful in case a real-world model were to be designed for a given application.

This study has shown that the plenoptic camera yields an acceptable performance (in qualitative terms), albeit, as was outlined earlier, it suffers from problems that restrain its application scope to close-range applications, contrary to conventional stereo, which can be applied in almost every situation. The best asset of the camera is inarguably the fact that it provides horizontal and vertical parallax at the same time. This is not available with any other single-lens stereo system, and only achievable with a plurality of precisely positioned and calibrated regular cameras. The optics of the camera avoid that the correspondence problem set in, and provide an analytical approximation for
finding disparities. In other stereo systems, the resort to correlation techniques is usually computationally taxing, hence limiting their usability for real-time systems. On the other hand, the plenoptic camera does not bring any solution to the main limitation of parallax-based techniques, that is, the lack of high frequency information in target images. It also suffers from viewpoint aliasing which biases the disparity estimation. In close range applications, the default wide aperture of the plenoptic camera leads to a shallow depth of field. This blurs out details in the image which contain rapid changes (textures and edges for e.g.). The solution for this would require that all depth levels be in sharp focus, hence, separated by small depth values. In this case, a high resolution sensor should be used to resolve these depth levels correctly. To sum up, the plenoptic camera has a very unique design that clearly turns out to be insufficient for applications that demand a good depth resolution. This is mainly due to its viewpoint aliasing problem which cannot be obviated with image processing techniques. This is one of the most important design elements on which the performance of this camera relies.

6.1 Potential Applications

There are two key factors that determine the suitability of the plenoptic camera for a given application: depth resolution and the size of the aperture. Owing to its low depth resolution, the plenoptic camera can be used in applications that are not accuracy-critical such as shape recognition. The camera's baseline which is limited to the size of the aperture, plays against the accuracy of the matches recovery, however, it also constitutes a tremendous advantage for very close-range applications, such as microscopy. Being able to image from several viewpoints in a small environment is a very delicate process that can require high precision equipment and a considerable positioning process. This problem can be potentially solved by a high resolution plenoptic camera.

6.2 Future Work

This work has laid the groundwork for other ideas that can be investigated in the near future. Namely: (1) calibration; this study was based on the assumption that the camera parameters are
known. In the opposite case, the plenoptic camera provides enough images for self-calibration. (2) The issue of microscopic applications can be further explored so that specific designs are proposed based on the concept of plenoptic imaging. (3) New original lenticular sheet geometries (paraboloidal, hyperboloidal, etc.) can also be tested. This will increase the field of view of the camera and potentially bring an additional visual cue that can be exploited for depth recovery purposes. (4) The least-squares gradient approximation used to find disparities delivers inaccurate results. Improving this approximation is a crucial problem that can be explored further as well if the camera capabilities were to be used effectively.
Appendix A

Monte Carlo Integration

A.1 Overview of Monte Carlo Techniques

Estimating the integral of a function $f$ can be achieved through quadrature:

$$\int_{x \in S} f(x)p(x)dx \approx \frac{1}{N} \sum_{i=1}^{N} f(x_i).$$  \hspace{1cm} (A.1)

The sequence of $N$ samples we choose to estimate our integral is usually determined in advance and has a uniform distribution. Numerical techniques based on quadrature do not work very well in high-dimensions [Vea97], in fact their efficiency decreases rapidly with dimension, a phenomenon known as dimensionality curse. Besides, they are impracticable when the integrand is discontinuous.

Monte Carlo techniques on the other hand, aim at estimating the value of integral $I = \int_{x \in S} f(x)dx$ using a set of random samples:

$$I = \int_{x \in S} f(x)dx = \lim_{N \to \infty} \sum_{i=1}^{N} \frac{1}{N} f(\xi_i),$$  \hspace{1cm} (A.2)

where the samples $\xi_i$ are distributed according to a PDF $p(x)$ that is non-zero on the integration domain. According to the law of large numbers, the probability that this approximation is equal
to the exact value of the integral converges to 1 when $N$ tends toward infinity. Therefore, for this formula to yield a good estimate of the integral, we should provide as many samples as possible, in fact, by doing so, we will reduce the variance of $f/p$. Variance is a quality indicator of the reliability of Monte Carlo and is subject to the following rule

$$\text{var} \left( \frac{1}{N} \sum_{i=1}^{N} \frac{f(\xi_i)}{p(\xi_i)} \right) = \frac{\text{var} \left( \frac{f(\xi_i)}{p(\xi_i)} \right)}{N}$$

which confirms the need for a large number of samples. However, because the variance is proportional to $1/N$, the standard deviation of the integral estimator is proportional to $1/\sqrt{N}$, hence, to reduce the error, which behaves similarly to standard deviation, by one half, we will need to quadruple the number of samples. This results in a high computational cost. To obviate this problem, Monte Carlo techniques should be used with proper probability density functions that reduce the variance without the need for a very large set of samples. These variance reduction techniques are extremely useful in calculating integrals.

### A.2 Variance Reduction Techniques

There are several variance reduction strategies within Monte Carlo techniques that have their use in the RE:

#### A.2.1 Importance sampling

the idea behind this sampling strategy is to come up with a PDF $p$ that exhibits the same behavior as the integrand itself $f$, that is, ideally we would like to have $f = kp$, where:

$$k = \frac{1}{\int_{y \in S} f(y)dy}.$$  

This will result in a zero variance estimation. However, this is not possible, since we would need to know constant $k$, and consequently the value of the integral, in which case we would not be seeking
an approximation! However, by choosing a $p$ with a shape close to that of $f$, this approach will yield a good variance reduction.

A.2.2 Stratified sampling

Stratified sampling consists in dividing the integration domain into $n$ smaller intervals $I_i$ and picking a random representative value from within each $I_i$. This results in a good estimate since a good distribution of the samples in the integration interval ensures that a low variance is obtained.
Bibliography


BIBLIOGRAPHY


(JC95) Henrik Wann Jensen and Niels Jørgen Christensen. Photon maps in bidirectional monte
1995.

(JG94) B. Jahne and P. Geissler. Depth from focus with one image. In *Computer Vision and


(Kaj86) James T. Kajiya. The rendering equation. In *Computer Graphics (SIGGRAPH ’86

(KMH95) Craig Kolb, Don Mitchell, and Pat Hanrahan. A realistic camera model for computer


Based Rendering*. PhD thesis, Cornell University - Department of Computer Science,
Ithaca, NY, February 1996.

Department of Control Engineering and Information Technology - Technical University

(Las02) SzirmayKalos Laszlo. Private communication, 2002.

(LK81) B.D. Lucas and T. Kanade. An iterative image registration technique with an applica-
tion to stereo vision. In *International Joint Conference on Artificial Intelligence*, pages

on Computer Vision and Pattern Recognition*, pages 82–87, Fort Collins, Colorado,
June 1999.


Vita Auctoris

Mohamed Amtoun

Born: August 1979 in Casablanca, Morocco.

Education:

Master’s of Science in Computer Science - University of Windsor. September 2000 - April 2003

Bachelor’s of Science in Computer Science (Networks and Systems) - Ecole Des Hautes Etudes de Casablanca. September 1996 - July 2000