AN INVESTIGATION OF METHODS FOR DETERMINING DEPTH FROM FOCUS

By

John E. Ens

B. A. Sc. (Electrical Engineering) University of British Columbia

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

in

THE FACULTY OF GRADUATE STUDIES

ELECTRICAL ENGINEERING

We accept this thesis as conforming

to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

July 1990

© John E. Ens, 1990

51

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of Electrical Engineering

The University of British Columbia
Vancouver, Canada

Date July 24, 1990

Abstract

The concept of *depth from focus* involves calculating distances to points in an observed scene, by modelling the effect that the camera's focal parameters have on images acquired with a small depth of field. This technique does not require special scene illumination, and needs only a single camera. This thesis provides a background understanding of the concept and theory of depth from focus, surveys the literature on current methods of obtaining depth from focus, analyzes the key problems, and presents a novel solution to the problem, complete with experimental results.

Deconvolving and modelling the defocus operator, is the most difficult segment of calculating depth from focus. Isolating the defocus operator has conventionally been performed by taking local spatial regions, and inverse filtering in the spatial frequency domain. This thesis exposes some fundamental problems with this method: inaccuracies in finding the frequency domain representation and the presence of border effects.

To solve the general depth from focus problem, a novel application of an iterative matrix based method is presented. This method uses two images of the same scene, obtained under different conditions of defocus. The defocus operator may be assumed using a parametric model, or experimentally measured. Trade-offs in implementation are resolved through regularization. The method is theoretically justified, and shown to eliminate the problems mentioned above.

A constrained inverse filtering method and the author's iterative matrix based method are experimentally implemented on four scenes. The experiments show the iterative matrix solution consistently yielding more accurate results.

Table of Contents

Abstract ii

List of Tables ix

List of Figures x

Acknowledgement xiii

Nomenclature xiv

1 INTRODUCTION AND OVERVIEW 1

1.1 CONTEXT 1

1.2 MOTIVATION FOR RESEARCH 2

1.3 CONCEPT OF DEPTH FROM FOCUS 3

1.4 STATEMENT OF PROBLEM 3

1.5 OBJECTIVES OF THESIS 3

2 RATIONALE 5

2.1 INTRODUCTION 5

2.2 OVERVIEW 5

2.3 ADVANTAGES OF DEPTH FROM FOCUS 5

2.4 CONNECTION TO BIOLOGICAL VISION 8

2.5 DISADVANTAGES OF DEPTH FROM FOCUS 8

2.6 PROBLEMS COMMON WITH OTHER PASSIVE RANGING SYSTEMS 10

3 THEORY OF DEFOCUS 12

3.1 INTRODUCTION 12

3.2 OVERVIEW 12

3.3 MODELLING DEFOCUS AS CONVOLUTION BY AN OPERATOR . 13

3.4 GEOMETRIC OPTICS 14

3.4.1 Point Spread Function 14

3.4.2 Optical Transfer Function 17

3.5 DIFFRACTION OPTICS 18

3.5.1 Optical Transfer Function 18

3.6 COMPARISON OF GEOMETRIC OPTICS AND DIFFRACTION OPTICS 20

3.6.1 Optical Transfer Function 20

3.6.2 Point Spread Function 22

3.7 WORKING AREA AND RESOLUTION 24

3.7.1 Working Area Of Depth From Focus 24

3.7.2 Scene And Camera Resolution 27

3.8 CONCLUSIONS 28

4 LITERATURE REVIEW 29

4.1 INTRODUCTION 29

4.2 OVERVIEW 29

4.3 DEPTH FROM AUTOMATIC FOCUS 30

4.3.1 Biological Accommodation 31

4.3.2 Image Processing For Automatic Focus 31

4.3.3 Computing Depth From Lens Position 35

4.3.4 A Pyramid Approach 36

4.4 DEPTH FROM FOCAL GRADIENT 36
4.4.1 Concept Of Focal Gradient 36
4.4.2 Introduction Of Extra Cues 37
4.4.3 Assumptions Made About The Scene 39
4.4.4 More Than One Image Taken 40
4.5 PATENTED APPROACHES 45
4.5.1 Camera Autofocus 46
4.5.2 Other Depth From Focus Patents 47
4.6 COMBINING DEPTH FROM FOCUS WITH OTHER METHODS . . 47
4.7 CONCLUSIONS 48

5 ANALYSIS OF DEPTH FROM FOCUS PROBLEM 50
5.1 INTRODUCTION 50
5.2 OVERVIEW 50
5.3 PROBLEM DEFINITION 50
5.4 DESIRED CHARACTERISTICS OF SOLUTIONS 51
5.4.1 General Purpose Solutions 51
5.4.2 Accuracy 53
5.4.3 Minimal Data Required 53
5.4.4 Speed 54
5.5 KEY PROBLEMS AND TRADE-OFFS 54
5.5.1 Isolating And Modelling The Defocus Operator 54
5.5.2 Problem Ill-posed 54
5.5.3 Size Of Local Region 56
5.6 APPROACHES TO SOLUTIONS 57
5.6.1 Inverse Filtering 58

5.6.2 Matrix Based Solutions 58

5.7 CONCLUSIONS 59

6 THEORETICAL BASIS FOR NEW METHOD 60

6.1 INTRODUCTION 60

6.2 OVERVIEW 60

6.3 COMPUTATIONAL GOAL IS TO FIND $h_3(x, y)$ 61

6.4 NO NOISE IN IMAGES 63

6.4.1 A Formula For $h_3(x, y)$ 63

6.4.2 Uniqueness Of $h_3(x, y)$ For Geometric Optics 64

6.5 ONE BLURRED IMAGE WITHOUT NOISE 66

6.5.1 Matrix Solution For $h_3(x, y)$ 67

6.5.2 Inverse Filtering Solution For $h_3(x, y)$ 68

6.5.3 Problems With Inverse Filtering 69

6.6 TWO BLURRED IMAGES WITHOUT NOISE 76

6.7 TWO BLURRED IMAGES WITH NOISE 78

6.7.1 Closed Form Matrix Solution 79

6.7.2 Iterative Matrix Solution 80

6.7.3 Constrained Inverse Filtering 80

6.8 CONCLUSIONS 85

7 IMPLEMENTATION OF NEW METHOD 87

7.1 INTRODUCTION 87

7.2 OVERVIEW 87

7.3 CALCULATING A TABLE OF $h_3(x, y)$ PATTERNS 89

7.3.1 Justification For Calculations In One Dimension 90

7.3.2 An Example Of Calculating One $h_3(x, y)$ Pattern 94

7.3.3 Table Of $h_3(x, y)$ Patterns For Geometric Optics 100
7.4 ACQUIRING THE IMAGES 100
7.5 CORRECTING THE IMAGES 101
7.5.1 Radiometric Correction 101
7.5.2 Geometric Correction 108
7.6 FINDING EDGES 108
7.7 FINDING DISTANCE TO SCENE 109
7.8 CALCULATING AN OPTIMIZED TABLE OF $h_3(x, y)$ PATTERNS . . 111
7.9 CONCLUSIONS 116

8 EXPERIMENTAL RESULTS 118
8.1 INTRODUCTION 118
8.2 OVERVIEW 118
8.3 SCENE WITH TWO-DIMENSIONAL FEATURES 119
8.3.1 Constrained Inverse Filtering 119
8.3.2 Iterative Matrix Solution 123
8.4 SCENE WITH VARIOUS EDGES 126
8.4.1 Constrained Inverse Filtering 128
8.4.2 Iterative Matrix Solution 131
8.5 SCENE WITH RAMPED EDGES 131
8.5.1 Constrained Inverse Filtering 134
8.5.2 Iterative Matrix Solution 134
8.6 PHASE INVERSION 136
8.6.1 Constrained Inverse Filtering 137
8.6.2 Iterative Matrix Solution 139
8.7 CONCLUSIONS 140

9 SUMMARY, CONCLUSIONS AND FURTHER WORK **142**

9.1 OVERVIEW 142

9.2 SUMMARY OF DEPTH FROM FOCUS 142

9.3 CONCLUSIONS 144

9.4 OPPORTUNITIES FOR FURTHER WORK 148

9.4.1 Geometric Correction 148

9.4.2 Speed Up Implementation 148

9.4.3 Further Experimental Tests 149

9.4.4 Space-variant Solution 149

Appendices **151**

A GLOSSARY OF COMMON SYMBOLS **151**

B DECONVOLUTION OF LOCAL SIGNALS **154**

B.1 INTRODUCTION 154

B.2 OBJECTIVE 155

B.3 OVERVIEW 156

B.4 REPRESENTATION 157

B.4.1 Inverse Filtering Approach 157

B.4.2 Matrix Based Approach 160

B.5 SOLUTIONS 163

B.6 CONCLUSION 169

Bibliography **170**

List of Tables

6.1 Windowing errors incurred by inverse filtering 71

6.2 Windowing errors for matrix solution and inverse filtering 78

8.1 RMS errors using constrained inverse filtering on "alphabet" images . . 123

8.2 RMS errors using iterative matrix solution on "alphabet" images and assuming geometric optics 125

8.3 Values for R_{1p} and R_{2p} over the active range 125

8.4 RMS errors using iterative matrix solution on"alphabet" images and experimentally derived defocus operator 126

8.5 RMS errors using constrained inverse filtering on images of various edges 129

8.6 RMS errors using iterative matrix solution on images of various edges . 131

8.7 RMS errors using constrained inverse filtering on images of ramp edges . 134

8.8 RMS errors using iterative matrix solution on images of ramp edges . . 135

8.9 RMS errors using constrained inverse filtering on phase reversed images 138

8.10 RMS errors using iterative matrix solution on phase reversed images . . 140

List of Figures

3.1 Ray diagram to derive the geometric optics model of defocus. 15

3.2 The placement of α for the diffraction optics model of defocus 19

3.3 Optical transfer functions at a distance $D_o = 0.95\ m$ 21

3.4 Optical transfer functions at a distance $D_o = 0.85\ m$ 21

3.5 Point spread functions at a distance $D_o = 0.95\ m$ 23

3.6 Point spread functions at a distance $D_o = 0.85\ m$ 23

3.7 Example of working area of depth from focus 26

4.1 Thick-lens geometry 36

4.2 Basic principle of BIRIS system. 38

6.1 Block diagram of the depth from focus problem 62

6.2 $H_3(\rho_p)$ for distances $D_o = 0.90$ and 0.95 *meters* 65

6.3 Differences in scene area when the same image area is used 73

7.1 Flow chart of new, iterative matrix based method 88

7.2 Geometry of transforming $h_{3y}(y)$ to $h_3(r)$. 93

7.3 Examples of functions $h_{1y}(y)$ and $h_{2y}(y)$ 95

7.4 Example of $\mathbf{h_3}$ solved by matrix inversion. 96

7.5 Example of $\mathbf{h_3}$ solved by regularization with smoothing. 97

7.6 Example of $\mathbf{h_3}$ solved by regularization with smoothing and suppression of ends 99

7.7 Comparison of $\hat{\mathbf{h}}_2$ and $\mathbf{h_2}$. 99

7.8 Histogram of noise with fitted Gaussian 102

7.9 Radiometric calibration to Kodak Grey Scale 104

7.10 Radiometric calibration to a radiometer 105

7.11 Example of space-variant radiometric errors 106

7.12 Example of defocus operator with aberrations 112

7.13 Experimental examples of functions $h_{1y}(y)$ and $h_{2y}(y)$ 114

7.14 Models of the experimental functions $h_{1y}(y)$ and $h_{2y}(y)$ 115

7.15 Example of experimental $\mathbf{h}_3$ solved by regularization with smoothing and suppression of ends. 116

8.1 Slanted image of alphabet at $f_1 = 2.0$. 120

8.2 Slanted image of alphabet at $f_2 = 1.3$. 120

8.3 Depth values using constrained inverse filtering on "alphabet" images . 121

8.4 Depth values using iterative matrix solution on "alphabet" images and assuming geometric optics 124

8.5 Depth values using iterative matrix solution on "alphabet" images and experimentally derived defocus operator 127

8.6 Top view of scene geometry to obtain images with various edges. 129

8.7 Image with various edges at $f_1 = 2.0$. 130

8.8 Image with various edges at $f_2 = 1.3$. 130

8.9 Actual scene of ramped edges. 132

8.10 Image with various ramped edges at $f_1 = 2.0$. 133

8.11 Image with various ramped edges at $f_2 = 1.3$. 133

8.12 RMS errors using constrained inverse filtering on images of ramp edges . 135

8.13 RMS errors using iterative matrix solution on images of ramp edges . . 136

8.14 Images of vertical bars showing phase inversion 137

8.15 Depth using constrained inverse filtering on phase reversed images . . . 138

8.16 Depth using iterative matrix solution on phase reversed images 139

B.1 Deconvolving local segments of larger images 162

Acknowledgement

I would like to thank everyone whose encouragement and support made this thesis possible. In particular, I would like to express my gratitude to

- Dr. Peter Lawrence, my supervisor, for guiding me through the perilous path of producing a thesis;
- Dr. John MacDonald, who encouraged me to pursue a graduate degree, and instilled the confidence that someone from a small town can, on occasion, succeed;
- Dr. James McEwen, for graciously loaning me the CCD camera and frame grabber used for the experimental work;
- Peter George and Dr. Bruce Sharpe, for allowing me to use the computing facilities at MacDonald Dettwiler;
- The Natural Sciences and Engineering Research Council of Canada and the Science Council of British Columbia for their financial support;
- The staff in the Interlibrary Loans division of the U.B.C. Main Library, for their ability to ferret out even the most obscure references;
- My wife, Sherry, for her constant words of encouragement which kept my world in balance;
- My parents, who had little opportunity for formal schooling, but highly revered the value of an education.

Nomenclature

To avoid misinterpretation, the following notes clarify the nomenclature used in this thesis. A glossary of common symbols is also found in Appendix A.

- The words *depth*, *range* and *distance* are used interchangeably to denote distance from the camera to a point of interest in the scene.
- The word *scene* is used to describe the part of the world observed, or a two-dimensional view of it, uncorrupted by noise and perfectly in focus throughout.
- The word *image* describes a picture captured with a sensory apparatus such as a camera, which may be corrupted with noise or blurred in some areas. A complete image will be referred to as i_1 or i_2, and a local region of the image will be denoted as $i_1(x, y)$ or $i_2(x, y)$, where the Cartesian coordinates x and y locate the pixels.
- The term *frequency domain* always implies the *spatial frequency domain*.
- A function for convolution in the spatial domain is called either a *point spread function* or an *operator*.
- A function for multiplication in the spatial frequency domain is called a *frequency transfer function* or an *optical transfer function*.
- Where applicable, lowercase letters denote functions in the spatial domain and uppercase letters denote functions in the spatial frequency domain.
- Vectors are denoted as bold letters (ie. $\mathbf{h}$ or $\mathbf{H}$).

- Matrices are denoted as bold letters in square braces (ie. $[\mathbf{H}]$ or $[\mathbf{h}_{\mathbf{C}}]$). The subscript, if present, gives some clues about the structure of the matrix
  - **C**: Circulant;
  - **D**: Diagonal;
  - **T**: Toeplitz;
  - **B**: Block form; and
  - **S**: Stacked vector.
- Equations are denoted in round braces and references are denoted in square braces. When several references are given, they are placed in chronological order.
- All spatial frequency domain plots have been shifted so that $f_x = f_y = 0$ is in the middle of the abscissa.

Chapter 1

INTRODUCTION AND OVERVIEW

1.1 CONTEXT

The goal of a computer vision system is to transform two-dimensional data into a description of the three-dimensional world [150]. The perception of depth information is often an intermediary step in this transformation, especially when the observed scene contains unconstrained or unknown objects.[1]

A variety of techniques for depth perception currently exist and have been surveyed by several researchers [79][73][77][135][128][113][9][11][10]. The techniques include

- Stereo disparity;
- Motion parallax;
- Depth from image brightness or texture gradient;
- Active ranging with sonar, microwave or light;
- Introducing constraints by structured lighting of the scene;
- Size and linear perspective; and
- Depth from focus.

[1]Several researchers have shown [102][6] that depth information is not a *necessary* requirement to recognize and orient objects when some explicit knowledge about those objects is known *a priori*.

A system for determining depth should ideally make use of a variety of techniques for obtaining cues from the observed scene. The human vision system, for example, relies on most[2] of the techniques mentioned above, as well as contextual and other cues, which relate certain shapes, textures and objects to specific dimensions. Making use of all the information available, gives a system robustness to be able to discern cues that are fuzzy, or ambiguous.

To allow for the development of such a robust, general purpose system for determining depth, more specific research needs to be done in the individual techniques of depth perception. Recently there has been renewed interest in the technique of obtaining *depth from focus* [120][117][140]. This thesis presents a detailed investigation into this technique. The methods described in this thesis could be used as a segment of a larger, general, depth perception system, or for specific applications, the method of depth from focus may suffice. Other researchers are attempting to answer the problem of turning range data into a useful description of the three-dimensional world [126][12].

1.2 MOTIVATION FOR RESEARCH

The motivation for this work is two-fold. It is expected that methods for determining depth from focus can be used:

1. For specific applications in areas such as robotic sensing or industrial inspection, which are appropriate to the strengths and weaknesses of the technique; or

2. In conjunction with other complementary techniques for depth perception, as part of a larger, general purpose computer vision system.

[2] Active ranging is of course excluded.

1.3 CONCEPT OF DEPTH FROM FOCUS

A camera's focal parameters, such as the aperture size, the position of the image plane, and the focal length, determine the distance to points in the scene that will be in perfect focus, and the *depth of field*.[3] If a camera has a very limited depth of field, then only points in the observed scene that are the same distance from the camera will be in perfect focus. Other points in the scene, at different distances from the camera, will be out of focus. Therefore, the distance to an object in a scene can be deduced by knowledge of current camera focal parameters *and* the degree that the object is out of focus.

In summary, the concept of depth from focus involves calculating distances to points in an observed scene, by modelling the effect that the camera's focal parameters have on images acquired with a small depth of field. This technique is passive and requires only a single camera.

1.4 STATEMENT OF PROBLEM

The problem addressed by this thesis, is given a blocks scene with featured objects an unknown distance from the camera, but within a known range, derive the distance to features in the scene using the concept of depth from focus. This will be attempted in a setting that is as generalized as possible.

1.5 OBJECTIVES OF THESIS

The objectives of this thesis are to:

[3]The *depth of field* is the range of distances over which objects are fairly well focused [66, p. 25].

- Provide a background understanding to the concept and theory of depth from focus;
- Survey the literature for current methods of obtaining depth from focus;
- Analyze the merits and problems of current methods;
- Investigate new general methods for obtaining depth from focus; and
- Experimentally implement the most general methods.

Chapter 2

RATIONALE

2.1 INTRODUCTION

This chapter provides the rationale for the method of depth from focus. Its purpose is to give the reader a summary of the strengths and weaknesses of this method, with some comparisons made to other passive ranging techniques. Once this background has been established, Chapter 3 proceeds with the mathematical theory of defocus.

2.2 OVERVIEW

This chapter opens with Section 2.3, which outlines the general advantages of depth from focus over most other passive ranging techniques. In addition to these advantages, Section 2.4 explores the connections depth from focus has with biological vision systems.

To present a balanced view, the next two sections look at some of the problems with depth from focus. Section 2.5 describes the disadvantages of depth from focus, that are somewhat unique to this technique. Then Section 2.6 looks at disadvantages that depth from focus shares with other passive ranging systems.

2.3 ADVANTAGES OF DEPTH FROM FOCUS

This section will look at some of the advantages of depth from focus, that make the technique attractive.

No correspondence problem: Any passive optical system, imaging an unknown object, requires at least two rays of light from the object, to determine the distance to the object. Simply put, a single light ray carries no information as to how far it has travelled, after reflecting from an object.[1]

In binocular stereo vision, the two rays of light from the same point on an object, are obtained from two imaging systems which have a different view or projection. Combining the conjugate points on both images requires a clever, often hierarchical, *correspondence* algorithm. This process is computationally intensive and often susceptible to errors [55].

In a depth from focus system, the two rays of light are intrinsically matched by the symmetry of a normal lens system. The optical system in most cameras reproduces the entire observed scene into a small three-dimensional image space behind the lens. For an ideal single lens, all light rays from a point object at a distance D_o from the lens *must* meet at the image point, a distance D_i on the other side of the lens, according to the Gaussian lens law [114]

$$\frac{1}{D_o} + \frac{1}{D_i} = \frac{1}{F}\,, \tag{2.1}$$

where F is the focal length. Therefore, in a depth from focus system, the *correspondence problem* is eliminated, and replaced instead by a series of image processing tasks to locate D_i within the image space.

Suitable for hardware implementation: As will be explained in Section 3.3, defocus can be modelled as a convolution of a scene with a defocus operator. Therefore, in solving the depth from focus problem, the key sub-problem is isolating and modelling this defocus operator. As will be seen in Chapter 6, this can be

[1] With coherent light, some information could be found in the phase difference between very close neighbouring rays, however normal lighting is usually incoherent.

described as a series of deterministic image processing tasks (ie. Fast Fourier transforms, convolutions, or matrix operations). The predictable computational path simplifies the task of hardware implementation.

Suitable for parallel processing: In solving the depth from focus problem, *all* of the information regarding the state of focus can be found within a local region, *independent* of neighbouring regions. Although focus information from neighbouring regions may be useful as a continuity constraint, it is not essential. Since depth from focus is primarily a local task, it is quite suitable for parallel processing, where any number of processing elements can determine the depth of local regions independent of each other.

Requires only one camera: Another important advantage of the depth from focus technique, is that only one camera is required. Although two or more images of the same scene may be required with different focal parameters, these can be obtained from the same camera, since a different viewing angle or projection is not required. A one-camera system decreases the total cost and lends itself to a more compact implementation.

No occlusion problem: In calculating depth from focus, two or more images may have to be obtained with different focal parameters, however these images are from the same perspective and viewing angle. Therefore *all* objects present in one image are also present in the other image (although they may be blurred by differing amounts). This avoids the *occlusion problem*, which can occur in other ranging techniques such as binocular stereo or depth from motion [142].

2.4 CONNECTION TO BIOLOGICAL VISION

Another reason for pursuing depth from focus, is a connection to biological vision, although it may not be readily apparent. It is recognized that accommodation[2] is an inaccurate cue for depth perception in humans [125, p. 110][120, p. 527]. That is, the sole act of focusing from one object to another, yields little appreciation of their absolute distance from the viewer.

However, Pentland has reported that for a fixed accommodative state, the amount of defocus in the rest of the scene is a direct cue to relative depth [119][120]. He proposes a simple experiment to affirm this hypothesis.

> "First make a pinhole camera by poking a small, clean hole through a piece of stiff paper or metal. Imposition of a pinhole in the line of sight causes the depth of field to be very large, thus effectively removing this depth cue from the image. Close one eye and view the world through the pinhole, holding it as close as possible to the surface of your eye, and note your impression of depth (for those of you with glasses, things will look sharper if you are doing it correctly). Now quickly remove the pinhole and view the world normally (still using only one eye). The change in the sense of depth is remarkable; many observers report that the change is nearly comparable to the difference between monocular and binocular viewing, or the change which occurs when a stationary object begins to move." [120, p. 527]

2.5 DISADVANTAGES OF DEPTH FROM FOCUS

To provide a balanced view of depth from focus, this section will describe some of the disadvantages of this technique.

[2] Accommodation means changing the focal length of the eye's lens.

Error increases with distance squared: One important problem with using the depth from focus technique, is that the error increases with distance squared [83, p. 235], whereas the error in binocular stereo is proportional only to the distance.[3] Therefore, by definition, the depth from focus technique is only suitable for *near-field* perception. As will be shown in Section 3.7, the placement and range of this *near-field* is selected by changing the camera's focal parameters. It has been shown by Pentland [120], that assuming human visual system parameters, even at a distance of 10 *meters*, the error in using focus information is only twice that of using stereopsis.

Problem ill-posed: As will be described in Section 5.5.2, depth from focus involves solving an *inverse optics* problem, and is often ill-posed. Therefore, a carefully constructed, robust technique will be required to adequately solve this problem.

Ambiguity about point of focus: As will be shown in Section 3.4.1, the theoretical defocus operator is symmetrical on either side of the point of best focus.[4] Therefore, even if the defocus operator is accurately identified, the distance to the point of interest in the scene can still be one of two values. To resolve this ambiguity, requires a third image be taken with different focal parameters (see Section 7.2), or more commonly, the objects in the scene will have to be restricted to a known range of distances. The latter approach will be taken in the experiments described in Chapter 8.

No scene motion permitted: If more than one image is obtained with the same camera, then it is critical that no motion (in the scene, or of the camera) take

[3] For simple two camera geometry, as given by Horn [66, pp. 300–301].

[4] In a camera with non-ideal optics, it is possible for the defocus operator to be unsymmetrical on either side of the point of best focus, due to aberrations (this was indeed true for the camera used by the author for the experiments in Chapters 7 and 8), however this difference is probably too slight to be reliably detected.

place between acquired images. This can be ameliorated by using a two camera system for depth from focus, such as presented by Pentland [117], where both images are acquired simultaneously. However, motion during the acquisition of the images may still affect the results, depending on the details of the system.

Another approach to the problem of scene motion is to combine depth from focus with *motion parallax*, since the two are complementary ranging techniques. Subbarao has proposed a system for monocular motion recovery and binocular recovery of depth and motion [142].

2.6 PROBLEMS COMMON WITH OTHER PASSIVE RANGING SYSTEMS

In addition to the disadvantages cited in Section 2.5, depth from focus also has problems which it shares with other passive ranging systems.

Illuminated scene features required: Since depth from focus is passive,[5] it is necessary for the scene to contain high-contrast features such as edges. For example, this technique would not function if the scene was a blank, featureless wall. A similar requirement also exists for other passive ranging techniques, such as binocular stereo or motion parallax.

In addition to a scene with high-contrast features, a sufficient external illumination of the scene is also required. The required intensity of illumination is, of course, dependent on the aperture size and sensitivity of the camera used to acquire the images.

[5] Although, as shown in Section 4.4.2, some researchers have simplified the depth from focus problem by introducing active elements into the scene, solving the problem does not *require* active elements and therefore will be called passive.

Repetitive patterns: Repetitive patterns present an ambiguity problem for binocular stereo, since the *correspondence problem* can be solved in several ways, implying different depths [107, p. 113]. Although the correspondence problem does not arise in depth from focus, repetitive patterns may induce a phase inversion in the defocused image [51, §6-4], which is also problematic. As will be shown in Section 8.6, this problem is partially alleviated by the author's method of calculating depth from focus, as long as only one phase inversion has occurred.

Other foibles: Additionally, there are other foibles which depth from focus shares with other passive ranging systems:

- Camera sensor saturation or black-out introduces non-linear intensities into the acquired images;
- Specularity and mirrored surfaces image the light source, giving inaccurate depth values of the surface; and
- Semi-transparent surfaces, such as dirty windows, offer intermingled cues of two or more different depths.

Chapter 3

THEORY OF DEFOCUS

3.1 INTRODUCTION

Chapter 2 has shown that depth from focus is a viable technique for depth perception. To understand the nature of defocus, this chapter outlines the theoretical models given in the literature. This forms a necessary background for Chapter 4, which is a comprehensive summary of methods for determining depth from focus, as currently available in the literature.

3.2 OVERVIEW

Section 3.3 models defocus as a convolution of a scene by a point spread function. The next two sections then outline two traditional methods for analyzing defocus:

1. *Geometric Optics*, which relies on ray-tracing and results in a first-order approximation.

2. *Diffraction Theory*, which uses the wave theory of light and its results are exact.

After these models have been presented, Section 3.6 compares the two models in both the spatial and spatial frequency domains. The last section of this chapter presents some simple equations for calculating the working area of depth from focus, and the minimum resolution required.

3.3 MODELLING DEFOCUS AS CONVOLUTION BY AN OPERATOR

The transformation of an optical system can be modelled as a convolution operation. If it is assumed that

- The optical system is linear and shift-invariant; and
- The light source is incoherent,

then Goodman [51, p. 109] gives the following equation

$$i(x,y) = K \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |h_c(x-\xi, y-\eta)|^2 s(\xi,\eta) d\xi d\eta \ , \tag{3.1}$$

where

- $i(x,y)$ is the measured image intensity distribution;
- $s(x,y)$ is the intensity distribution of the scene being viewed;
- $h_c(x,y)$ is the impulse response obtained with coherent illumination; and
- K is a real constant.

If the incoherent impulse response $h(x,y)$, is defined as

$$h(x,y) = K|h_c(x,y)|^2 \ , \tag{3.2}$$

then (3.1) can be rewritten as

$$\begin{aligned} i(x,y) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x-\xi, y-\eta) s(\xi,\eta) d\xi d\eta \ \text{ and} \\ i(x,y) &= h(x,y) \otimes s(x,y) \ , \end{aligned} \tag{3.3}$$

where $\otimes$ is the convolution operator.

Equation (3.3) relates the scene to the acquired image by a simple convolution relationship. In the spatial frequency domain, this can be expressed as

$$I(f_x, f_y) = H(f_x, f_y) \cdot S(f_x, f_y) \tag{3.4}$$

where

$$i(x,y) \iff I(f_x, f_y) \ , \tag{3.5}$$

$$h(x,y) \iff H(f_x, f_y) \ , \text{ and} \tag{3.6}$$

$$s(x,y) \iff S(f_x, f_y) \ , \tag{3.7}$$

are Fourier pairs.

3.4 GEOMETRIC OPTICS

This section outlines the *geometric optics* model of defocus by ray tracing in the spatial domain. The result is a first-order approximation.

3.4.1 Point Spread Function

The geometric optics model for defocusing is explained by Horn [65][66, p. 127].

From Figure 3.1, similar triangles yield a formula for R, the radius of the blur circle,

$$R = \frac{L\delta}{2D_{fi}} \ , \tag{3.8}$$

where

- L is the diameter of the lens or the aperture;
- D_{fi} is the distance from the lens to a sharply focused image of a particular object; and

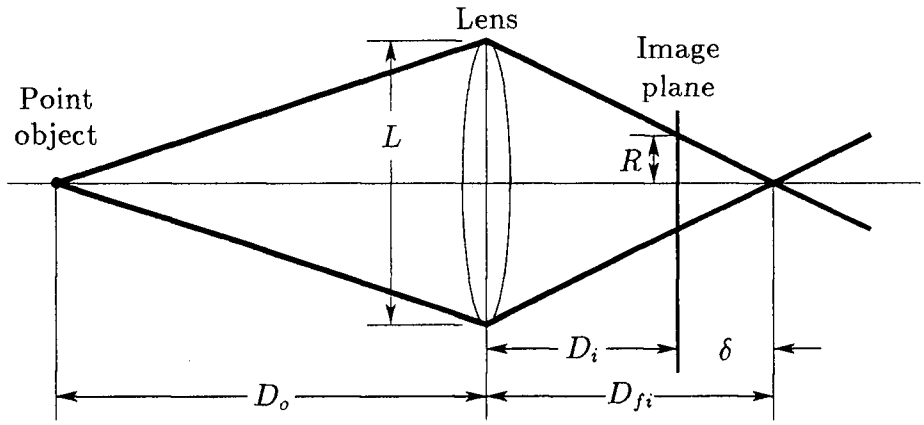


Figure 3.1: Ray diagram to derive the geometric optics model of defocus.

- δ is the displacement of the image plane from sharp focus.[1]

To a first order approximation, the brightness within the blur circle is uniform. The defocus operator $h(x, y)$, where x and y are coordinates in the image plane, is defined as

$$h(x,y) = \begin{cases} \frac{1}{\pi R^2} & \text{if } x^2 + y^2 \leq R^2 \\ 0 & \text{if } x^2 + y^2 > R^2 \end{cases} . \tag{3.9}$$

Blurring due to defocus can now be modelled as convolution with the pillbox given in (3.9) [66, p. 127].

It is easy to turn (3.8) into an expression which relates the radius of the blur circle with the distance to an object. Also obtained from Figure 3.1, is the relation

$$\delta = D_{fi} - D_i \ , \tag{3.10}$$

where D_i is the distance from the lens to the defocused image.

[1] To avoid ambiguity, it will be assumed that objects are *in front of* the point of best focus, since it can be seen that a blur circle of equal radius is also formed on the other side of the point of perfect focus.

It is useful to note that for an optical system, the diameter of the aperture L, is often given as [66, p. 208][60, p. 152]

$$L = \frac{F}{f} , \tag{3.11}$$

where F is the focal length, and f is the f-number. Substituting (3.10) and (3.11) into (3.8) yields

$$R = \frac{FD_{fi} - FD_i}{2fD_{fi}} . \tag{3.12}$$

Then, solving the lens law, where D_o is the distance from the lens to a point object,

$$\frac{1}{D_{fi}} + \frac{1}{D_o} = \frac{1}{F} , \tag{3.13}$$

for D_{fi} yields

$$D_{fi} = \frac{FD_o}{D_o - F} . \tag{3.14}$$

Substituting (3.14) into (3.12), eliminates D_{fi},

$$R = \frac{FD_i + FD_o - D_iD_o}{2fD_o} . \tag{3.15}$$

Or, solved for D_o, (3.15) can be rewritten as

$$D_o = \frac{FD_i}{2fR + D_i - F} . \tag{3.16}$$

Equations (3.15) and (3.16) are simple expressions which relate the radius of the blur circle R, to the object distance D_o. Equation (3.16) shows that defocus (designated by R), is a unique indicator of depth for a point source.

Since the experimental results will be obtained in distances mensurated in pixels, it is helpful to also express R in pixel units.[2] Therefore, if P is the digitization ratio (ie. pixels/mm) of the camera, and R_p is R in pixel units, then

$$R_p = R \cdot P . \tag{3.17}$$

[2]Currently, R takes on the same units as F, D_i, and D_o.

3.4.2 Optical Transfer Function

Since the spatial domain model for defocus is circularly symmetric, the optical transfer function of defocus can be obtained by finding the Hankel transform of (3.9) [18, p. 244][66, p. 127].

To find the Hankel transform, polar coordinates in both the spatial and spatial frequency domains must be defined:

$$x = r\cos\theta \quad \text{and} \quad y = r\sin\theta \ , \tag{3.18}$$

$$f_x = \rho\cos\phi \quad \text{and} \quad f_y = \rho\sin\phi \ , \tag{3.19}$$

where

- r is the radial component in the spatial domain;
- θ is the angular component in the spatial domain;
- f_x and f_y are the spatial frequency coordinates, expressed in cycles/unit, *not* radians;
- ρ is the radial component in the spatial frequency domain; and
- ϕ is the angular component in the spatial frequency domain.

Then, if a circularly symmetric $f(x,y) = f(r)$, Bracewell gives the formula for the Hankel transform of $f(r)$ as [18, p. 248]

$$F(\rho) = 2\pi \int_0^\infty r f(r) J_0(2\pi r\rho) dr \ , \tag{3.20}$$

where $J_0(x)$ is the zeroth-order Bessel function.

Using (3.20), the Hankel transform of (3.9) is

$$H_g(\rho) = 2\pi \int_0^R \frac{r}{\pi R^2} J_0(2\pi r\rho) dr \quad \text{and} \tag{3.21}$$

$$H_g(\rho) = 2\frac{J_1(2\pi R\rho)}{(2\pi R\rho)} \ , \tag{3.22}$$

using the identity

$$\frac{d}{dz} z J_1(z) = z J_0(z) \ . \tag{3.23}$$

Equation (3.22) is an expression for the optical transfer function of defocus, assuming geometric optics.

To maintain compatibility with the discrete Hankel transform, which will be discussed in Section 3.6.2, the radial frequency component ρ, can with the help of (3.17), be expressed in terms of pixels as ρ_p [123, §12.1],

$$\rho = \frac{\rho_p P}{N} \ , \tag{3.24}$$

where N is the number of points in the transform.

3.5 DIFFRACTION OPTICS

The *diffraction optics* model of defocus uses the wave theory of light and its results are exact.[3] The theory is described by Hopkins [64][63][61]. It is also covered in the standard reference on optics by Born and Wolf [16], and in limited form by other references [51][82][134]. The details will not be presented here, so the reader is referred to these references. Unlike the geometric optics model, the optical transfer function is usually derived first, and the point spread function is derived later.

3.5.1 Optical Transfer Function

For incoherent monochromatic light, and a circular aperture, Hopkins [61] gives the normalized[4] optical transfer function of defocus to be

$$H_d(s) = \frac{4}{\pi a} \int_0^{\sqrt{1-\frac{s^2}{4}}} sin \left[a \left(\sqrt{1-y^2} - \frac{|s|}{2} \right) \right] dy \ , \tag{3.25}$$

[3] That is, assuming the optics are ideal.

[4] The word "normalized" implies that the optical transfer function at $f_x = f_y = 0$ is one.

where s is a reduced spatial frequency related to the spatial frequency in radial coordinates ρ, by

$$s = \frac{\lambda \rho}{\sin \alpha} . \tag{3.26}$$

Here λ is the wavelength of the monochromatic light, and $\sin \alpha$ can be calculated from Figure 3.2 as

$$\sin \alpha = \frac{\frac{L}{2}}{\sqrt{D_{fi}^2 + \frac{L}{2}^2}} . \tag{3.27}$$

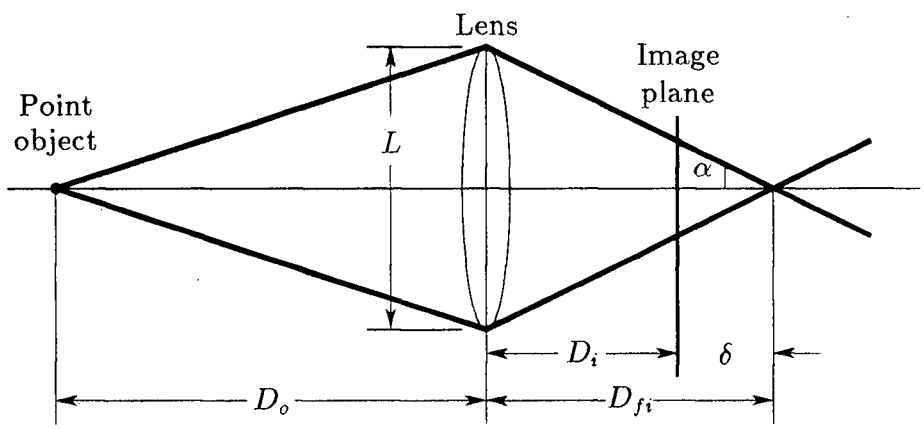


Figure 3.2: Ray diagram to illustrate the placement of α for the diffraction optics model of defocus.

Using (3.14) and (3.11), (3.27) can be transformed into

$$\sin \alpha = \frac{D_o - F}{\sqrt{(D_o 2f)^2 + (D_o - F)^2}} . \tag{3.28}$$

The other dimensionless parameter a, in (3.25), is given by Hopkins as

$$a = \frac{4\pi w |s|}{\lambda} , \tag{3.29}$$

where the path error of defocus w, is

$$w = \frac{\delta \sin^2 \alpha}{2} . \tag{3.30}$$

Using (3.10), (3.30) can be transformed into

$$w = \frac{D_{fi} - D_i \sin^2 \alpha}{2} . \tag{3.31}$$

To calculate the optical transfer function of defocus, (3.25) can be numerically integrated, or Hopkins also gives (3.25) as a series of Bessel functions,

$$\begin{aligned} H_d(s) &= \frac{4}{\pi a} \cos\left(\frac{as}{2}\right) \left\{ \beta J_1(a) + \sum_{n=1}^{\infty} (-1)^{n+1} \left(\frac{\sin 2n\beta}{2n}\right) \left[J_{2n-1}(a) - J_{2n+1}(a)\right] \right\} - \\ & \frac{4}{\pi a} \sin\left(\frac{as}{2}\right) \left\{ \sum_{n=0}^{\infty} (-1)^n \left(\frac{\sin(2n+1)\beta}{2n+1}\right) \left[J_{2n}(a) - J_{2(n+1)}(a)\right] \right\} , \end{aligned} \tag{3.32}$$

where

$$\beta = \cos^{-1}\left(\frac{s}{2}\right) . \tag{3.33}$$

Other numerical approximations are given by Hopkins [62], and Stokseth [134].

3.6 COMPARISON OF GEOMETRIC OPTICS AND DIFFRACTION OPTICS

3.6.1 Optical Transfer Function

To compare the optical transfer function of defocus, using the geometric optics model and the diffraction optics model, several physical parameters will be assumed. Let

- The focal length of the lens be $F = 50\ mm$; and
- The point of perfect focus be set at $D_o = 1.0\ m$.

The distance to the image plane can be found by (2.1), to be $D_i = 52.63\ mm$. It will also be assumed that

- The wave length, $\lambda = 500\ nm$;[5]

[5] The ends of the visible range, $\lambda = 400$ and $\lambda = 700\ nm$ were also tried, with no appreciable change in results.

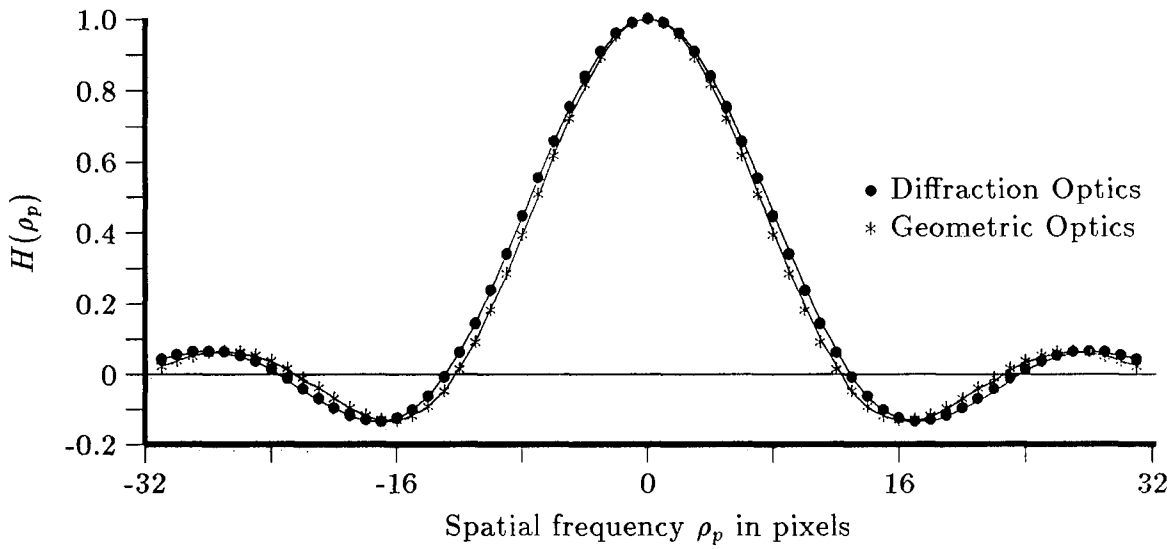


Figure 3.3: Cross-sections of optical transfer functions for a point object at a distance $D_o = 0.95\ m$ from camera.

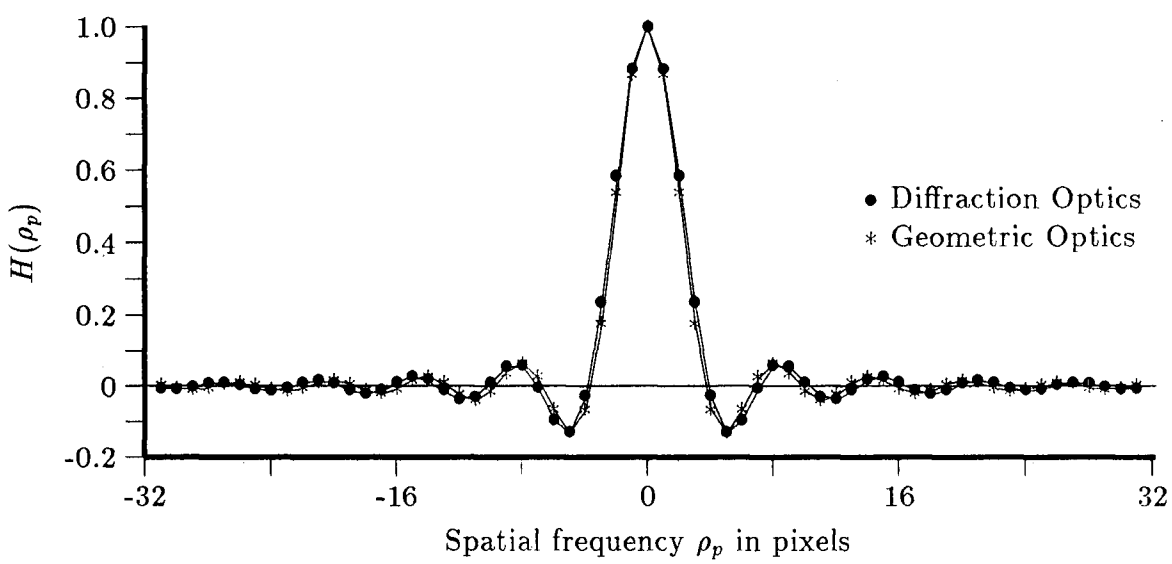


Figure 3.4: Cross-sections of optical transfer functions for a point object at a distance $D_o = 0.85\ m$ from camera.

- The f-number is $f = 1.3$; and
- The digitization ratio of the camera is $P = 60.0$ pixels$/mm$.

Then using (3.15), (3.24) and (3.22), the optical transfer function for geometric optics can be calculated. Using (3.28), (3.26), (3.31), (3.29), (3.33) and (3.32) the optical transfer function for diffraction optics can be calculated. Figure 3.3 shows a cross-section of these two optical transfer functions for relatively little defocus, a point object at a distance $D_o = 0.95\ m$. Figure 3.4 shows the same optical transfer functions, but for a point object at a distance $D_o = 0.85\ m$, which is a large amount of defocus. In the spatial frequency domain, the two transfer functions obtained by geometric optics and diffraction optics, are quite similar.

3.6.2 Point Spread Function

Born [16] and Potmesil et al [122] have shown that the point spread function for diffraction optics can be expressed as a complex series of Bessel functions. But since the optical transfer function has already been presented and calculated for two examples in Section 3.5.1, a simpler approach will be to take the inverse Hankel transform [116][144][52] of the cross-sections in Figures 3.3 and 3.4.[6] The point spread function for geometric optics is given by (3.15) and (3.9).

Figures 3.5 and 3.6 show the point spread functions for point object distances of $D_o = 0.95$ and $D_o = 0.85\ m$ respectively. It can be seen quite clearly that the point spread functions for the two models are very similar. Considering the heavy computational requirements for the diffraction optics model, for the optical parameters used in the examples, the geometric optics model provides an adequate model for defocus.

[6] Or, the inverse Fourier transform of the full two-dimensional function can be taken and the cross-section of the point spread function extracted. This technique, although more computationally intensive than the inverse Hankel transform, yielded better results. It should also be noted that taking a discrete

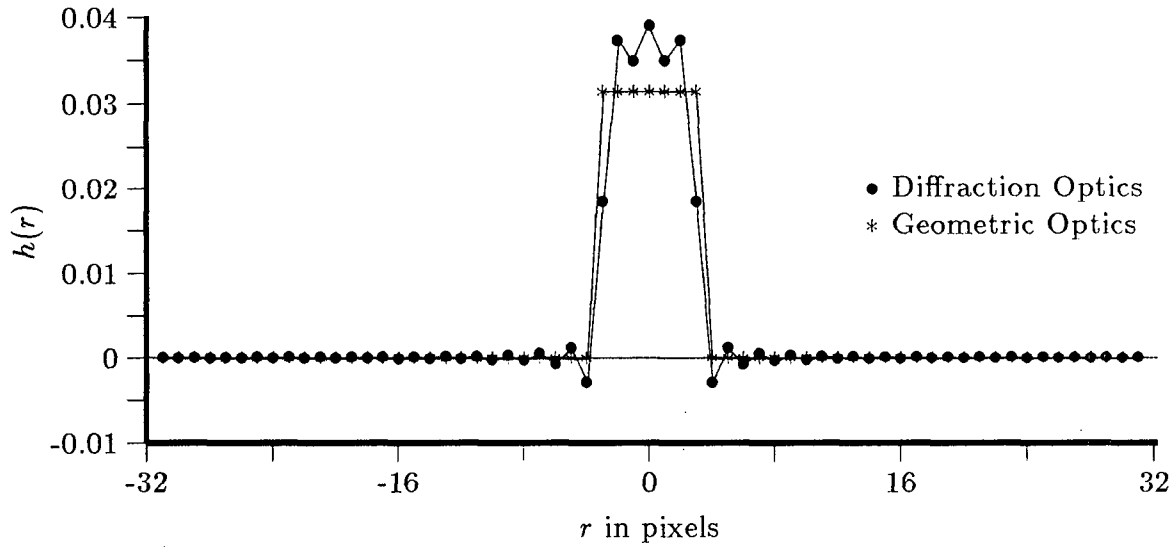


Figure 3.5: Cross-sections of point spread functions for a point object at a distance $D_o = 0.95\ m$ from camera.

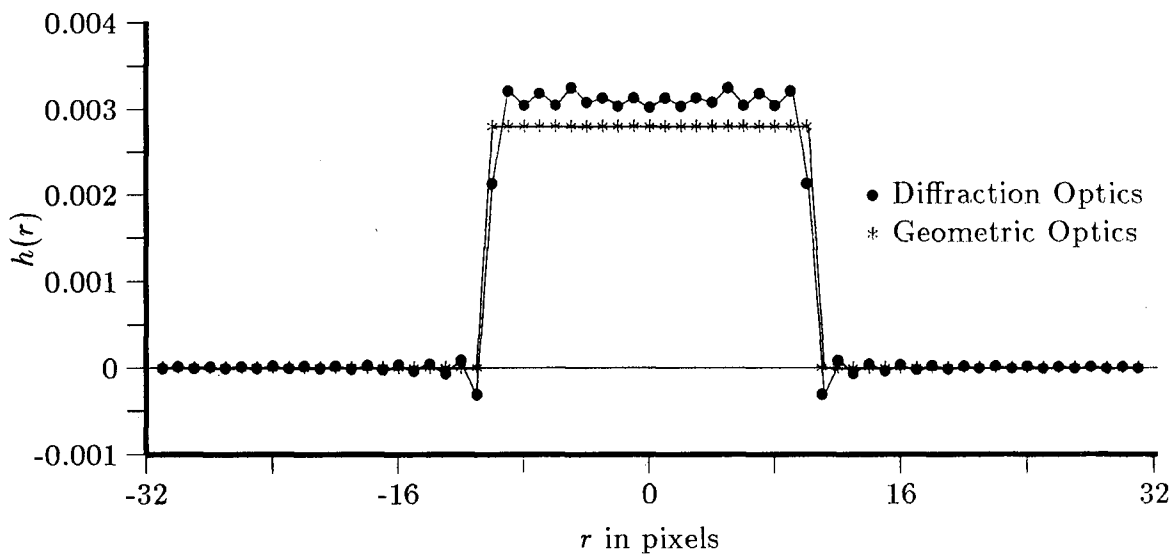


Figure 3.6: Cross-sections of point spread functions for a point object at a distance $D_o = 0.85\ m$ from camera.

3.7 WORKING AREA AND RESOLUTION

The *working area* of depth from focus, is the range of object distances which can be recovered. That is, the working area is between the *closest* object distance and the *farthest* object distance which can be accurately measured. If an application calls for a specific working area, the optical parameters of the camera can usually be selected to accommodate this requirement. Once the working area is known, the required resolution can be calculated.

3.7.1 Working Area Of Depth From Focus

The working area is governed by (3.16),

$$D_o = \frac{FD_i}{2fR + D_i - F} \,. \tag{3.34}$$

Using this equation requires two steps.

1. A digital, depth from focus algorithm will usually require the radius of the defocus operator to be within an optimal range of pixel values ($R_{p(\min)}$ to $R_{p(\max)}$). That is, if the defocus operator is too small then it cannot be recovered, and if the defocus operator is too large, then it will not be contained within the local area examined. Using (3.17) converts $R_{p(\min)}$ and $R_{p(\max)}$ in pixel values to distance units $R_{(\min)}$ and $R_{(\max)}$. This determines the range of R.

2. After the range of R is determined, then D_i, F, and f can be experimentally adjusted as desired (although they are constrained to values obtained by available lenses), to achieve the desired working area $D_{o(\min)}$ to $D_{o(\max)}$.

---

transform can introduce small aliasing effects into the result, however it is not critical to this comparison.

Therefore, using (3.17) and (3.34), the working area is between

$$D_{o(\max)} = \frac{FD_i}{\frac{2fR_{p(\min)}}{P} + D_i - F}, \tag{3.35}$$

and

$$D_{o(\min)} = \frac{FD_i}{\frac{2fR_{p(\max)}}{P} + D_i - F}. \tag{3.36}$$

To illustrate the selection of a working area, the same camera optical parameters used for the experiments in Chapter 8 will be assumed. For this example, let

- The blur circle radius vary from $R_{p(\min)} = 3$ to $R_{p(\max)} = 15$ pixels;
- The focal length of the lens be $F = 50$ mm;
- The f-number be $f = 1.3$;
- The distance from the lens to the image plane be adjustable from $D_i = 50$ to 52 mm, which corresponds to the point of perfect focus adjusted from infinity to 1.3 *meters*; and
- The digitization ratio $P = 60$ pixels/mm.

Substituting these parameters and ranges into (3.35) and (3.36), produces the working area shown in Figure 3.7. For any given setting of the image plane D_i, the black line above it represents the object distances D_o which can be detected.

It can be seen from the graph, that for a 50 mm lens, the working area can potentially be very large. For the lens focused close to infinity, the working area is about 4 to 18 meters and for the lens focused at one *meter*, the working area is about 80 to 95 cm. Different working area distributions can be obtained by selecting other values for the focal length F, and the f-number f. Selecting either a shorter focal length or a larger f-number, moves the working area closer to the camera.

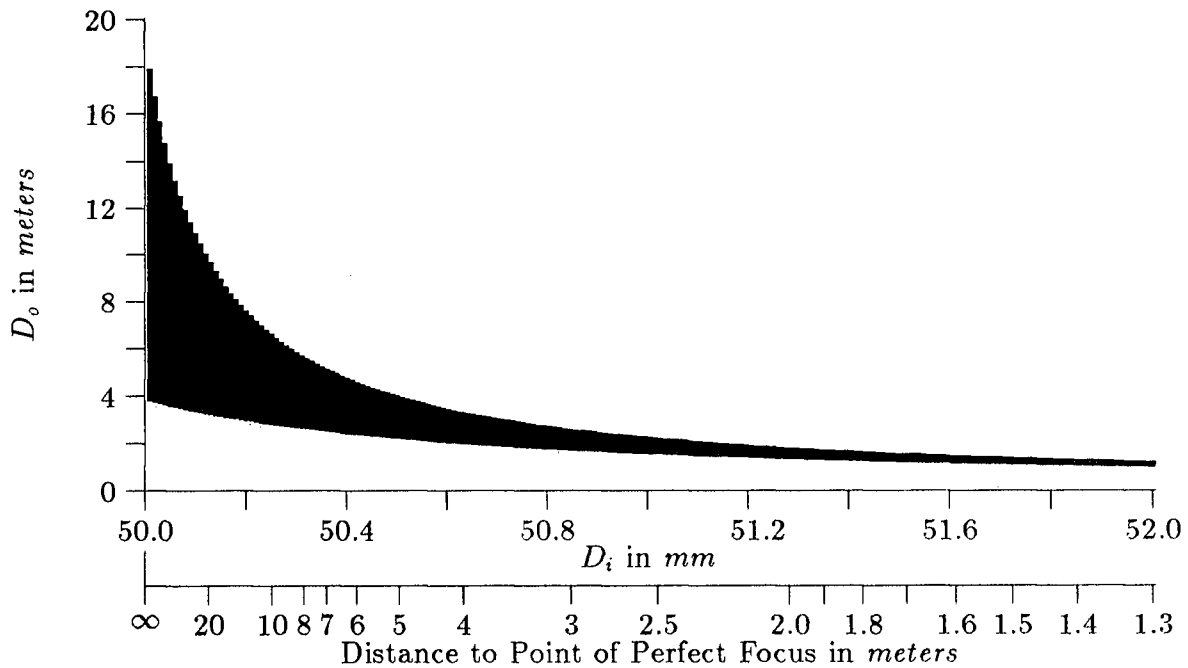


Figure 3.7: Example of working area of depth from focus, for the focal parameters given in the text.

3.7.2 Scene And Camera Resolution

Depending on the nature of the application, the resolution problem may be stated in one of two ways:

1. The smallest size of objects in the scene is known, and the camera resolution is desired; or

2. The camera resolution is known, and the size of objects that can be discerned is desired.

The lateral magnification m, of a simple lens system is [82, p. 168]

$$m = \frac{D_{fi}}{D_o} = \frac{S_i}{S_o}, \tag{3.37}$$

where S_o is the size of the object and S_i is the size of the imaged object. Using (3.14), (3.37) becomes

$$\frac{S_i}{S_o} = \frac{F}{D_o - F}. \tag{3.38}$$

For the first resolution problem, S_o is given and S_i is calculated, and for the second resolution problem S_i is given and S_o is calculated. The optical parameters F and D_o must be known in both cases.

For example, if

- The camera resolution is $S_i = 1/60\ mm$ per pixel;

- The focal length $F = 50\ mm$; and

- The object distance $D_o = 1\ meter$.

Then, using (3.38), any object less than $S_o \approx 1\ mm$ cannot be resolved.

3.8 CONCLUSIONS

Defocus can be modelled as convolution by a point spread function. The nature of this point spread function can be modelled in two traditional ways: geometric optics and diffraction optics. For the optical camera parameters to be used in the experimental analysis of Chapter 8, it was shown that the diffraction optics model yields a similar result to geometric optics. Considering the complexity of the diffraction optics model, geometric optics should provide an adequate theoretical representation.

The equations for determining the working area of depth from focus were presented. The location and depth of this working area is determined by selecting the appropriate optical parameters. From the example, it was shown that for a 50 *mm* lens focused close to infinity, the working area is about 4 to 18 *meters* and for the lens focused at close distances, the working area is about 80 to 95 *cm*. Selecting either a shorter focal length, or a larger *f*-number moves the working area closer to the camera. Equations were also given for calculating the scene resolution, given the camera resolution, and vice versa.

Chapter 4

LITERATURE REVIEW

4.1 INTRODUCTION

This chapter is a comprehensive summary of the work done by other researchers in solving the depth from focus problem. Analysis is deferred to Chapter 5. References were gathered by consulting

- Results of a bibliographic database search;
- Proceedings of major conferences in Computer Vision and Pattern Recognition;
- Surveys of ranging techniques [79][73][77][135][128][113][9][11][10];
- Results of a Patent search; and
- Citations from papers in the field.

4.2 OVERVIEW

Section 4.3 presents methods of *depth from automatic focus*. Early researchers tried to gain an understanding of the biological vision system and develop similar methods for *automatically focusing* a camera on the point of interest. The distance to the point in the scene could then be determined from the position of the lens. However, a key limitation of this technique was that it measured depth, *one* point at a time and required an adjustment of the lens setting for each point.

Section 4.4 looks at methods of calculating depth by measuring the amount of defocus, called the *focal gradient*, throughout the image. Since one unfocused image of an unknown scene is an underconstrained problem (soft edges in the images may be either unfocused hard edges in the scene, *or* focused soft edges in the scene), three approaches exist:

- Introduce extra cues into the scene;
- Make assumptions about the scene; or
- Take several images with different defocus operators.

Section 4.5 examines patented approaches to solving the depth from focus problem, and the last section looks at the work which has begun in integrating depth cues from defocus, with other means of depth perception.

4.3 DEPTH FROM AUTOMATIC FOCUS

The technique of depth from automatic focus uses the following simple operating scenario for each depth point:

1. Identify a local region in the image for which a depth value is desired;
2. Automatically focus on that region, through a feedback loop which uses an algorithm to assess the quality of the image, and moves the lens to obtain the position of sharpest focus; and
3. Use the position of the lens to calculate the distance to the region of interest.

The most difficult task, of course, is devising a robust automatic focusing algorithm.

4.3.1 Biological Accommodation

Since humans are able to effortlessly and accurately perform this task, early studies concentrated on understanding and modelling the human accommodation system. An extensive survey was done in 1966 by Crane [36], who summarized the available literature at that time,[1] and attempted to model the human visual accommodation system. He did so, by trying to answer the following three questions:

1. What portion of the retinal picture is involved in accommodation control;
2. How that portion of the picture is processed to derive a measure of defocus; and
3. How that signal in turn is used to control the ciliary muscles.

He concluded that the relevant portion of the retina is the fovea, but more importantly, he showed how neural circuits based on lateral inhibition produce a measure of defocus.[2] To model the control of the ciliary muscles, he proposed a system which used the information from the accommodative saccades to increase the depth of field.

4.3.2 Image Processing For Automatic Focus

In the late sixties, the work on automatic focusing seemed to depart from the psychophysics and lay more purely in the image processing domain. Since defocus tends to limit the presence of high spatial frequencies, several schemes were developed which assessed the relative presence of high spatial frequencies.

In 1968, Horn proposed and implemented an algorithm for automatic focusing, based on the Fourier transform [65]. He took a one-dimensional FFT through the

[1] Crane primarily cites the work of F.W. Campbell (eg. [24][22]), who published numerous papers in the late fifties, outlining the parameters and constraints of human accommodation.

[2] This concept of visual *channels*, that are sensitive to specific spatial frequencies, was later developed by Campbell and Robson [23] and Marr [107].

region of interest, and obtained the sum of the energy in a region greater than some minimum frequency. This was then divided by the energy in a lower frequency window to normalize it. In principle, this was a good approach, however it was computationally expensive, since the FFT computed information concerning the magnitude and phase information for each frequency band, which was not explicitly required.

Another approach, explored by Tenenbaum in 1970 [145], examined the effect of defocusing an image, by the changes in the nature of edges. Since defocusing tends to decrease the gradient of an edge, the principle employed was to find the point of greatest focus by trying to maximize the *magnitude of the gradient* in the region of interest. The gradient was found using the Sobel operator,

$$\textit{magnitude of the gradient} = \sqrt{i_x^2 + i_y^2}\ , \tag{4.1}$$

and the convolution kernels

$$i_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} \quad i_y = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix} . \tag{4.2}$$

In 1972, Mendelsohn, Mayall and Kujoory [109][93] proposed a simple algorithm for focusing human metaphase chromosomes viewed under a microscope. The algorithm was maximizing the *intensity over a threshold*,

$$\textit{intensity over a threshold} = \sum_x \sum_y (i(x,y) - \Psi)\ , \qquad i(x,y) > \Psi\ , \tag{4.3}$$

where $i(x,y)$ is the image intensity and Ψ is a threshold set in the midrange of image greyness. This technique was also implemented by Johnson and Goforth in 1974 [76], and Mason and Green in 1975 [108]. It appears, however, that this technique was not easily generalized to other scenes.

Preston designed a feed-back mechanism in 1973, which focused a microscope by maximizing the high frequency content of a video photomultiplier during a raster scan of the slide [124]. A similar algorithm using a video camera was also implemented by Johnson and Goforth [76]. Also in 1973, Ohteru et al. combined autofocus with pattern recognition for their WABOT project [115].

In 1974, Muller and Buffington [110] proposed three new criteria for automatic focusing

1. *Squared gradient*: The difference between pixels was squared and maximized. For example, horizontally

$$squared\ gradient = \sum_x \sum_y \left[i(x,y) - i(x,y-1)\right]^2 \ . \tag{4.4}$$

2. *Laplacian*: The second derivative, found by

$$Laplacian = \nabla^2 i(x,y) = \frac{\partial^2 i}{\partial x^2} + \frac{\partial^2 i}{\partial y^2} \ , \tag{4.5}$$

was maximized, using the kernel

$$Laplacian = \begin{pmatrix} 1 & 4 & 1 \\ 4 & -20 & 4 \\ 1 & 4 & 1 \end{pmatrix} . \tag{4.6}$$

3. *Signal power*: The image energy was maximized,

$$signal\ power = \sum_x \sum_y \left[i(x,y)\right]^2 \ . \tag{4.7}$$

Erteza also implemented the *squared gradient* and *Laplacian* methods in 1976 [42][41].

Jarvis, in 1976 [72] proposed three other focus criteria.

1. *Histogram entropy*: If $P(i)$ was the frequency of occurrence of the grey-level i, then

$$histogram\ entropy = -\sum_i P(i)\ln[P(i)]\ , \qquad P(i) \neq 0\ , \tag{4.8}$$

was to be minimized.

2. *Grey level variance*: If the grey levels were viewed as random variables, then the criterion was to maximize the variance,

$$grey\ level\ variance = \frac{1}{N^2}\sum_x\sum_y[i(x,y) - \mu^2]\ , \tag{4.9}$$

where μ is the mean of the grey levels.

3. *Sum-modulus-difference*: This was perhaps the simplest criterion proposed. The principle was to maximize the absolute value of the intensity difference between neighbouring pixels. Implemented horizontally, this would be

$$sum\text{-}modulus\text{-}difference = \sum_x\sum_y \mid i(x,y) - i(x,y-1) \mid\ . \tag{4.10}$$

Jarvis found that all three criteria showed considerable promise in focus optimization, and were relatively easy to calculate. Of the three, the *sum-modulus-difference* was the least ambiguous for the images tested.

A concise comparison of most of the previously mentioned algorithms was performed by Ligthart and Groen in 1982 [101]. They concluded that the *squared gradient* and the *grey level variance* methods yielded the best results. Also in 1982, Hanma et al. developed a hardware technique of maximizing the high frequency content of the video signal [58], to implement automatic focusing in a video camera.

In 1983, Schlag et al. [131][19], evaluated five autofocusing methods, and found the *magnitude of the gradient* to be best. A few years later, in 1987, Yamamoto et al.

implemented an autofocus camera system in hardware [154]. Their method maximized the height of the first derivative.

Also in 1987, Luo et al. [103][104] presented two slightly different algorithms

1. *Sum*: The total sum of the grey levels was maximized,

$$sum = \sum_x \sum_y i(x,y) \ . \tag{4.11}$$

2. *Histogram variance*: The variance in the histogram of grey levels was maximized,

$$histogram\ variance = \frac{1}{\sum_i N(i)} \sum_i \left(i - \overline{N}(i)\right)^2 \ , \tag{4.12}$$

where $N(i)$ was the number of pixels with intensity i and $\overline{N}(i)$ was the mean.

The most recent work on depth by automatic focusing, was done by Krotkov [90][84][88][83][91][92], who implemented most of the previously mentioned algorithms. Krotkov found after experimenting with several different images, that maximizing the *magnitude of the gradient*, found with the Sobel operator, proved superior to the others in monotonicity about the peak and robustness in the presence of noise. Using this technique, he was able to discern object distances between 1 and 3 meters to an accuracy of 2.5%.

4.3.3 Computing Depth From Lens Position

Once the position of the lens has been established through the use of an automatic focusing algorithm, the physical distance from the camera to the scene can easily be calculated with simple *geometric optics* [114]. For the thick-lens geometry shown in Figure 4.1, the Gaussian lens law (2.1) holds

$$\frac{1}{D_o} + \frac{1}{D_i} = \frac{1}{F} \ . \tag{4.13}$$

For a particular lens, the position of the principal planes, P1 and P2 can usually be found by calibration with known distances.

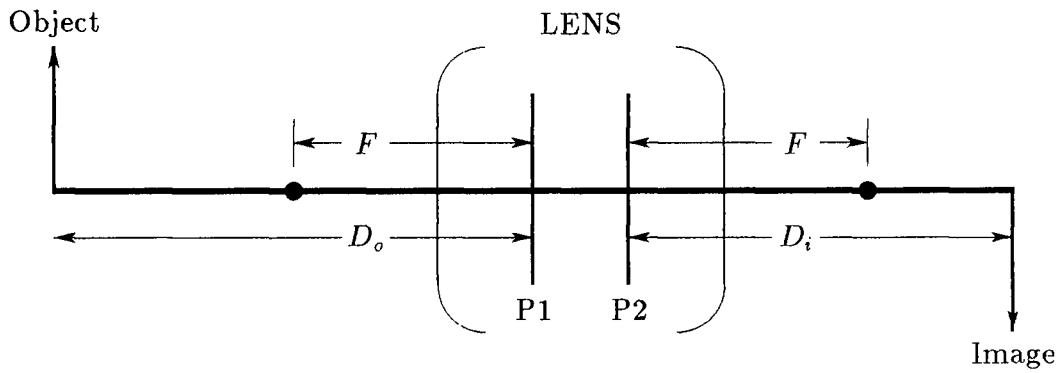


Figure 4.1: Thick-lens geometry. P1 and P2 are the principal planes and F is the focal length.

4.3.4 A Pyramid Approach

A key limitation of determining depth by automatic focusing, is that it measures the depth, *one* point at a time, and requires an adjustment of the lens setting for each point. This process was streamlined in 1987 by Darrell and Wohn [37][38], who acquired images at five different lens positions. Then using Laplacian and Gaussian pyramids to isolate frequency ranges, they were able to interpolate depth to various points in the scene.

Although this process makes it possible to acquire depth at many points in the scene, it would be more desirable to find a method that requires only one or two images of the scene. The next section examines that possibility.

4.4 DEPTH FROM FOCAL GRADIENT

4.4.1 Concept Of Focal Gradient

If the amount of *defocus* could be measured throughout the image, then this would reduce the number of images needed. In 1970 Cornsweet [35, pp. 45–47] recorded that two images formed with different apertures were a source of depth information.

Jarvis also hinted at this in 1976 [72], however the first experimental work was done by Pentland in 1982 [118]. He used the term *focal gradient* [121] to describe the amount of defocus throughout the image. Then depth could be estimated at many points of the scene.

Unfortunately, one unfocused image of an unknown scene does not contain enough information to recover focal gradient. Soft edges in the images may be either unfocused hard edges in the scene, *or* focused soft edges in the scene. To further constrain the problem three approaches exist:

- Introduce extra cues into the scene;
- Make assumptions about the scene; or
- Get more information by taking several images of the same scene with different defocus operators.

4.4.2 Introduction Of Extra Cues

Several researchers have taken a number of novel approaches to constrain the scene with various forms of structured lighting. Although structured lighting is a common technique used in range finding [73][9][10], most methods assume that all parts of the image are in focus. The methods presented in this section take advantage of a *limited* depth of field, where defocused regions are observed.

One technique, called the BIRIS system, was reported by Blais and Rioux in 1986 [14][127]. The extra cue was vertical lines of laser light scanned across the scene. The depth sensor consisted of a CCD camera fitted with a mask, having two small apertures on a horizontal axis. As shown by Figure 4.2, the mask caused an object which was illuminated at point B, and because of its position was out of focus, to

project two points, b' to the camera. The distance between these two points could then be triangulated to discern the depth to the object.

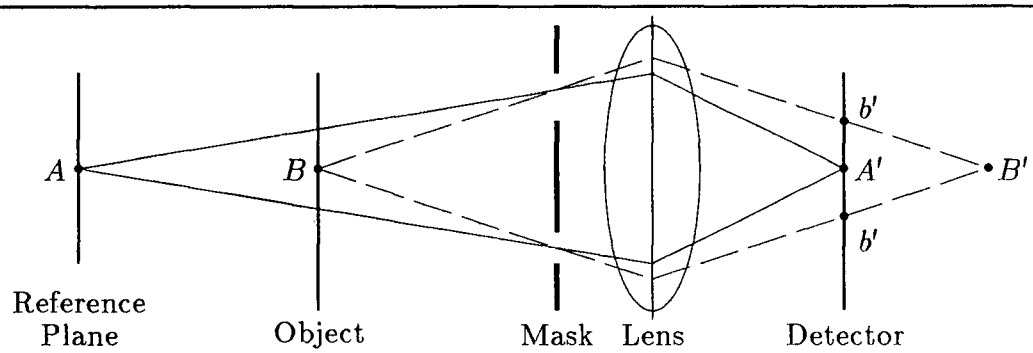


Figure 4.2: Basic principle of BIRIS system.

Also in 1986, Kinoshita et al. developed a range sensor based on a conical beam of light passing through an objective lens [81]. The lens was moved until the circle of light became a point, and range was inferred from the position of the lens. A simple photodiode was used as the sensor.

Another technique was developed by Engelhardt and Häusler in 1988 [40]. Similar to the Talbot effect, which uses the self-imaging property of coherently illuminated gratings [26][96], they used an incoherently illuminated grating and detected spatial frequencies.

On a much smaller scale, Corle et al. in 1987, used focusing principles to measure distances down to 0.04μm [34]. The sample was vibrated in a Type II confocal scanning microscope to obtain a differential measurement.

The most recent work in using structured lighting was by Girod and Scherock in 1989 [47]. Range was calculated from the standard deviation of intensities along illuminated stripes. Unsymmetrical illumination patterns were used to solve the ambiguity about the point of perfect focus.

4.4.3 Assumptions Made About The Scene

The second approach to constraining the problem is to make some assumptions about the scene.

Sharp edges assumed

A common, and sometimes reasonable assumption, is that edges in the scene are sharp black to white transitions. Tenenbaum in 1970 [145], measured the blur at step discontinuities in the scene. This assumption was also used recently by Pentland in 1982 [118][121][120],[3] who used the Laplacian to measure the width of the blurring function. Grossmann's work in 1987 [56], was similar to Pentland, except he used the directional gradient of the image. Additionally, Grossmann considered the case of ramp edges.

In 1988, Garibotto and Storace expanded on the work of Pentland and Grossmann by looking at the ratio of the directional gradient of the image convolved with two Gaussians of different width [45].

The work of Pentland, and Garibotto and Storace assumed that the point spread function of defocus was a Gaussian. Subbarao in 1988 [143], removed that assumption, so that any rotationally symmetric point spread function could be used. He presented a method for recording the *line spread function* of a camera.[4] Depth recovery of a step edge was done by measuring the *spread parameter*, which was considered to be the *square root of the second central moment* of the camera's *line spread function*.

Bandwidth of edge assumed known

As a different approach, Sayeh et al. [130] introduced some general equations relating the bandwidth of an edge, to degree of focus and hence distance. But no experimental

[3] Pentland also considered other cases which are presented in Section 4.4.4.
[4] Other work by Subbarao is presented in Section 4.4.4.

results were given, and it is not clear how the bandwidth of the unfocused edge could be robustly estimated.

Defocus operator assumed space-invariant

In 1987, Chuang presented a novel approach which assumed that the defocus operator did not change over a large area in the image (ie. the objects in the scene were all the same distance from the camera)[29]. One image of the scene was taken, and spatially displaced slices were extracted. By using Cepstrum analysis [15][27][28], the spectral effects of the scene tended to average out, while space-invariant spectral characteristics of the defocus operator remained. Once the defocus operator was identified, distance could easily be determined.

4.4.4 More Than One Image Taken

The third, and most general approach to constraining the ambiguity of defocused edges, is to acquire more than one image of the same scene but with different defocus operators. This is the most difficult, but also the most interesting work in deriving depth from focus. Researchers have used numerous methods for isolating the defocus operator from the scene content.

Theory of varying the defocus operator

The defocus operator can be changed by varying one or more of three camera parameters

- Position of image plane;
- Focal length; or

- Aperture.

Assuming that the point spread function was a Gaussian, Subbarao derived the equations which related changes in each of these parameters to the Gaussian spread parameter and hence the distance to the scene. This was done for both *infinitesimal* changes [138][137], and *finite* changes [139]. In a follow-up paper [136] this was also extended to other defocus operators.

Inverse filtering in the spatial frequency domain

The most popular, and most general, method of isolating the defocus operator from the scene, is inverse filtering in the spatial frequency domain.

Pentland introduced this work in 1982 [118], with follow-up papers in 1985 [121], 1987 [120] and 1989 [117]. He assumed that the defocus operator was a Gaussian. As already mentioned in Section 4.4.3, one of his approaches was to further assume sharp edges. The other, more general approach, was to acquire two images with two different aperture settings. This was formulated as

$$\frac{i_1(r,\theta)}{i_2(r,\theta)} = \frac{s(r,\theta) \otimes g(r,\sigma_1)}{s(r,\theta) \otimes g(r,\sigma_2)} \tag{4.14}$$

where

- $s(r,\theta)$ is the scene in focus;
- $i_1(r,\theta)$ is a local region of the image blurred with $g(r,\sigma_1)$ the Gaussian representing the defocus operator for one aperture and;
- $i_2(r,\theta)$ is a local region of the image blurred with $g(r,\sigma_2)$ the Gaussian representing the defocus operator for the other aperture.

If the following are Fourier pairs,

$$s(r,\theta) \iff S(\rho,\phi)\ , \tag{4.15}$$

$$i_1(r,\theta) \iff I_1(\rho,\phi)\ , \tag{4.16}$$

$$i_2(r,\theta) \iff I_2(\rho,\phi)\ , \tag{4.17}$$

$$g(r,\sigma_1) \iff G\left(\rho, \frac{1}{\sqrt{2\pi}\ \sigma_1}\right)\ , \text{ and} \tag{4.18}$$

$$g(r,\sigma_2) \iff G\left(\rho, \frac{1}{\sqrt{2\pi}\ \sigma_2}\right)\ , \tag{4.19}$$

$$\tag{4.20}$$

then

$$\frac{I_1(\rho)}{I_2(\rho)} = \frac{G(\rho,\sigma_1)\sigma_2}{G(\rho,\sigma_2)\sigma_1}\ , \tag{4.21}$$

$$= \frac{\sigma_2^2}{\sigma_1^2}\exp\left(\rho^2 2\pi^2(\sigma_2^2-\sigma_1^2)\right)\ , \tag{4.22}$$

where an integration has occurred over ϕ.

Then after taking the natural log of (4.22), formulating it as a linear regression of ρ^2, and assuming that one aperture was a pinhole ($\sigma_1 = 0$), Pentland gave this equation for distance to the scene,[5]

$$D_o = \frac{FD_i}{D_i - F - \sigma_2 kf}\ , \tag{4.24}$$

where k was an experimentally determined constant.

In his latest paper [117], Pentland's technique was extended. He estimated the power in one or more frequency bandwidths, for one image taken with a pinhole aperture, and one image taken with another aperture. These two values were then compared

[5] Pentland uses the relation $f = F/r$ for the f-number of a lens, where r is the radius of the aperture. Since the correct relation is $f = F/(2r)$ [66, p. 208][60, p. 152], (4.24) should be

$$D_o = \frac{FD_i}{D_i - F - 2\sigma_2 kf}\ . \tag{4.23}$$

to a lookup table to produce an estimate of range. Parseval's theorem was used to perform the calculations in the spatial domain.

Pentland also extended his data acquisition equipment to include a two camera system which acquired two images, at different apertures simultaneously. By performing the calculations in hardware, he was able to process an impressive four images a second, with an RMS accuracy of 2.5%.

Using a similar two-camera system and acquiring one image with a pinhole aperture, Bove [17] described a slightly different technique. He took the two-dimensional Fourier transform of windowed regions in each image, collapsed them to the radial dimension and divided the two frequency domain signals. Then he used a higher order regression solution than Pentland, to model the division result to a polynomial approximation of the *geometric optics* model of defocus. No accuracy results were given.

Subbarao has performed the theoretical work for a more general solution [136][140]. He removed Pentland's constraint of one image being formed by a pin-hole camera, and gave a closed form solution assuming a Gaussian defocus operator. The reader is referred to the references for the details, but the key equation was

$$(\alpha^2 - 1)\sigma_2^2 + 2\alpha\beta\sigma_2 + \beta^2 = \frac{1}{A}\int\int_R \frac{-2}{f_x^2 + f_y^2} \ln\left(\frac{I_1(f_x, f_y)}{I_2(f_x, f_y)}\right) df_x df_y \tag{4.25}$$

where

$$\alpha = \frac{k_1 L_1}{k_2 L_2} \tag{4.26}$$

$$\beta = k_1 L_1 \left(\frac{1}{F_1} - \frac{1}{F_2} + \frac{1}{D_{2i}} - \frac{1}{D_{1i}}\right) \tag{4.27}$$

and

- k_1 and k_2 were experimentally derived constants;
- R was the local region in (f_x, f_y) frequency space;

- A was the area of local region R;
- $I_1(f_x, f_y)$ and $I_2(f_x, f_y)$ were the Fourier transforms of the two local image regions; and
- σ_2 was the spread parameter of the Gaussian defocus operator.

Note that some, or all of the camera parameters L_1, L_2 or F_1, F_2 or D_{1i}, D_{2i} may be changed, although if $F_1 \neq F_2$ or $D_{1i} \neq D_{2i}$ then the images must be adjusted for magnification changes. Changing the aperture seems more convenient since only a brightness adjustment is needed.

Equation (4.25) needs to be solved for σ_2 and the distance to the object is obtained from

$$D_o = \frac{k_2 L_2 D_{2i} F_2}{k_2 L_2 (D_{2i} - F_2) - F_2 \sigma_2} \,. \tag{4.28}$$

Subbarao also extended this technique to other models of the defocus, by generalizing σ to be the *square root of the second central moment* of the defocus operator. He claimed that (4.28) still holds. Subbarao gave no experimental results to confirm these methods.

Other methods for isolating the defocus operator

Other methods for isolating the defocus operator have also been presented in the literature .

In 1988, Jarvis [74] defined a ratio R as

$$R = \begin{cases} \dfrac{|\nabla (\nabla^2 G(r,\sigma) \otimes i_1(x,y))|}{|\nabla (\nabla^2 G(r,\sigma) \otimes i_2(x,y))|} & \text{(at } \nabla^2 G \otimes i_1(x,y) \text{ zero crossings)} \\ \text{undefined} & \text{(elsewhere)} \end{cases} \tag{4.29}$$

where

- $G(r, \sigma)$ was the Gaussian;
- $i_1(x, y)$ was an image obtained with a pin-hole aperture; and
- $i_2(x, y)$ was an image obtained with a larger aperture and objects were more out of focus towards the front.

R was then uniquely related to distance. This method had a good intuitive basis, but no theory proving that it would work for all types of edges was presented. The technique was tried on real images, but no accuracy results were given.

Another interesting technique was implemented by Hwang, Clark and Yuille in 1988 [70][69]. They proposed a differential algorithm which exploited the following property of the Gaussian,

$$\frac{\partial}{\partial \sigma} G(r, \sigma) = \sigma \nabla^2 G(r, \sigma) \ . \tag{4.30}$$

Then the relation t is related to distance to the scene,

$$t = \frac{\frac{\partial i(x, y)}{\partial D_i}}{\nabla^2 i(x, y)} \ , \tag{4.31}$$

where D_i is the distance from the lens to the image, and $i(x, y)$ is the imaged scene. Hwang et al. took two images, varying D_i, although this method could be adapted to a change in aperture. Since the technique is implicitly dependent on (4.30), it will not generally work for focus operators other than the Gaussian.

4.5 PATENTED APPROACHES

The patent literature reveals developments in depth from focus systems, which because of proprietary reasons, may not be found in the scientific literature. Patents, by nature, usually address very specific problems. In the patent search done by the author, a large number of patents were found which used information concerning the state of

focus to determine the distance to a *single* object. No patents were found which used information concerning the state of focus to determine the distance to a plurality of objects in the same scene simultaneously. Some of the techniques for single object distance detection, may however, be extended to determine depth over an entire scene.

4.5.1 Camera Autofocus

Most of the Assignees for these patents were the manufacturers of Single-Lens Reflex cameras, who require "through-the-lens" automatic focusing methods, and need to find the distance to the center of the image. Golberg [48], explained three schemes that have been patented.

Pentax autofocus

The Pentax system did something which the human eye cannot do: it simultaneously compared the quality of focus at two different focal lengths. It employed a beam splitter which optically placed two one-dimensional detectors equidistant on either side of, and parallel to, the film plane. When the film plane was in focus, the pattern on each detector strip, *although out of focus*, was identical. The feedback loop to control the lens position compared the two detector strips, and indicated if the lens was going in the right direction, as well as when the scene was in focus.

Honeywell autofocus

The Honeywell system [94][39] was based on the relationship between an object and the resulting light distribution which leaves the exit pupil of the lens. When the object was in focus, the intensity profile across the pupil was constant, regardless of the viewing angle from the optical axis. The system had a row of detector pairs perpendicular to

the optical axis, with tiny microlenses that split light from each half of the exit pupil onto its corresponding detector. The object was in focus, if the light distribution on all detectors was equal along each half.

Leitz Correfot system

The Leitz Correfot System employed the same principle as the Honeywell system, except only two pair of detectors were used and a prismatic grating scanned perpendicular to the optical axis.

4.5.2 Other Depth From Focus Patents

Other techniques in the patent literature included:

- A beam splitting arrangement [111] which optically produced multiple images of the object (similar to Blais [14][127]), where the distance between the images relates to the distance to the object.
- A device [71] which correlated various parts of the image to form weighted signals corresponding to three different distances. These weighted signals were interpolated to determine the distance to the object.
- A device [112] which correlated the point spread function of a defocused object in two different optical planes to determine the distance to the object.

4.6 COMBINING DEPTH FROM FOCUS WITH OTHER METHODS

As indicated in the introductory chapter, a robust vision system for discerning depth must integrate several means of depth perception. Several researchers have already begun to combine depth from focus systems with other methods.

Subbarao combined his earlier work in depth from focus outlined in Section 4.4, plus work in monocular motion recovery to propose a new system for binocular recovery of depth and motion [142]. Krotkov and Kories have combined their depth from focus system together with a stereo ranging system [89][86][87][85].

Abbott and Ahuja have sought to dynamically integrate focus, camera vergence and stereo [2], and Clark and Ferrier [33] addressed the problem of controlling an attentive vision system with depth from focus as one of its attributes.

4.7 CONCLUSIONS

This chapter has summarized the work of other researchers in solving the depth from focus problem. Early researchers tried to gain an understanding of the biological vision system, and develop similar methods for automatically focusing a camera on the point of interest. The distance to the point in the scene could then be determined from the position of the lens. However, a key limitation of this technique was that it measured depth, *one* point at a time and required an adjustment of the lens setting for each point. Key papers were published by Horn [65], Jarvis [72], Ligthart and Groen [101], Krotkov [90][84][88][83][91], and Darrell and Wohn [37][38].

More recently, researchers have attempted to measure the amount of defocus, called *focal gradient* in different parts of an image and therefore infer distance to a variety of points in the scene. Since one unfocused image of an unknown scene was an underconstrained problem (soft edges in the images may be either unfocused hard edges in the scene, *or* focused soft edges in the scene), three approaches were presented:

- Introduce extra cues into the scene;
- Make assumptions about the scene; or

- Take several images with different defocus operators.

The key papers published in this area were by Pentland [118][121][119][120][117], Subbarao [138][137][139][142][143][136][141][140] and Bove [17].

Chapter 5

ANALYSIS OF DEPTH FROM FOCUS PROBLEM

5.1 INTRODUCTION

This chapter builds on the literature review of Chapter 4. The problem of depth from focus is defined and general purpose approaches to the problem are outlined. This chapter provides the necessary background for Chapter 6, which describes the theoretical basis for the author's new approach.

5.2 OVERVIEW

Section 5.3 defines the problem of depth from focus, which is addressed by this thesis. To sift through the variety of solutions presented in Chapter 4, Section 5.4 describes the desired characteristics of solutions to the problem. It is desirable for the solutions be as general as possible. The key problems and trade-offs in solving the problem are itemized in Section 5.5. The last section, draws from the work of other researchers to outline approaches to general purpose solutions of the problem.

5.3 PROBLEM DEFINITION

The problem addressed by this thesis, is given a blocks scene with featured objects an unknown distance from the camera, but within a known range, derive the distance to features in the scene using the concept of depth from focus. This will be attempted in a setting that is as generalized as possible.

5.4 DESIRED CHARACTERISTICS OF SOLUTIONS

This section defines the desired characteristics of solutions to the problem of depth from focus. Although it is not possible to define the "best" solution to the problem, this section seeks to bring out characteristics which would allow the solutions to be applied to the broadest number of applications.

5.4.1 General Purpose Solutions

A key motivation in this research was to investigate general solutions to the problem, rather than a solution which took advantage of special features which may be found in a particular application. The characteristics of general purpose solutions are outlined below.

Least constraints on scene

Other than the constraints given in Section 2.6, which are common to most passive ranging systems, it is desirable to place no additional constraints on the scene observed. It is recognized that some particular applications may lend themselves to the introduction of constraints, which in turn simplify the problem. But for more general solutions, it is *not* desirable to

- Introduce any structured illumination into the scene, since this illumination may be confused with features in the scene;
- Assume that the edges in the scene are perfect black to white transitions or that the bandwidth of scene features is known, since this may not be the case; or
- Require one image to be taken with a pin-hole aperture, since this in turn requires a very sensitive camera, or high scene illumination.

Method independent of defocus operator model

Although the defocus operator has to be modelled, it is desirable that the method of solving the depth from focus problem be *independent* of any particular model. Some researchers have simplified the problem of depth from focus by restricting their solutions to a particular defocus operator. The Gaussian function, advocated by Pentland [118][121][120][117], seems to be the most popular in the literature. But as observed by Subbarao [140] and confirmed by the experiments of the author in Section 7.8, the Gaussian may not be an appropriate model.

Although the defocus operator can be defined theoretically for an ideal optical system as shown in Chapter 3, an actual lens, unless it is of very high quality, will have aberrations which produce a unique defocus operator. Therefore a general method for depth from focus should still work, regardless of which model may be found appropriate for the defocus operator.

Selective control of trade-offs

It is inevitable that the solutions to the problem will involve trade-offs which determine the nature of the solution. To keep the solutions general however, it is desirable that the trade-offs remain explicit and adjustable by the user. The key trade-offs for the depth from focus problem are described in Section 5.5.

Space-variant solution

The most general solution to the depth from focus problem is to model the defocus operator as space-variant

$$i(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(x,y;\xi,\eta)s(\xi,\eta)d\xi d\eta \ , \tag{5.1}$$

where

- $s(x, y)$ is the scene;
- $i(x, y)$ is the acquired image; and
- $h(x, y; \xi, \eta)$ is the defocus operator which is dependent on its global location in the scene, as indicated by the co-ordinates x and y.

The changing shape of the defocus operator is, of course, a direct result of the changing depth throughout the scene.

All the present literature on depth from focus simplifies this problem to a space-invariant formulation, so that within a local region (5.1) becomes

$$i(x, y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(x-\xi, y-\eta)s(\xi, \eta)d\xi d\eta \text{ and} \tag{5.2}$$

$$i(x, y) = h(x, y) \otimes s(x, y) \,. \tag{5.3}$$

This simplification was also followed by the author, since the more general solution is beyond the scope of this research.[1]

5.4.2 Accuracy

It is desirable for the solutions to approach the fundamental limitations of the technique. The fundamental limits are determined by camera resolution, noise in the system, the nature of the scene observed, and the difference in size between two defocus operators.

5.4.3 Minimal Data Required

It is desirable for the solutions to require a minimum amount of data. This usually means that less time is required for data acquisition, and it decreases the probability that something will move or change in the scene, between acquired images.

[1] More is said about the space-variant solution in Section 9.4.4.

5.4.4 Speed

It is a reasonable assumption that most applications desire fast solutions. However, in the course of this research, accuracy will not be explicitly sacrificed for speed.[2]

5.5 KEY PROBLEMS AND TRADE-OFFS

This section will articulate the key problems and trade-offs faced in the depth from focus problem.

5.5.1 Isolating And Modelling The Defocus Operator

The most difficult segment of the depth from focus problem is deconvolving and modelling the defocus operator, or a ratio of defocus operators from the scene. Solving this segment of the problem is the focus of Chapter 6.

5.5.2 Problem Ill-posed

Subbarao [140] states that depth from focus is a well-posed problem, although perhaps ill-conditioned. Hadamard [57] defined a mathematical problem to be well-posed if

- An answer exists;
- The answer is unique; and
- A small error induced into the data does not produce large fluctuations (instability) in the answer.

The depth from focus problem satisfies the first two criterion, but fails miserably on the third. This instability is best illustrated in the spatial frequency domain. Let

[2]This is summarized by the colloquial statement, "First you get good, then you get fast."

the Fourier transforms of the local regions $I_1(f_x, f_y)$ and $I_2(f_x, f_y)$ be written as

$$I_1(f_x, f_y) = S(f_x, f_y) \cdot H_1(f_x, f_y) + N_1(f_x, f_y) \text{ and} \tag{5.4}$$

$$I_2(f_x, f_y) = S(f_x, f_y) \cdot H_2(f_x, f_y) + N_2(f_x, f_y) \ , \tag{5.5}$$

where

- $S(f_x, f_y)$ is the Fourier transform of the scene;
- $H_1(f_x, f_y)$ and $H_2(f_x, f_y)$ are the optical transfer functions of the two defocus operators; and
- $N_1(f_x, f_y)$ and $N_2(f_x, f_y)$ are the Fourier transforms of the noise in the image acquisition process.

Then the ratio of the defocus operators $H_3(f_x, f_y)$, can be found by inverse filtering. That is, the Fourier transforms of the two local regions $I_2(f_x, f_y)$ and $I_1(f_x, f_y)$, are divided to form

$$H_3(f_x, f_y) = \frac{I_2(f_x, f_y)}{I_1(f_x, f_y)} = \frac{S(f_x, f_y) \cdot H_2(f_x, f_y) + N_2(f_x, f_y)}{S(f_x, f_y) \cdot H_1(f_x, f_y) + N_1(f_x, f_y)} \ . \tag{5.6}$$

As shown by (5.6), the noise although usually small, prevents the frequency domain scene $S(f_x, f_y)$ from cancelling in the equation. Therefore, the zero-crossings of $S(f_x, f_y)$ will cause $H_3(f_x, f_y)$ to become unstable. In the spatial domain, (5.6) can be formulated as a matrix equation, whose solution requires inverting a matrix (see Appendix B). Although dependent on the scene, noise will usually cause instability in the inverted matrix.

There are two ways of dealing with the ill-posed nature of the problem.

1. **Smoothing**: Most approaches in the literature employ some form of smoothing or averaging to control the instability of the problem. For example, Pentland [117, §3] multiplies his power estimates by a Gaussian to minimize instability.

2. **Impose Known Shape onto Solution**: The shape of the defocus operator, as a function of depth, can be known *a priori*, so that it can be imposed onto the solution. A recognized approach to solving ill-posed problems of this nature is the *regularization method* of Tikhonov [146][8].

   For example, consider the matrix equation

$$[\mathbf{i_1}] \cdot \mathbf{h_3} = \mathbf{i_2} \ , \tag{5.7}$$

   which is to be solved for $\mathbf{h_3}$, where the solution is known to be one of a family of patterns. The regularized form of (5.7) minimizes the functional

$$\|[\mathbf{i_1}] \cdot \mathbf{h_3} - \mathbf{i_2}\|^2 + \lambda \, \|[\mathbf{C}] \cdot \mathbf{h_3}\|^2 = \text{minimum} \tag{5.8}$$

   where

   - λ is a scalar parameter; and
   - $[\mathbf{C}]$ is a matrix which minimizes the magnitude of the second term if $\mathbf{h_3}$ belongs to the expected family of patterns.

Intuitively, regularization is preferred above smoothing, since the solution space is constrained to a known family of solutions. To the author's knowledge, applying regularization to the depth from focus problem has not been previously reported in the literature.

5.5.3 Size Of Local Region

One of the most critical trade-offs is the size of the local region which is extracted from the images. Since depth changes throughout the scene, it is desirable for this region to be as small as possible, yet accurate identification of the defocus operator is only possible if the area is *larger* than the defocus operator.

The best compromise is probably an iterative approach, where the size of the area considered varies, depending on the estimated size of the defocus operator. To guard against false solutions, the initial area considered would have to be slightly greater than the largest defocus operator expected.

5.6 APPROACHES TO SOLUTIONS

In the literature review of Chapter 4, two main approaches to the depth from focus problems were presented:

- Depth from *automatic focus*; and
- Depth from *focal gradient*.

Since the first approach measures requires a mechanical lens adjustment for every depth value, the second approach, which requires only one optical parameter adjustment for the whole scene, has a greater speed potential. Therefore, based on the desired criteria of Section 5.4.4, the first approach will not be considered and depth from *focal gradient* will be analyzed.

As indicated in Section 4.4.1, this implies that at least two images of the scene will need to be acquired, with two different defocus operators. Local regions from these two images will be selected, and then either one, or a ratio of the two operators must be isolated.[3] There are at least two different methods for isolating the defocus operator:[4]

1. Inverse filtering; and
2. A matrix based approach.

[3]Either approach is valid since Subbarao [140], explains that the two operators are necessarily linked by known camera parameters.

[4]Other methods of deconvolution exist [7][133], but were not actively considered. The methods presented, were chosen because they have the potential of working on very small local areas, which is critical to the depth from focus problem.

5.6.1 Inverse Filtering

From a theoretical stand-point, inverse filtering in the frequency domain is elegant, because convolution in the spatial domain now becomes multiplication [50, §5.4]. Therefore, at first glance, the ratio of the defocus operators $H_3(f_x, f_y)$, can be found simply by dividing the Fourier transforms of the two local regions in the images $I_2(f_x, f_y)$ and $I_1(f_x, f_y)$,

$$H_3(f_x, f_y) = \frac{I_2(f_x, f_y)}{I_1(f_x, f_y)} = \frac{S(f_x, f_y) \cdot H_2(f_x, f_y)}{S(f_x, f_y) \cdot H_1(f_x, f_y)}, \tag{5.9}$$

where

- $S(f_x, f_y)$ is the Fourier transform of the scene; and
- $H_1(f_x, f_y)$ and $H_2(f_x, f_y)$ are the optical transfer functions of the two defocus operators.

However, there are several fundamental problems with inverse filtering, which will be detailed in Section 6.5.2.

5.6.2 Matrix Based Solutions

Solving the depth from focus problem by matrix analysis in the spatial domain avoids the fundamental problems associated with inverse filtering, however the processing requirements are often greater. The matrix based approach seeks to deconvolve the defocus operator from the images by characterizing the problem as a large system of linear equations, similar to the superposition method of Backus and Gilbert [5], and Frieden [43]. The exact nature of matrix based solutions and their justification is presented in the next chapter.

5.7 CONCLUSIONS

This chapter has defined the problem of depth from focus, and outlined some of the desired characteristics of general purpose solutions to the problem. After the key problems and trade-offs were examined, a general approach to the problem was proposed, motivated by the work of other researchers.

This approach will be to acquire at least two images of the scene, blurred with two different defocus operators. Local regions from these two images will be selected, and then either one, or a ratio of the two operators must be isolated. Two methods of isolating the defocus operator are

1. Inverse filtering; and

2. A matrix based approach.

Chapter 6

THEORETICAL BASIS FOR NEW METHOD

6.1 INTRODUCTION

This chapter builds on the problem analysis of Chapter 5 to present several solutions to the depth from focus problem. The theoretical justifications for the solutions are constructed incrementally, with a number of simplifying assumptions introduced initially, and gradually relaxed to build up the solutions. The author's new matrix solution is presented and theoretically justified. The implementation of this solution is the subject of Chapter 7.

6.2 OVERVIEW

Section 6.3 articulates the computational goal in solving the problem of depth from focus. A defocus operator $h_3(x, y)$ called a *convolution ratio* is introduced.

In Section 6.4 it is assumed that no noise is present in the acquired images. A formula for $h_3(x, y)$ is derived and shown to be a unique indicator of depth. Continuing with the assumption of noise and allowing only one image to be blurred, Section 6.5 introduces the author's new matrix based solution to the depth from focus problem and contrasts it to the traditional approach of inverse filtering. Even under the simplified assumptions, it will be shown that inverse filtering in the frequency domain has two serious problems:

- Difficulties in accurately finding the frequency domain representation and distortions caused by windowing; and
- Presence of border effects.

It will also be shown that these problems do not occur with the proposed matrix based approach, and that the defocus operator can be exactly recovered.

In Section 6.6, one assumption is relaxed and the second image can now be defocused as well. Finally, Section 6.7 re-introduces noise into the model and explains that a regularized approach should be taken. A closed form matrix based solution in the spatial domain is given, as well as a more general, iterative solution. The inverse filtering approach is also extended to a constrained inverse filtering solution.

Although the solutions will be given in two dimensions, some of the formulas and illustrations may be presented only in one dimension for simplicity. In these cases, extensions to two dimensions are straightforward.

6.3 COMPUTATIONAL GOAL IS TO FIND $h_3(x, y)$

The solutions presented in this chapter are based on Figure 6.1. Two images i_1 and i_2 of the same scene are acquired with two different defocus operators. A local region of the scene will be called $s(x, y)$. If $h_1(x, y)$ is the defocus point spread function of the smaller defocus operator, and $n_1(x, y)$ is the noise in the image acquisition process, then $i_1(x, y)$, a local region of the acquired image can be written as

$$i_1(x, y) = s(x, y) \; [\otimes] \; h_1(x, y) + n_1(x, y) \; . \tag{6.1}$$

The operator $[\otimes]$ designates restricted convolution, where the borders of the kernel $h_1(x, y)$ are *not* convolved past the borders of the scene $s(x, y)$. This restricted definition is necessary, because the result of convolving the kernel past the ends of the local

region of the image is *unknown*.

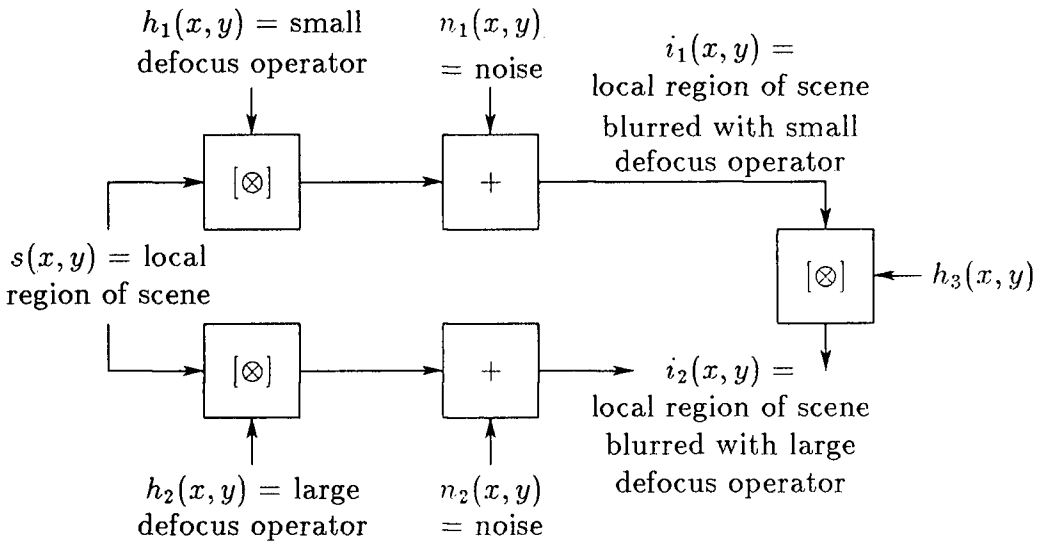


Figure 6.1: Block diagram of the depth from focus problem. The computational goal is to find $h_3(x,y)$.

Similarly, if $h_2(x,y)$ is the defocus point spread function of the large defocus operator, $n_2(x,y)$ is the added noise, and $i_2(x,y)$ is a local region of the image blurred with this operator, then

$$i_2(x,y) = s(x,y)\ [\otimes]\ h_2(x,y) + n_2(x,y)\ . \tag{6.2}$$

It will be assumed that the noise, $n_1(x,y)$ and $n_2(x,y)$:

- Has a zero mean;
- Is additive to the signal; and
- Has a Gaussian distribution.

The computational goal will be to find a new point spread function $h_3(x, y)$, which will transform $i_1(x, y)$ into $i_2(x, y)$ according to

$$i_1(x, y) \; [\otimes] \; h_3(x, y) = i_2(x, y) \; . \tag{6.3}$$

Various solutions for $h_3(x, y)$ are developed in the following sections.

6.4 NO NOISE IN IMAGES

To understand how $h_3(x, y)$ can be obtained, it is necessary to first make some simplifying assumptions, which will be relaxed later. Initially it will be assumed that there is no noise in the acquired images. That is

$$n_1(x, y) = n_2(x, y) = 0 \tag{6.4}$$

Based on this assumption, a formula for $h_3(x, y)$ is easy to derive.

6.4.1 A Formula For $h_3(x, y)$

This section will derive a formula for $h_3(x, y)$ assuming there is no noise in the acquired images. Then (6.1) becomes

$$i_1(x, y) = s(x, y) \; [\otimes] \; h_1(x, y) \; , \tag{6.5}$$

and (6.2) becomes

$$i_2(x, y) = s(x, y) \; [\otimes] \; h_2(x, y) \; . \tag{6.6}$$

Substituting these equations into (6.3) produces

$$\begin{aligned} s(x, y) \; [\otimes] \; h_1(x, y) \; [\otimes] \; h_3(x, y) &= s(x, y) \; [\otimes] \; h_2(x, y) \;\; \text{and} \\ h_1(x, y) \; [\otimes] \; h_3(x, y) &= h_2(x, y) \; . \end{aligned} \tag{6.7}$$

Since $h_1(x,y)$ has a limited spatial extent, (6.7) can be generalized to

$$h_1(x,y) \otimes h_3(x,y) = h_2(x,y) \ . \tag{6.8}$$

In (6.8), $h_3(x,y)$ will be called the *convolution ratio* of the two defocus operators $h_2(x,y)$ and $h_1(x,y)$.

6.4.2 Uniqueness Of $h_3(x,y)$ For Geometric Optics

Since Chapter 3 has shown that a defocus operator is a unique indicator of depth, then $h_1(x,y)$ and $h_2(x,y)$ correspond uniquely to scene depth. Although it is not possible to prove that $h_3(x,y)$ corresponds uniquely to depth for any arbitrary defocus operator,[1] it is possible to show this uniqueness assuming geometric optics. The uniqueness of $h_3(x,y)$ is best illustrated by showing the uniqueness of $H_3(f_x, f_y)$, its Fourier transform. Deriving $h_3(x,y)$ in the spatial domain is a little more difficult, and is deferred to the next chapter.

Using (3.22) and (6.8),

$$H_3(\rho) = \frac{R_1 \rho J_1(2\pi R_2 \rho)}{R_2 \rho J_1(2\pi R_1 \rho)} \ . \tag{6.9}$$

Then, substituting in (3.24)

$$H_3(\rho_p) = \frac{R_1 \rho_p J_1\left(\frac{2\pi R_2 \rho_p P}{N}\right)}{R_2 \rho_p J_1\left(\frac{2\pi R_1 \rho_p P}{N}\right)} \ . \tag{6.10}$$

It can also be stated from (3.15) that

$$R_1 = \frac{FD_i + FD_o - D_i D_o}{2 f_1 D_o} \quad \text{and} \tag{6.11}$$

$$R_2 = \frac{FD_i + FD_o - D_i D_o}{2 f_2 D_o} \ . \tag{6.12}$$

[1] Although by showing this is true for geometric optics, it will be assumed to be true for defocus in general.

Finally, substituting (6.11) and (6.12) into (6.10) yields

$$H_3(\rho_p) = \frac{\rho_p f_2 J_1\left(2\pi \frac{FD_i + FD_o - D_i D_o}{N 2 f_2 D_o} \rho_p P\right)}{\rho_p f_1 J_1\left(2\pi \frac{FD_i + FD_o - D_i D_o}{N 2 f_1 D_o} \rho_p P\right)} . \tag{6.13}$$

Since F, D_i, N, P, f_1 and f_2 are usually fixed, then (6.13) shows that $H_3(\rho_p)$ is a unique function of the distance D_o. In Figure 6.2, using fixed parameters $F = 50\ mm$, $D_i = 52.63\ mm$, $f_1 = 2.0$, $f_2 = 1.3$, $N = 64$, and $P = 60.0$ pixels/mm, the shape of $H_3(\rho_p)$ is graphed using (6.13) for distances $D_o = 0.90$ and 0.95 *meters*. The unique characteristic of $H_3(\rho_p)$ is the down-turned quadratic-type shape, centered at $\rho_p = 0$. The width of this quadratic varies uniquely with distance. This relationship will be formulated in Section 6.7.3.

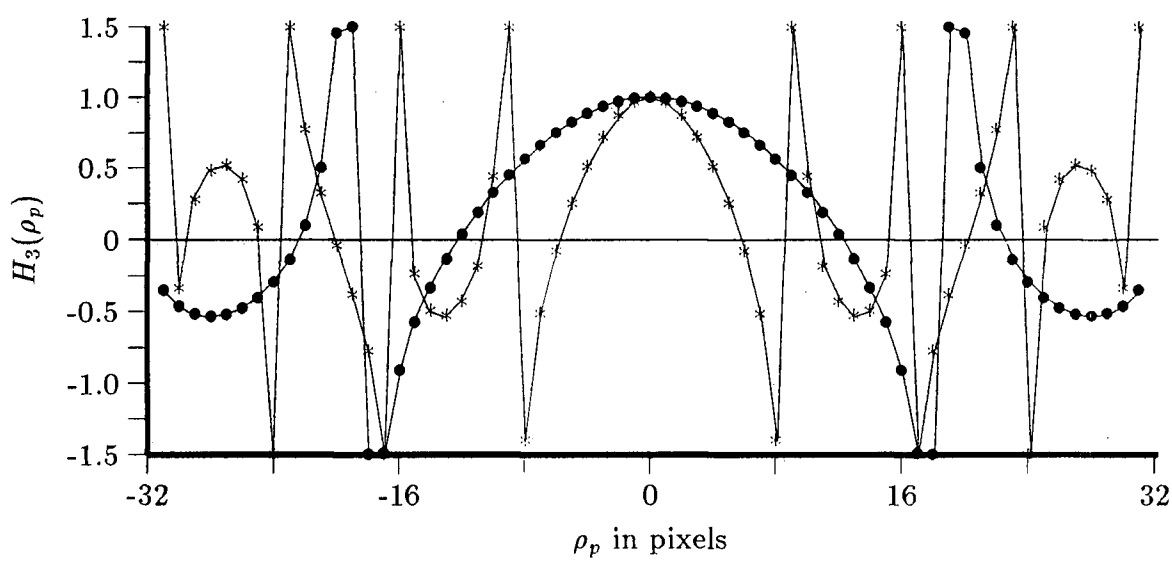


Figure 6.2: With the fixed parameters given in the text, $H_3(\rho_p)$ is shown graphed for distances $D_o = 0.90$ (*) and 0.95 (•) *meters*.

6.5 ONE BLURRED IMAGE WITHOUT NOISE

To develop solutions for $h_3(x, y)$, it is necessary to first make some simplifying assumptions, which will be relaxed later. Initially it will be assumed that:

1. i_1 is acquired with a pinhole aperture ($h_1(x, y)$ is an impulse function);
2. There is no noise ($n_1(x, y) = n_2(x, y) = 0$);
3. The spatial extent of $h_3(x, y)$ is less than, or equal to N points; and

Based on these assumptions, $i_1(x, y) = s(x, y)$ and $h_2(x, y) = h_3(x, y)$. Then (6.3) becomes

$$s(x, y) \; [\otimes] \; h_3(x, y) = i_2(x, y) \; . \tag{6.14}$$

There are at least two main approaches to the goal of solving (6.14) for $h_3(x, y)$:

1. Inverse filtering approach; and
2. Matrix based approach.

It will be argued that the matrix based approach has an accuracy greater than or equal to the inverse filtering approach, and that for most scenes, it is more accurate.

This can be shown in two ways.

1. Appendix B contains a mathematical proof, that in one dimension, the exact recovery of a kernel $h(x)$ where

$$s(x) \; [\otimes] \; h(x) = i(x) \; , \tag{6.15}$$

can be performed for a larger family of signals $s(x)$ and $i(x)$ using the matrix based approach in the spatial domain, than the inverse filtering approach in the spatial frequency domain. Unfortunately, the proof yields little intuitive insight into *why* errors are incurred using inverse filtering, and their corresponding symptoms.

2. This section will give an analysis of the key sources of error in the inverse filtering approach, and how these are eliminated in the matrix based approach.

6.5.1 Matrix Solution For $h_3(x, y)$

A convolution in the spatial domain can be exactly represented by a series of matrix equations. The details are given in Appendix B, but the restricted convolution equation (6.3) in one dimension, can be written as the matrix equation

$$[\mathbf{i_{1T}}] \cdot \mathbf{h_3} = \mathbf{i_2} \ , \tag{6.16}$$

where $[\mathbf{i_{1T}}]$ is a Toeplitz matrix formed from $i_1(x)$ according to (B.37), $\mathbf{h_3}$ is a vector formed from $h_3(x)$, and $\mathbf{i_2}$ is a vector formed from $i_2(x)$.

Equation (6.16) can be solved for $\mathbf{h_3}$ as

$$\mathbf{h_3} = [\mathbf{i_{1T}}]^{-1} \cdot \mathbf{i_2} \ . \tag{6.17}$$

To invert $[\mathbf{i_{1T}}]$ in (6.17), there are traditional methods by Levinson [100][99][123, §2.8] and Trench [148] which take advantage of the persymmetry of the Toeplitz matrix. A recent method by Yarlagadda and Babu treats the Toeplitz matrix as a submatrix of a circulant matrix [155]. For two dimensional images, it would be necessary to invert a Toeplitz-block Toeplitz matrix. Extensions have been developed for the two dimensional case [3][78][151], however, even for a local image region of 64×64 pixels, this would involve solving a linear equation with 4096 variables!

Therefore, there is an incentive to try to reduce the computational requirements by performing the deconvolution with inverse filtering in the frequency domain. However, it will be shown that this cannot generally be done without incurring errors.

6.5.2 Inverse Filtering Solution For $h_3(x, y)$

From a theoretical stand-point, the inverse filtering solution in the frequency domain is convenient, because convolution in the spatial domain now becomes multiplication. To enable transformation to the frequency domain, (6.1) is windowed [59] with $w_1(x, y)$,[2]

$$i_1(x, y) \cdot w_1(x, y) = [s(x, y) \; [\otimes] \; h_1(x, y)] \cdot w_1(x, y) \; . \tag{6.18}$$

Taking the Fourier transform of both sides,

$$I_1(f_x, f_y) \otimes W_1(f_x, f_y) = [S(f_x, f_y) \cdot H_1(f_x, f_y)] \otimes W_1(f_x, f_y) \; , \tag{6.19}$$

where the following are Fourier pairs:

$$i_1(x, y) \iff I_1(f_x, f_y) \; , \tag{6.20}$$

$$h_1(x, y) \iff H_1(f_x, f_y) \; , \tag{6.21}$$

$$w_1(x, y) \iff W_1(f_x, f_y) \text{ and} \tag{6.22}$$

$$s(x, y) \iff S(f_x, f_y) \; . \tag{6.23}$$

Similarly for $i_2(x, y)$,

$$I_2(f_x, f_y) \otimes W_2(f_x, f_y) = [S(f_x, f_y) \cdot H_2(f_x, f_y)] \otimes W_2(f_x, f_y) \; . \tag{6.24}$$

To isolate the defocus ratio operator from the scene, (6.24) is divided by (6.19)

$$\frac{I_2(f_x, f_y) \otimes W_2(f_x, f_y)}{I_1(f_x, f_y) \otimes W_1(f_x, f_y)} = \frac{(S(f_x, f_y) \cdot H_2(f_x, f_y)) \otimes W_2(f_x, f_y)}{(S(f_x, f_y) \cdot H_1(f_x, f_y)) \otimes W_1(f_x, f_y)} \; . \tag{6.25}$$

It is generally assumed that $W_1(f_x, f_y)$ and $W_2(f_x, f_y)$ have a small spread in the frequency domain,[3]

$$\frac{I_2(f_x, f_y)}{I_1(f_x, f_y)} \approx \frac{S(f_x, f_y) \cdot H_2(f_x, f_y)}{S(f_x, f_y) \cdot H_1(f_x, f_y)} \text{ and} \tag{6.26}$$

$$\frac{I_2(f_x, f_y)}{I_1(f_x, f_y)} \approx \frac{H_2(f_x, f_y)}{H_1(f_x, f_y)} = H_3(f_x, f_y) \; . \tag{6.27}$$

[2] Recall that the noise $n_1(x, y)$ is assumed to be zero for now.

[3] Although this causes problems which are described in Section 6.5.3.

The defocus ratio function $H_3(f_x, f_y)$, the Fourier transform of $h_3(x, y)$, has been isolated by inverse filtering. From it, depth can be inferred.

6.5.3 Problems With Inverse Filtering

Inverse filtering by division in the frequency domain is fraught with two key problems.

Accurate conversion to frequency domain and window effects

Difficulties arise in accurately estimating the frequency domain representation $I_1(f_x, f_y)$ and $I_2(f_x, f_y)$ of local regions $i_1(x, y)$ and $i_2(x, y)$. An important constraint, is that since depth changes throughout the scene, it is desirable for these regions to be as small as possible, yet accurate frequency domain analysis benefits from larger local areas.

Frequency domain conversion and spectral analysis is an active field of research and many techniques have been proposed in the literature [105]. The conventional method of finding the frequency domain representation is by computing the Fourier transform of a windowed portion from the spatial domain. All researchers working on the depth from focus problem in the frequency domain have advocated the Fourier transform.

One alternate form of frequency estimation, the Wigner Distribution [152][30][31][32], was attempted by the author with little success. The Wigner Distribution is noted for its ability to provide a better resolution than the FFT [46, §3.2]. Unfortunately this better resolution comes at the expense of cross-terms. The presence of these cross-terms occluded the answer when the two frequency domain images were divided according to (6.26).

Although windowing is necessary to calculate the Fourier transform of a local region [59], it causes other residual problems. Looking at (6.26), $I_2(f_x, f_y)$ contains two types of zero crossings:

1. Those that arise from the scene $S(f_x, f_y)$; and

2. Those that arise from $H_2(f_x, f_y)$.

The result for $H_3(f_x, f_y)$ must be independent of the scene $S(f_x, f_y)$, since $H_1(f_x, f_y)$ and $H_2(f_x, f_y)$ are independent of the scene. Therefore, the zero crossings in $I_2(f_x, f_y)$ that are due to $S(f_x, f_y)$ must be matched by zero crossings in $I_1(f_x, f_y)$, so that they can cancel.[4] Unfortunately, the result of convolving $I_2(f_x, f_y)$ with $W_2(f_x, f_y)$ and $I_1(f_x, f_y)$ with $W_1(f_x, f_y)$ is that the zero crossings shift slightly. This small movement can cause large variations in the quotient $H_3(f_x, f_y)$.

Since the errors incurred in inverse filtering due to windowing are dependent on a number of factors, including the scene $s(x, y)$ and the windowing functions $w_1(x, y)$ and $w_2(x, y)$, it is difficult to provide a closed form equation. But by making some reasonable assumptions, an example of some of the errors incurred can be shown.

Dealing in one dimension for simplicity, it will be assumed that one defocus operator $h_1(x)$, is an impulse function and hence from (6.1), $i_1 = s$, since the noise is assumed to be zero for now. Let the scene be a simple black to white transition,

$$s = i_1 = (0, \ldots, 0, 0, 1, 1, 1, \ldots, 1) \ . \tag{6.28}$$

It will also be assumed that the other blurring operator is a gate function,

$$h_2(x) = (1, 1, 1) \ . \tag{6.29}$$

Also, using (6.8),

$$h_3(x) = h_2(x) = (1, 1, 1) \ . \tag{6.30}$$

The second blurred image can be calculated using (6.2),

$$i_2 = (0, \ldots, 0, 1, 2, 3, 3, \ldots, 3) \ . \tag{6.31}$$

[4] The zero crossings caused by $S(f_x, f_y)$ that may match the zero crossings of $H_2(f_x, f_y)$ are, of course, excluded.

Using the Gaussian window,[5]

$$w_1(x) = w_2(x) = \frac{1}{\sqrt{2\pi}\,\sigma}\; e^{-\frac{x^2}{2\sigma^2}} \; . \tag{6.32}$$

Working in the frequency domain, if $\mathcal{F}^{-1}$ is the inverse Fourier operator, it can be deduced from (6.25) and (6.27) that

$$\hat{h}_3(x) = \mathcal{F}^{-1}\left(\frac{W_2(f_x) \otimes I_2(f_x)}{W_1(f_x) \otimes I_1(f_x)}\right) \; , \tag{6.33}$$

where $\hat{h}_3(x)$ is the estimated pattern for $h_3(x)$. The error ϵ_f, caused by calculating $\hat{h}_3(x)$ by inverse filtering, can be written as

$$\epsilon_f = \sum_{x=0}^{N-1} \left[h_3(x) - \hat{h}_3(x)\right]^2 \; , \tag{6.34}$$

$$\epsilon_f = \sum_{x=0}^{N-1} \left[h_3(x) - \mathcal{F}^{-1}\left(\frac{W_2(f_x) \otimes I_2(f_x)}{W_1(f_x) \otimes I_1(f_x)}\right)\right]^2 \; . \tag{6.35}$$

For various sizes N, of data used, the RMS errors calculated from (6.35), are shown in Table 6.1. The tables shows that the error decreases as N increases, however, to

N	% Error
4	65.8
8	23.7
16	6.4
32	1.1

Table 6.1: Errors which result from windowing a blurred black to white transition in the spatial domain, before inverse filtering by division in the frequency domain.

drive the error down to 1%, it is necessary to observe a length that is about an *order of magnitude* larger than the blurring operator!

[5]The Gaussian is used, because as shown by Gabor [44], it is simultaneously, optimally localized in both the spatial and frequency domains.

Using a matrix based approach, this example can be solved *exactly*, using dimensions as small as $N = 3$. A Toeplitz matrix $[\mathbf{i_{1T}}]$, is constructed by inverting i_1 given by (6.28), and shifting sections into each row, according to (B.37),

$$[\mathbf{i_{1T}}] = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} . \tag{6.36}$$

To confirm that $\mathbf{h_3}$ can be calculated *exactly* by matrix manipulation, (6.36) and (6.31) can be substituted into the one-dimensional form of (6.16) and solved for $\mathbf{h_3}$,

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} \cdot \mathbf{h_3} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} , \tag{6.37}$$

$$\mathbf{h_3} = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} , \text{ and} \tag{6.38}$$

$$\mathbf{h_3} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} . \tag{6.39}$$

This example has illustrated the errors caused by windowing a scene in the spatial domain, before inverse filtering in the frequency domain. The correct solution was obtained exactly by a matrix based solution.

Border effects

Another problem with inverse filtering by frequency domain division, is selecting the most appropriate spatial area for analysis. According to conventional inverse filtering,

the *same* sizes of $i_1(x,y)$ and $i_2(x,y)$ are used to calculate $I_1(f_x,f_y)$ and $I_2(f_x,f_y)$, so that in (6.26) frequency components of the same value can be divided. As shown by Figure 6.3, using the same size $N \times N$ of $i_1(x,y)$ and $i_2(x,y)$, infers that *different* dimensions $(N+N_1-1)^2$ and $(N+N_2-1)^2$ of the scene $s(x,y)$ are used. Therefore $i_2(x,y)$ may contain spurious data from bordering edges.

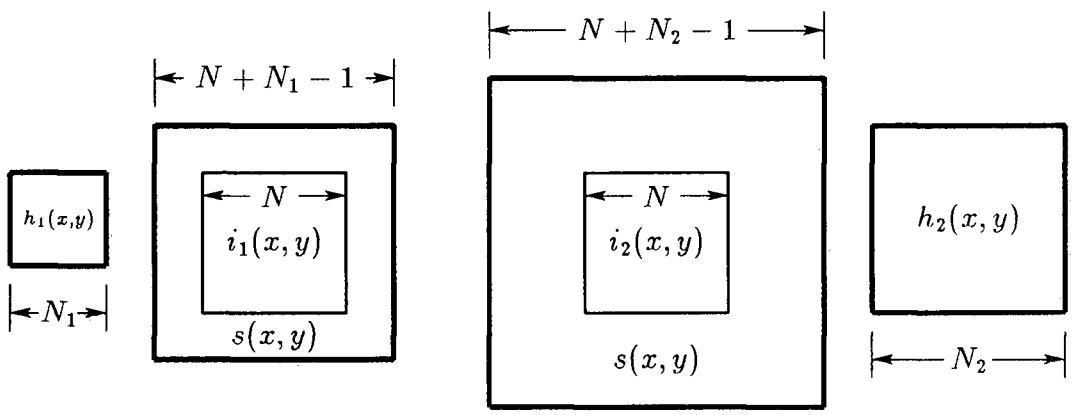


Figure 6.3: Differences in scene area $s(x,y)$, when the same image areas for $i_1(x,y)$ and $i_1(x,y)$ are used. The dimensions of $i_1(x,y) = s(x,y)\ [\otimes]\ h_1(x,y)$ and $i_2(x,y) = s(x,y)\ [\otimes]\ h_2(x,y)$ are illustrated.

Pentland briefly refers to the problem of border effects [120] and offers Gaussian windowing as the solution. Since windowing has other problems associated with it, as shown in the last section, perhaps depth from focus analysis by inverse filtering could be improved, if $I_1(f_x,f_y)$ and $I_2(f_x,f_y)$ were the Fourier transforms of different dimensional areas.[6] Since the Fourier transforms now have different discrete component separation, a common set would have to be interpolated, and substituted into (6.26). However, the problems associated with accurately estimating the frequency domain images and windowing, still remain. This method is mentioned again in Section 9.4.2.

[6] A caveat to this technique is that N_1 and N_2 are not known exactly until the problem is solved. This suggests an iterative approach.

To illustrate the border effect problems, a one-dimensional example will be presented. It will be assumed that one defocus operator $h_1(x)$, is an impulse function, and hence from (6.1), $i_1 = s$, since the noise is assumed to be zero for now. Let the scene be

$$s = i_1 = (1, \ldots, 1, 1, 0, 0, 0, 1, 3, 1, 0, 0, 0, 1, 1, \ldots, 1) \ . \tag{6.40}$$

Again, it will also be assumed that the other blurring operator is a gate function,

$$h_2(x) = (0, 0, 0, 1, 1, 1, 0, 0) \ . \tag{6.41}$$

Also, Using (6.2) and (6.3),

$$h_3(x) = h_2(x) = (0, 0, 0, 1, 1, 1, 0, 0) \ . \tag{6.42}$$

The second blurred image can be calculated using (6.2),

$$i_2 = (3, \ldots, 3, 2, 1, 0, 1, 4, 5, 4, 1, 0, 1, 2, 3, \ldots, 3) \ . \tag{6.43}$$

A local, symmetric region of eight points of $i_1(x)$ and $i_2(x)$ will be selected,[7] so that the Fourier transform is wholly real.[8] Then

$$i_1(x) = (0, 0, 0, 1, 3, 1, 0, 0) \text{ and} \tag{6.44}$$

$$i_2(x) = (1, 0, 1, 4, 5, 4, 1, 0) \ . \tag{6.45}$$

If the discrete Fourier transforms of $i_1(x)$ and $i_2(x)$ are $I_1(f_x)$ and $I_2(f_x)$ respectively, then

$$I_2(f_x) = (0.000, -1.657, 4.000, 9.657, 16.000, 9.657, 4.000, -1.657) \text{ and} \tag{6.46}$$

$$I_1(f_x) = (1.000, 1.586, 3.000, 4.414, 5.000, 4.414, 3.000, 1.586) \ . \tag{6.47}$$

[7] It is assumed that the kernel $h_3(x)$ is known to be less than or equal to eight points long.

[8] According to usual conventions, symmetry is about the point to the right of the middle of the sequence. The Fourier transforms will be shifted so that the zero frequency is in the middle.

If the two frequency domain signals are divided similar to (6.33) then

$$\hat{h}_3(x) = \mathcal{F}^{-1}\left(\frac{I_2(f_x)}{I_1(f_x)}\right) , \tag{6.48}$$

where $\hat{h}_3(x)$ is the estimated pattern for $h_3(x)$. Using (6.48),

$$\hat{h}_3(x) = (0.448, -0.171, 0.067, 0.971, 1.019, 0.971, 0.067, -0.171) . \tag{6.49}$$

The result is close, but not equal to the original pattern of

$$h_3(x) = (0, 0, 0, 1, 1, 1, 0, 0) . \tag{6.50}$$

Using the matrix solution, where $N = 8$ to parallel the inverse filtering case, the convolution can be represented by (6.16)

$$[\mathbf{i_{1T}}] \cdot \mathbf{h_3} = \mathbf{i_2} , \tag{6.51}$$

where

$$[\mathbf{i_{1T}}] = \begin{bmatrix} 3 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 3 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 3 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 3 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 3 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 3 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 3 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 3 \end{bmatrix} \quad \text{and} \tag{6.52}$$

$$\mathbf{i_2} = \begin{bmatrix} 1 & 0 & 1 & 4 & 5 & 4 & 1 & 0 \end{bmatrix}^T . \tag{6.53}$$

Then solving (6.51) by inverting $[\mathbf{i_{1T}}]$, yields

$$\mathbf{h_3} = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}^T , \tag{6.54}$$

which is the expected result. Since the convolution kernel $h_3(x)$, is essentially only three points long, it could also have been recovered for $N = 3$.

This example has shown that using inverse filtering, border effects can cause significant distortions to the recovery of a convolution kernel, *throughout* the extent of the kernel. This problem is caused by the use of *equal* spatial areas for the scene $i_1(x)$, and the blurred image $i_2(x)$. With the matrix based approach, border effects are taken into account since *unequal* spatial areas of $i_1(x)$ and $i_2(x)$ are considered. Therefore, in the absence of noise, a kernel can be recovered exactly.[9]

6.6 TWO BLURRED IMAGES WITHOUT NOISE

One of the constraints imposed at the beginning of the chapter will now be relaxed, and i_1 can be acquired with any defocus operator, as long as the spread of $h_1(x, y)$ is less than the spread of $h_2(x, y)$.[10] Although this does not affect the form of the matrix based or inverse filtering solutions, it changes the nature of $h_3(x, y)$ considerably. Under the previous assumption, $h_3(x, y)$ had a limited spatial support, but now it will be shown that even if both $h_1(x, y)$ and $h_2(x, y)$ have a finite spatial extent, $h_3(x, y)$ can have an infinite extent.

Therefore, a windowing operation will have to be performed on $h_3(x, y)$, so that it can be represented within a local spatial region of less than N points. Although Section 6.5.2 was critical of windowing $i_1(x, y)$ and $i_2(x, y)$ in order to perform inverse filtering in the frequency domain, it will be shown that windowing $h_3(x, y)$ has less serious implications.

If $w_3(x, y)$ is the window applied to $h_3(x, y)$, then the error caused by windowing in

[9] That is, assuming the scene has sufficient features to allow the effect of the kernel to be seen.
[10] This was an original condition, noted in Figure 6.1.

the spatial domain ϵ_m, is expressed simply as

$$\epsilon_m = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\left[h_3(x,y) - w_3(x,y)\cdot h_3(x,y)\right]^2 . \tag{6.55}$$

The contributions to ϵ_m come primarily from the truncated tails, which are generally small.

This can be compared to (6.35) in two dimensions, the error caused by windowing for inverse filtering in the frequency domain,

$$\epsilon_i = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\left[h_3(x,y) - \mathcal{F}^{-1}\left(\frac{W_2(f_x,f_y)\otimes I_2(f_x,f_y)}{W_1(f_x,f_y)\otimes I_1(f_x,f_y)}\right)\right]^2 . \tag{6.56}$$

In Section 6.5.3, the effect that windowing $i_1(x,y)$ and $i_2(x,y)$ had on $H_3(f_x,f_y)$ was discussed. From this it can be expected that contributions to ϵ_i can occur throughout the extent of $h_3(x,y)$. Although dependent on the nature of $i_1(x,y)$ and $i_2(x,y)$, as well as the windows employed, it can be generally expected that ϵ_i will be larger than ϵ_m.[11]

To demonstrate that ϵ_i is generally larger than ϵ_m, the following one-dimensional example can be considered. Let

$$h_1(x) = (0,\ldots,0,1,1,1,0,\ldots,0) , \tag{6.57}$$

$$h_2(x) = (0,\ldots,0,1,1,1,1,1,0,\ldots,0) \text{ and} \tag{6.58}$$

$$s = (0,\ldots,0,0,0,1,1,1,\ldots,1) . \tag{6.59}$$

Now the local images can be calculated

$$i_1(x) = s(x)\,[\otimes]\,h_1(x) \text{ and} \tag{6.60}$$

$$i_2(x) = s(x)\,[\otimes]\,h_2(x) , \tag{6.61}$$

[11] It is assumed that $i_1(x,y)$ and $i_2(x,y)$ are windowed with non-adaptive windows (ie. Gaussian, box-car, etc.). If adaptive windows are used, based on $i_1(x,y)$ and $i_2(x,y)$ so that the integrity of the zero crossings discussed in Section 6.5.3 are maintained, then conceivably ϵ_i can equal ϵ_m. The form of these adaptive windows, is however not known, and may be very difficult to calculate.

and the Toeplitz matrix $[\mathbf{i_{1T}}]$ can be formed according to (B.40). This example is rather extreme, since $h_3(x)$ has an infinite extent. This is caused by the sharp edges on $h_2(x)$. If $h_2(x)$ had more rounded edges, then the spatial extent of $h_3(x)$ would decrease. Recall that, by definition, $h_2(x)$ is wider than $h_1(x)$.

To calculate the error, a comparatively large section of $h_3(x)$ was calculated using (6.17) with a matrix size of 64. Let the windowing functions be a Gaussian for $w_1(x)$ and $w_2(x)$, and a box-car for $w_3(x)$ of width N,

$$w_1(x) = w_2(x) = \frac{1}{\sqrt{2\pi}\,\sigma}\, e^{-\frac{x^2}{2\sigma^2}} \text{ and} \tag{6.62}$$

$$w_3(x) = (1, 1, \ldots, 1, 1)\,. \tag{6.63}$$

Equations (6.55) and (6.56) were used to calculate ϵ_m and ϵ_i respectively. The results are shown in Table 6.2 for various matrix sizes N. This shows that even for this extreme example, the errors caused by windowing $h_3(x)$ in the spatial domain ϵ_m, are still less severe than the errors ϵ_i, caused by windowing $i_1(x)$ and $i_2(x)$.

N	% Error	
	ϵ_i	ϵ_m
4	60.0	55.9
8	60.4	54.5
16	55.8	50.0
32	49.0	41.4

Table 6.2: Comparison of windowing errors ϵ_i and ϵ_m.

6.7 TWO BLURRED IMAGES WITH NOISE

In the absence of noise, and with one blurred image, it has been shown that the matrix based deconvolution of segments from image segments, can be performed exactly in the spatial domain. With two blurred images, only a windowed portion of $h_3(x, y)$ can

be recovered. These operations can be performed more accurately by using a matrix based approach, than by inverse filtering. When noise is added to the model, solving (6.16) for $h_3(x, y)$ becomes very unstable. This section augments the inverse filtering and matrix based solutions, to deal with noise in the image acquisition process.

6.7.1 Closed Form Matrix Solution

Because the depth from focus problem is unstable,[12] it is advisable to apply *all* the information available to the problem. An important, unused piece of information, is that the defocus operators $h_1(x, y)$ and $h_2(x, y)$ have shapes which can be theoretically or experimentally derived. Referring to (6.8), it can be seen that although $h_3(x, y)$ is *unknown*, it must belong to a family of patterns, which can be known *a priori*.[13] A recognized approach to solving ill-posed problems of this nature is the *regularization method* of Tikhonov [146][8].

The regularized form of (6.16) minimizes the functional,

$$\|[\mathbf{i_{1T}}] \cdot \mathbf{h_3} - \mathbf{i_2}\|^2 + \lambda \, \|[\mathbf{C}] \cdot \mathbf{h_3}\|^2 = \text{minimum} \, , \tag{6.64}$$

where

- λ is a scalar parameter; and
- $[\mathbf{C}]$ is a matrix which minimizes the magnitude of the second term if $\mathbf{h_3}$ belongs to the expected family of patterns.

The Euler equation [8] for (6.64) is solved for $\mathbf{h_3}$ as follows

$$\begin{aligned} [\mathbf{i_{1T}}]^T [\mathbf{i_{1T}}] \, \mathbf{h_3} - [\mathbf{i_{1T}}]^T \, \mathbf{i_2} + \lambda \, [\mathbf{C}]^T [\mathbf{C}] \, \mathbf{h_3} &= 0 \, , \quad \text{and} \\ \mathbf{h_3} &= \left([\mathbf{i_{1T}}]^T [\mathbf{i_{1T}}] + \lambda \, [\mathbf{C}]^T [\mathbf{C}] \right)^{-1} [\mathbf{i_{1T}}]^T \, \mathbf{i_2} \, . \end{aligned} \tag{6.65}$$

[12] Small changes in the data $i_1(x)$ due to noise, can produce large fluctuations in the matrix $[\mathbf{i_T}]^{-1}$ used in (6.17).

[13] Although this is a difficult step which is explained in detail later in Section 7.3.2.

In practice, solving (6.65) is computationally expensive. Additionally, finding [C] is difficult for anything other than simple parametric families (ie. Gaussian, quadratic, cubic) of $h_3(x, y)$. Since $h_3(x, y)$ can be quite complex (see Section 7.3.2), and not easily coerced into a parametric shape, it is not generally possible to find [C]. Therefore, in a generalized implementation, the closed form (6.65) cannot be used. An alternative approach is to introduce an iterative technique.

6.7.2 Iterative Matrix Solution

Using (6.8), a table of $h_3(x, y)$ patterns can be calculated *a priori*, to any desired detail. Each $h_3(x, y)$ pattern in the table corresponds to a known scene depth. Recall (6.3),

$$i_1(x, y) \; [\otimes] \; h_3(x, y) = i_2(x, y) \; . \tag{6.66}$$

Therefore, for any given image regions $i_1(x, y)$ and $i_2(x, y)$, the table of $h_3(x, y)$ patterns can be searched iteratively for the best $\hat{h}_3(x, y)$ which minimizes

$$\sum_{x=0}^{N-k} \sum_{y=0}^{N-k} \left[i_1(x, y) \; [\otimes] \; \hat{h}_3(x, y) - i_2(x, y) \right]^2 = \text{minimum} \; , \tag{6.67}$$

where the size of $i_1(x, y)$ is $N \times N$, the kernel $\hat{h}_3(x, y)$ is $k \times k$, and therefore $i_2(x, y)$ is $N - k + 1 \times N - k + 1$. The mechanics of this new approach are worked out in Chapter 7.

6.7.3 Constrained Inverse Filtering

Using regularization, the inverse filtering solution in the frequency domain can also be constrained, to make it more robust. For convenience, this solution will be derived using matrix notation, however it should not be confused with the matrix solution of the previous section. This inverse filtering solution, although constrained, still has the problems detailed in Section 6.5.3.

As shown in Figure 6.2, the uniquely characteristic part of $H_3(f_x, f_y)$ is the down-turned quadratic-type shape, centered at $f_x = f_y = 0$. Therefore $H_3(f_x, f_y)$ can be modelled by least squares fitting to a quadratic. The only problem with this approach, is that the quadratic model is valid only within a limited circle around $f_x = f_y = 0$, and the radius of this circle is dependent on the width of the quadratic. Assuming that only points within the valid radius are used,[14] regularization can be used to find $H_3(f_x, f_y)$. The regularized form of (6.27) minimizes the functional,

$$\|[\mathbf{I_{1DS}}] \cdot \mathbf{H_{3S}} - \mathbf{I_{2S}}\|^2 + \lambda \, \|[\mathrm{C}] \cdot \mathbf{H_{3S}}\|^2 = \text{minimum} \, , \tag{6.68}$$

where

- $[\mathbf{I_{1DS}}]$ is a matrix with $I_1(f_x, f_y)$ stacked along its diagonal as shown by Gonzalez and Wintz [50, §5.2];
- $\mathbf{H_{3S}}$ is a stacked vector formed from $H_3(f_x, f_y)$;
- $\mathbf{I_{2S}}$ is a stacked vector formed from $I_2(f_x, f_y)$;
- λ is a scalar parameter, which can be adjusted to be small and place more emphasis on fitting $\mathbf{H_{3S}}$ to the data, or large and place more emphasis on fitting $\mathbf{H_{3S}}$ to a quadratic shape; and
- $[\mathrm{C}]$ is a matrix which minimizes the magnitude of the second term if $\mathbf{H_{3S}}$ has a quadratic shape.

The matrix $[\mathrm{C}]$ is relatively easy to find. Let $H_3(f_x, f_y)$ be represented by the quadratic

$$H_3(f_x, f_y) = a(f_x^2 + f_y^2) + b \, . \tag{6.69}$$

[14] The experimental implementation in Section 8.3.1 avoids the problem of iteratively finding the valid radius by calculating it based on the *known* depth value.

Then in matrix form, (6.69) can be written as

$$\mathbf{H}_{3S} = [\mathbf{Q}] \cdot \mathbf{A} \ , \tag{6.70}$$

where

$$[\mathbf{Q}] = \begin{bmatrix} f_0^2 + f_0^2 & 1 \\ f_1^2 + f_0^2 & 1 \\ \vdots & \vdots \\ f_{N-1}^2 + f_0^2 & 1 \\ f_0^2 + f_1^2 & 1 \\ f_1^2 + f_1^2 & 1 \\ \vdots & \vdots \\ f_{N-1}^2 + f_1^2 & 1 \\ \vdots & \vdots \\ f_{N-1}^2 + f_{N-1}^2 & 1 \end{bmatrix} \quad \text{and} \tag{6.71}$$

$$\mathbf{A} = \begin{bmatrix} a \\ b \end{bmatrix} . \tag{6.72}$$

Using general least squares [123, §14.3], (6.70) can be solved for $\mathbf{A}$ as

$$\mathbf{A} = \left([\mathbf{Q}]^T [\mathbf{Q}] \right)^{-1} [\mathbf{Q}]^T \mathbf{H}_{3S} \ . \tag{6.73}$$

Now $\mathbf{A}$ contains the parameters to the best fitted quadratic, and the best quadratic is

$$[\mathbf{Q}] \mathbf{A} \ . \tag{6.74}$$

Multiplying (6.73) by $[\mathbf{Q}]$ on each side, yields

$$[\mathbf{Q}] \mathbf{A} = [\mathbf{Q}] \left([\mathbf{Q}]^T [\mathbf{Q}] \right)^{-1} [\mathbf{Q}]^T \mathbf{H}_{3S} \ . \tag{6.75}$$

Then the error between $\mathbf{H_{3S}}$ and the best quadratic is

$$\begin{aligned} & \|\mathbf{H_{3S}} - [\mathbf{Q}]\left([\mathbf{Q}]^T[\mathbf{Q}]\right)^{-1}[\mathbf{Q}]^T\mathbf{H_{3S}}\|^2 \text{ which} \\ = \; & \|\left([\mathbf{I}] - [\mathbf{Q}]\left([\mathbf{Q}]^T[\mathbf{Q}]\right)^{-1}[\mathbf{Q}]^T\right)\mathbf{H_{3S}}\|^2 , \end{aligned} \tag{6.76}$$

where $[\mathbf{I}]$ is the identity matrix. Comparing (6.76) with (6.68) reveals that

$$[\mathbf{C}] = [\mathbf{I}] - [\mathbf{Q}]\left([\mathbf{Q}]^T[\mathbf{Q}]\right)^{-1}[\mathbf{Q}]^T . \tag{6.77}$$

Now that an expression for $[\mathbf{C}]$ has been derived, the original regularization equation can be solved. The Euler equation [8] for (6.68) is solved for $\mathbf{H_{3S}}$ as follows

$$\begin{aligned} [\mathbf{I_{1DS}}]^T[\mathbf{I_{1DS}}]\mathbf{H_{3S}} - [\mathbf{I_{1DS}}]^T\mathbf{I_{2S}} + \lambda[\mathbf{C}]^T[\mathbf{C}]\mathbf{H_{3S}} &= 0 , \\ \left([\mathbf{I_{1DS}}]^T[\mathbf{I_{1DS}}] + \lambda[\mathbf{C}]^T[\mathbf{C}]\right)\mathbf{H_{3S}} &= [\mathbf{I_{1DS}}]^T\mathbf{I_{2S}} , \text{ and} \\ \mathbf{H_{3S}} &= \left([\mathbf{I_{1DS}}]^T[\mathbf{I_{1DS}}] + \lambda[\mathbf{C}]^T[\mathbf{C}]\right)^{-1}[\mathbf{I_{1DS}}]^T\mathbf{I_{2S}} . \end{aligned} \tag{6.78}$$

But since the quadratic parameters $\mathbf{A}$, are desired from $\mathbf{H_{3S}}$, then using (6.73), (6.78) becomes

$$\mathbf{A} = \left([\mathbf{Q}]^T[\mathbf{Q}]\right)^{-1}[\mathbf{Q}]^T\left([\mathbf{I_{1DS}}]^T[\mathbf{I_{1DS}}] + \lambda[\mathbf{C}]^T[\mathbf{C}]\right)^{-1}[\mathbf{I_{1DS}}]^T\mathbf{I_{2S}} . \tag{6.79}$$

The last step in solving the defocus problem by constrained inverse filtering, is to relate the quadratic parameters a and b in $\mathbf{A}$, to the distance to the scene D_o. This will be done in two steps: a and b will be related to the radius of the geometric optics defocus operator R, and the relationship of R to D_o has already been established.

The most convenient way of relating a and b to R, is to equate the zero crossing of the quadratic and the analogous pattern in $H_3(\rho_p)$. Recall from (6.10)

$$H_3(\rho_p) = \frac{R_1\rho_p J_1\left(\frac{2\pi R_2\rho_p P}{N}\right)}{R_2\rho_p J_1\left(\frac{2\pi R_1\rho_p P}{N}\right)} . \tag{6.80}$$

The zero crossings of (6.80) are found when

$$J_1\left(\frac{2\pi R_2 \rho_p P}{N}\right) = 0\,. \tag{6.81}$$

The first zero crossing occurs when the argument of J_1 is equal to 3.83. Therefore

$$\frac{2\pi R_2 \rho_p P}{N} = 3.83\,, \text{ and} \tag{6.82}$$

$$R_2 = \frac{3.83N}{2\pi\rho_p P}\,. \tag{6.83}$$

The same zero crossing in the quadratic model can be found by equating (6.69) to zero

$$0 = a(f_x^2 + f_y^2) + b\,. \tag{6.84}$$

Then, from the definition of ρ in (3.19), if a and b are in pixel units,

$$0 = a\rho_p^2 + b \text{ and} \tag{6.85}$$

$$\rho_p = \sqrt{-\frac{b}{a}}\,. \tag{6.86}$$

Substituting (6.86) into (6.83) yields

$$R_2 = \frac{3.83N}{2\pi P\sqrt{-\frac{b}{a}}}\,. \tag{6.87}$$

Finally, substituting (6.87) into (3.16) yields an equation for distance,

$$D_o = \frac{FD_i}{2f_2\frac{3.83N}{2\pi P\sqrt{-\frac{b}{a}}} + D_i - F}\,. \tag{6.88}$$

To summarize, constrained inverse filtering can be performed by taking two local regions $i_1(x,y)$ and $i_2(x,y)$ of the same size and windowing them with $w_1(x,y)$ and $w_2(x,y)$ respectively. After taking the discrete Fourier Transform, of both windowed regions, a regularized division and fit to a quadratic is done using (6.79). The quadratic parameters a and b in $\mathbf{A}$, are related to distance to the scene D_o, using (6.88).

6.8 CONCLUSIONS

This chapter has shown that solving the depth from focus problem can be condensed into the computational goal of finding the convolution ratio $h_3(x, y)$, which is a solution to the equation

$$i_1(x, y) \; [\otimes] \; h_3(x, y) = i_2(x, y) \; . \tag{6.89}$$

The operator $[\otimes]$ designates restricted convolution, where the borders of the kernel $h_3(x, y)$ are *not* convolved past the borders of the image $i_1(x, y)$. The local image region $i_1(x, y)$ was acquired with the defocus operator $h_1(x, y)$, and $i_2(x, y)$ was a local image region acquired with the defocus operator $h_2(x, y)$. It was also shown that with no noise, $h_3(x, y)$ can be expressed in terms of $h_1(x, y)$ and $h_2(x, y)$, as

$$h_1(x, y) \otimes h_3(x, y) = h_2(x, y) \; . \tag{6.90}$$

Using the geometric optics model for defocus, it was shown that $h_3(x, y)$ is a unique indicator of depth.

The solution for $h_3(x, y)$ can be formulated with inverse filtering or with a matrix based approach. When only one of the images was blurred, in a noiseless environment, it was shown that the inverse filtering solution in the frequency domain had two fundamental limits to accuracy:

- Finding the Fourier domain representation of a local signal requires windowing the spatial domain signal, however it introduces residual distortions in the spatial frequency domain; and
- Border effects from neighbouring regions can introduce spurious data.

A novel application of a matrix based method was introduced, which circumvents the two problems with the inverse filtering solution. When two images were blurred,

$h_3(x, y)$ must also be windowed. The problem of border effects was still solved by the matrix based method.

When noise was added to the model, the matrix based approach was augmented with regularization to deal with the ill-posed nature of the problem. Since a closed form solution was generally not possible for an arbitrary model of the defocus operator, an iterative solution was given, which solved for $h_3(x, y)$ and hence determined depth. To provide a fair comparison for the experimental work in Chapter 8, the inverse filtering approach was also augmented with regularization to produce a constrained inverse filtering solution.

This chapter has introduced two concepts which have previously not been reported in the depth from focus literature:

1. A generalized solution by a matrix based method; and

2. The application of regularization to the problem.

Chapter 7

IMPLEMENTATION OF NEW METHOD

7.1 INTRODUCTION

This chapter describes the implementation of the author's new iterative matrix based method for solving the depth from focus problem. Chapter 6 provided the theoretical basis for this method, and Chapter 8 will present the experimental results.

7.2 OVERVIEW

The new method proposed by the author for solving the depth from focus problem is summarized in Figure 7.1. Before the method is implemented, the family of possible patterns for $h_3(x, y)$ must be calculated. These patterns are stored in a table which will be searched, under the conditions below.

Two images of the same scene i_1 and i_2, are acquired with two different defocus operators. To change the defocus operator, any of three camera parameters can be varied:

1. Position of image plane;
2. Focal length; or
3. Aperture.

Changing each camera parameter has some adverse side effects: the first two affect

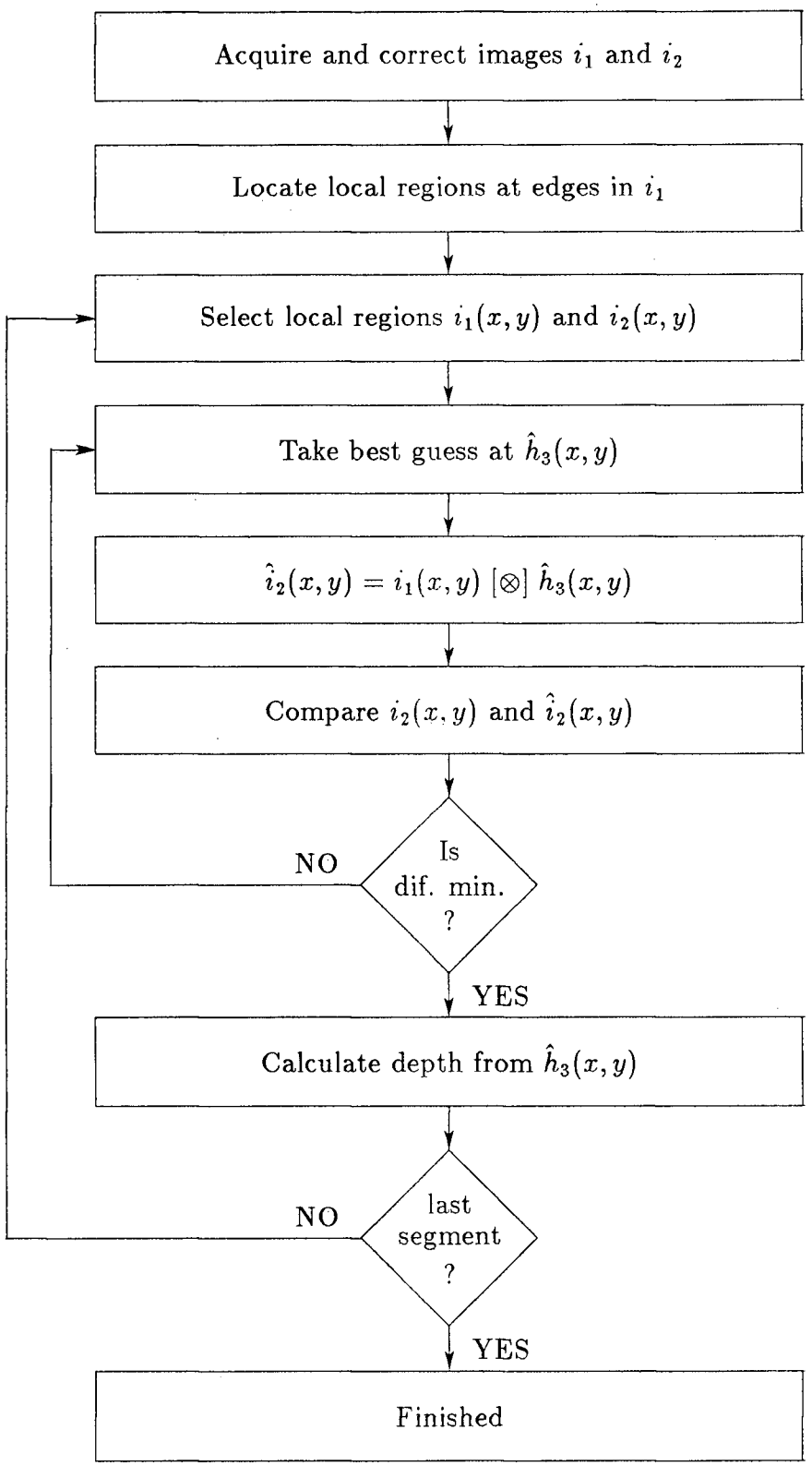


Figure 7.1: Flow chart of new, iterative matrix based method for solving the depth from focus problem.

the image magnification, and the third influences the image brightness.[1] Varying the aperture was selected as the best choice, although the method described in this chapter is not restricted to that choice.

After the images are corrected radiometrically, local regions are selected at edges in i_1. For each set of local regions $i_1(x, y)$ and $i_2(x, y)$, (6.67) is solved iteratively for the best $\hat{h}_3(x, y)$, as proposed in Section 6.7.2. This chapter provides the details for all the steps in Figure 7.1.

7.3 CALCULATING A TABLE OF $h_3(x, y)$ PATTERNS

To implement the new method, a table of all possible $h_3(x, y)$ patterns must be calculated, and this may be done *a priori*. To calculate this table, a model of the defocus operator must be assumed, or experimentally derived. To keep the method general, a table of $h_3(x, y)$ patterns will first be calculated by assuming the geometric optics model of defocus. This model has been used by several researchers [65][66][17], and has a good theoretical justification, as shown in Section 3.4. In Section 7.8, another table of $h_3(x, y)$ patterns will be calculated using an experimentally derived defocus operator, which is then optimized to a particular lens.

Calculation of a table of $h_3(x, y)$ patterns, begins with (6.8),

$$h_1(x, y) \otimes h_3(x, y) = h_2(x, y) \ . \tag{7.1}$$

Expressed in matrix form, (7.1) becomes

$$[\mathbf{h_{1BC}}] \cdot \mathbf{h_{3S}} = \mathbf{h_{2S}} \ , \tag{7.2}$$

where $[\mathbf{h_{1BC}}]$ is a *block circulant* matrix, and $\mathbf{h_{3S}}$ and $\mathbf{h_{2S}}$ are row-stacked vectors

[1] More than one camera parameter can be varied, as long as the affects don't cancel. Referring to (3.15), this means that for a constant D_o, the new parameters F, D_i, or f *must* change the value of R.

formed as shown by Gonzalez and Wintz [50, §5.2].[2] At first glance, (7.2) can be solved for $\mathbf{h_{3S}}$ by

$$\mathbf{h_{3S}} = [\mathbf{h_{1BC}}]^{-1} \cdot \mathbf{h_{2S}} \tag{7.3}$$

An immediate problem with solving (7.3) is its size. For example, if $h_1(x,y)$ and $h_2(x,y)$ are 64×64 pixels, this requires the inversion of $[\mathbf{h_{1BC}}]$, with a size of 4096×4096!

Fortunately, it is possible to take advantage of the fact that $h_1(x,y)$ and $h_2(x,y)$ are circularly symmetric. This can generally be expected of defocus operators formed from optical systems with circular apertures [140].[3] As shown in the next section, deconvolution of circularly symmetric operators can be performed in one dimension.

7.3.1 Justification For Calculations In One Dimension

To prove that (7.1) can be solved in one dimension, it will first be presented in two dimensions in the frequency domain. Let the following be Fourier pairs:

$$h_1(x,y) \iff H_1(f_x,f_y)\ , \tag{7.4}$$

$$h_2(x,y) \iff H_2(f_x,f_y) \text{ and} \tag{7.5}$$

$$h_3(x,y) \iff H_3(f_x,f_y)\ . \tag{7.6}$$

Then according to the convolution theorem, (7.1) becomes

$$H_1(f_x,f_y) \cdot H_3(f_x,f_y) = H_2(f_x,f_y)\ . \tag{7.7}$$

Since $h_1(x,y)$, and $h_2(x,y)$ are circularly symmetric, $H_1(f_x,f_y)$ and $H_2(f_x,f_y)$ are also circularly symmetric [18, p. 247]. By implication of (7.7), $H_3(f_x,f_y)$ is circularly symmetric as well. Also, $H_1(f_x,f_y)$, $H_2(f_x,f_y)$ and $H_3(f_x,f_y)$ can be centered at the origin

[2] Note that $h_1(x,y)$ has a limited spatial extent so that a *circulant* convolution matrix can be used. This is different from the situation examined in Appendix B, where $i_1(x)$ was a local part of an image, which resulted in a *Toeplitz* convolution matrix.

[3] Another approach, of course, would be to perform the computations in the frequency domain, however the reason for staying in the spatial domain will become obvious in Section 7.3.2, once regularization is introduced.

with no loss of generality. Then (7.7) can be expressed in one-dimensional form, since $H_1(f_x, f_y)$, $H_2(f_x, f_y)$ and $H_2(f_x, f_y)$ are uniquely identified by one slice through the origin. Again, with no loss of generality, this slice will be taken at $f_x = 0$. Then,

$$H_{1y}(f_y) \cdot H_{3y}(f_y) = H_{2y}(f_y) \ , \tag{7.8}$$

where

$$H_{1y}(f_y) = H_1(f_x, f_y)|_{f_x=0} \ , \tag{7.9}$$

$$H_{2y}(f_y) = H_2(f_x, f_y)|_{f_x=0} \ \text{and} \tag{7.10}$$

$$H_{3y}(f_y) = H_3(f_x, f_y)|_{f_x=0} \ . \tag{7.11}$$

To return to the spatial domain, let the following be Fourier pairs:

$$h_{1y}(y) \Longleftrightarrow H_{1y}(f_y) \ , \tag{7.12}$$

$$h_{2y}(y) \Longleftrightarrow H_{2y}(f_y) \ \text{and} \tag{7.13}$$

$$h_{3y}(y) \Longleftrightarrow H_{3y}(f_y) \ . \tag{7.14}$$

Then (7.8) becomes

$$h_{1y}(y) \otimes h_{3y}(y) = h_{2y}(y) \ . \tag{7.15}$$

According to the *projection slice theorem* [129, §8.2], if $H_{1y}(f_y)$ is one slice of $H_1(f_x, f_y)$ at $f_x = 0$, and $H_{1y}(f_y) = \mathcal{F}(h_{1y}(y))$ then

$$h_{1y}(y) = \int_{-\infty}^{\infty} h_1(x, y)dx \ . \tag{7.16}$$

Also similarly

$$h_{2y}(y) = \int_{-\infty}^{\infty} h_2(x, y)dx \ \text{and} \tag{7.17}$$

$$h_{3y}(y) = \int_{-\infty}^{\infty} h_3(x, y)dx \ . \tag{7.18}$$

That is, in the spatial domain, $h_{1y}(y)$, $h_{2y}(y)$ and $h_{3y}(y)$ are *not* slices of the two-dimensional functions, but rather an integration over x. Then (7.15) can be rewritten as

$$\int_{-\infty}^{\infty} h_1(x,y)dx \otimes \int_{-\infty}^{\infty} h_3(x,y)dx = \int_{-\infty}^{\infty} h_2(x,y)dx \ , \tag{7.19}$$

which is the expression of (7.1) in one dimension.

The final step involves reversing the integration process, and transforming the one-dimensional function $h_{3y}(y)$ back into the two-dimensional function $h_3(x,y)$. This is possible because $h_3(x,y)$ is rotationally symmetric, and can be written as $h_3(r)$. Figure 7.2 illustrates the geometry of the problem.[4] Let

- A_1 be the area in the annulus sector to the right of line 1;
- A_2 be the area in the large annulus sector to the right of line 2; and
- A_3 be the area in the small annulus sector to the right of line 2.

The transformation from $h_{3y}(y)$ to $h_3(r)$ is done according to the following steps:

$$h_3(N) = \frac{h_{3y}(N)}{A_1} \ , \tag{7.20}$$

$$h_3(N-1) = \frac{h_{3y}(N-1) - h_3(N) \cdot A_2}{A_3} \quad \text{and} \tag{7.21}$$

$$h_3(N-2) = \text{etc.} \tag{7.22}$$

Using these results, (7.2) becomes the matrix equation

$$[\mathbf{h_{1C}}] \cdot \mathbf{h_3} = \mathbf{h_2} \quad \text{and} \tag{7.23}$$

$$\mathbf{h_3} = [\mathbf{h_{1C}}]^{-1} \cdot \mathbf{h_2} \ , \tag{7.24}$$

where

[4]This is also based on the assumption that the ends of $h_{3y}(y)$ go to zero. This assumption will be confirmed in the Section 7.3.2.

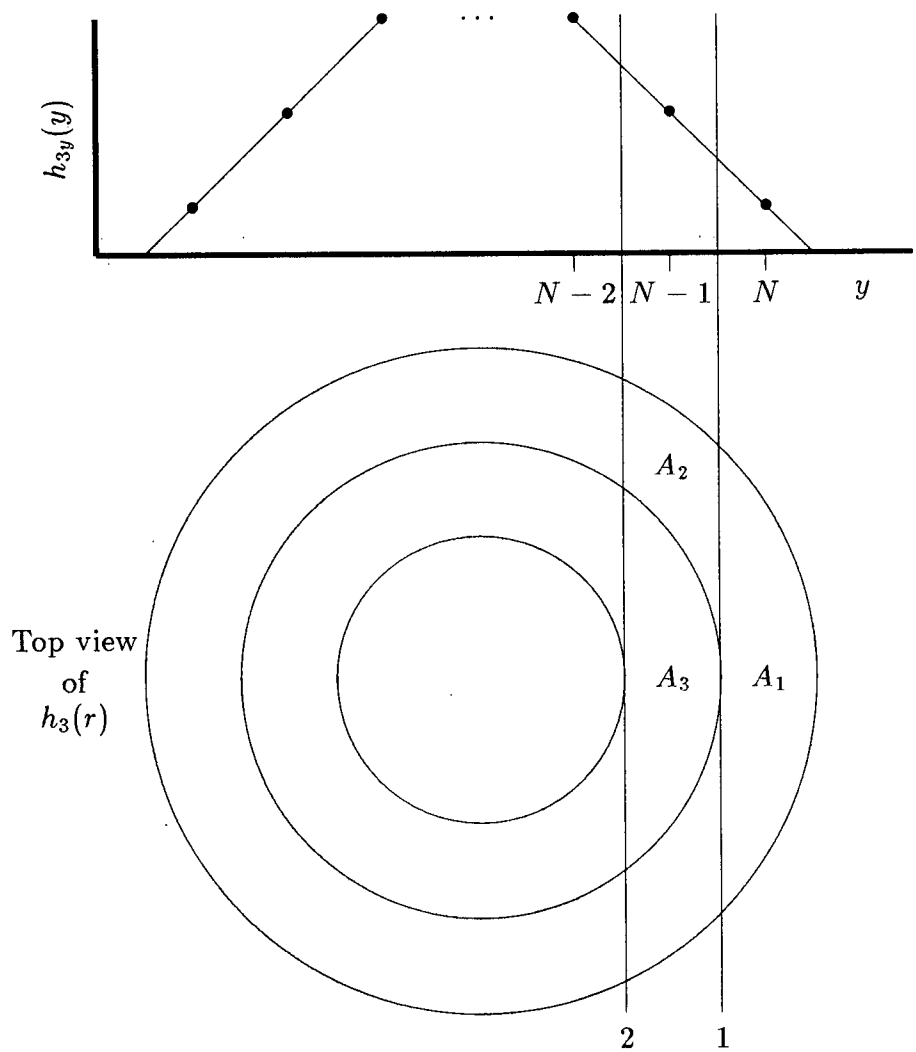


Figure 7.2: Geometry of transforming $h_{3y}(y)$ to $h_3(r)$.

- $\mathbf{h_3}$ and $\mathbf{h_2}$ are vectors formed from $h_{3y}(y)$ and $h_{2y}(y)$; and
- $[\mathbf{h_{1C}}]$ is a circulant matrix formed from $h_{1y}(y)$ according to Gonzalez and Wintz [50, §5.2].

Unfortunately, (7.24) will still not yield a usable result for $h_3(x, y)$. To illustrate the problems and what can be done about them, the next section will work through an example with specific defocus operators assigned to $h_1(x, y)$ and $h_2(x, y)$.

7.3.2 An Example Of Calculating One $h_3(x, y)$ Pattern

Continuing with the geometric optics model of defocus, one $h_3(x, y)$ pattern will be calculated in this section. Let

- The focal length of the lens $F = 50\ mm$; and
- The point of perfect focus be set at $D_o = 1.0\ meter$.

Then the distance to the image plane can be found by (2.1) to be $D_i = 52.63\ mm$.

Next, it will assumed that

- The distance to the object $D_o = 0.85\ meters$;
- Image i_1 is taken with an aperture of $f_1 = 2.0$;
- Image i_2 is taken with an aperture of $f_2 = 1.3$;
- The digitization ratio is $P = 60.0$ pixels$/mm$.

Then using (3.15) and (3.17) it can be seen that $h_1(x, y)$ is a pillbox with a radius of $R_{1p} = 6.97$ pixels, and $h_2(x, y)$ is a pillbox with a radius of $R_{2p} = 10.72$ pixels. Next, $h_1(x, y)$ and $h_2(x, y)$ can be integrated according to (7.16) and (7.17) to form $h_{1y}(y)$ and $h_{2y}(y)$. These examples are shown in Figure 7.3.

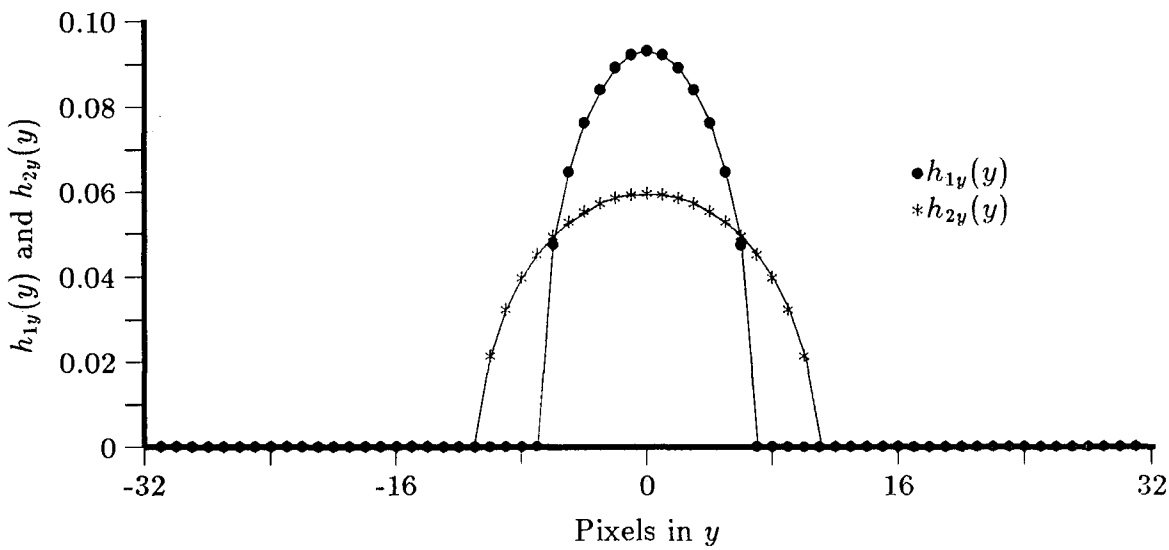


Figure 7.3: Examples of functions $h_{1y}(y)$ and $h_{2y}(y)$ with the camera parameters given in the text.

If (7.24) is now solved, using a dimension of $N = 64$ pixels, the resulting $\mathbf{h}_3$, shown in Figure 7.4, has two undesirable qualities:

- It shows wild fluctuations which are likely sensitive to any noise in $h_1(x, y)$ or $h_2(x, y)$; and
- The ends of the waveform do not taper to zero, which means that it cannot be contained within the spatial support given.

At this point there are two ways to proceed:

1. Define criterion for the "best" $\mathbf{h}_3$ and try to find it; or
2. Find any $\mathbf{h}_3$ which yields "good" qualitative results.

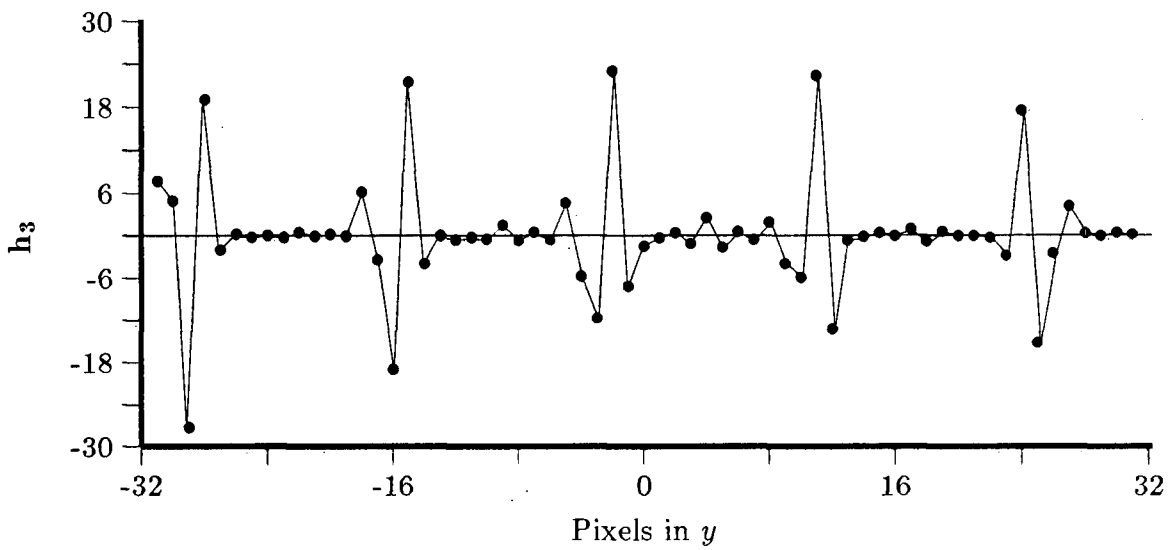


Figure 7.4: Example of h_3 solved by matrix inversion.

Experience in working with this problem has shown the author that the success of the proposed method is not critically dependent on finding the "best" pattern for $\mathbf{h}_3$.[5] Therefore any $\mathbf{h}_3$ which yields "good" qualitative results should suffice. The quality of $\mathbf{h}_3$ will be checked at the end of this Section, by substituting it back into (7.23).

Smoothing h_3

To smooth out $\mathbf{h}_3$, (7.23) can be incorporated into a regularization equation [8],

$$\| [\mathbf{h}_{1C}] \cdot \mathbf{h}_3 - \mathbf{h}_2 \|^2 + \lambda_1 \| [C_C] \cdot \mathbf{h}_3 \|^2 = \text{minimum} , \tag{7.25}$$

where

- $[C_C]$ can be any circulant matrix which detects the roughness of a function (such as a differentiating function); and

[5] This will be illustrated later in Section 8.3.2. The set of $\mathbf{h}_3$ patterns based on *geometric optics*, are significantly different from the set of $\mathbf{h}_3$ patterns based on the experimentally obtained defocus operator, yet the two will recover depth almost equally well.

- λ_1 is a scalar parameter which controls the emphasis on the data (the first term) or the smoothness (the second term).

The matrix $[\mathbf{C_C}]$, will be defined as the Laplacian of the Gaussian put into a circulant matrix, where

$$\nabla^2 G(y) = -\frac{1}{\sqrt{2\pi}\ \sigma^3}\left(1 - \frac{y^2}{\sigma^2}\right)\exp^{-\frac{y^2}{2\sigma^2}}, \tag{7.26}$$

and $\sigma = 1$.

The resulting $\mathbf{h_3}$ for $\lambda_1 = 0.1$ is shown in Figure 7.5.

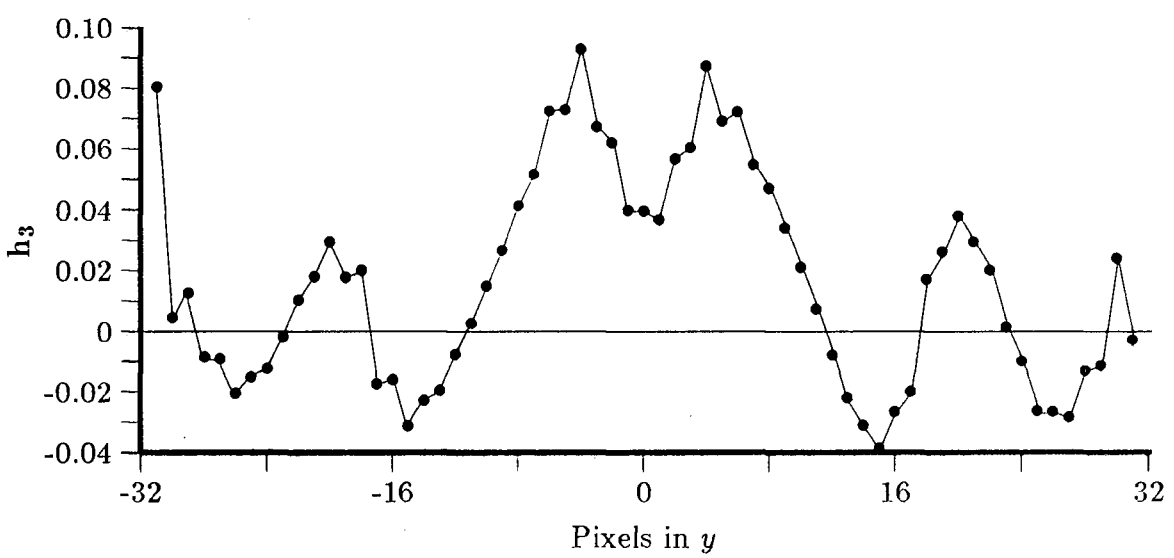


Figure 7.5: Example of $\mathbf{h_3}$ solved by regularization with smoothing.

Containing the spatial spread of $\mathbf{h_3}$

The next problem to be tackled, is containing the spatial spread of $\mathbf{h_3}$. Extra terms can be introduced into the regularization equation (7.25), to force the sides of $\mathbf{h_3}$ down

to zero. The new regularization equation is

$$\| [\mathbf{h_{1C}}] \cdot \mathbf{h_3} - \mathbf{h_2} \|^2 + \lambda_1 \| [\mathbf{C_C}] \cdot \mathbf{h_3} \|^2 + \lambda_2 \| [\mathbf{T_D}] \cdot \mathbf{h_3} \|^2 = \text{minimum} \ , \tag{7.27}$$

where

- $[\mathbf{T_D}]$ is a diagonal matrix, formed from a function which penalizes $\mathbf{h_3}$ for having non-zero tails; and
- λ_2 is a scalar parameter which controls the emphasis on the minimizing the tails of $\mathbf{h_3}$.

Although, somewhat arbitrary, $[\mathbf{T_D}]$ was formed as the diagonal form of the function

$$t(y) = \left[1 - \sin\left(\frac{\pi y}{N} + \frac{\pi}{2}\right)\right]^2 \ . \tag{7.28}$$

This function is zero at $y = 0$ and rises to one for $y = \pm N/2$. Finally with $\lambda_1 = 0.1$ and $\lambda_2 = 1.0$, the resultant $\mathbf{h_3}$ is shown in Figure 7.6.

The design of $\mathbf{h_3}$ involves trade-offs, however the regularization technique controls how far $\mathbf{h_3}$ strays from the original data, by adjustment of λ_1 and λ_2. The qualitative test of $\mathbf{h_3}$ is to substitute it back into (7.23) to get a new $\hat{\mathbf{h}}_2$,

$$\hat{\mathbf{h}}_2 = [\mathbf{h_{1C}}] \cdot \mathbf{h_3} \ . \tag{7.29}$$

The new $\hat{\mathbf{h}}_2$ can then be compared to the original $\mathbf{h_2}$. Figure 7.7 graphs the resulting $\hat{\mathbf{h}}_2$ as well as the original $\mathbf{h_2}$, to show that $\mathbf{h_3}$ is still faithful to the original data. After working through this example, a table of $h_3(x, y)$ patterns can now be calculated for a range of expected distances. For distances closer to the point of focus, $h_3(x, y)$ has a smaller spatial extent, so that it can be represented in less than 64×64 pixels.

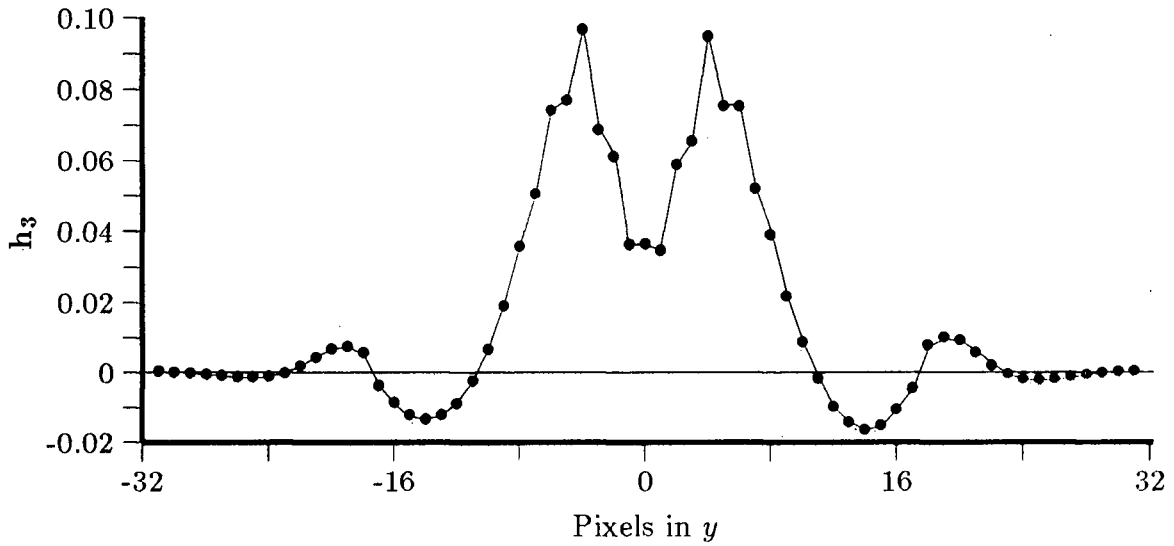


Figure 7.6: Example of $\mathbf{h}_3$ solved by regularization with smoothing and suppression of ends.

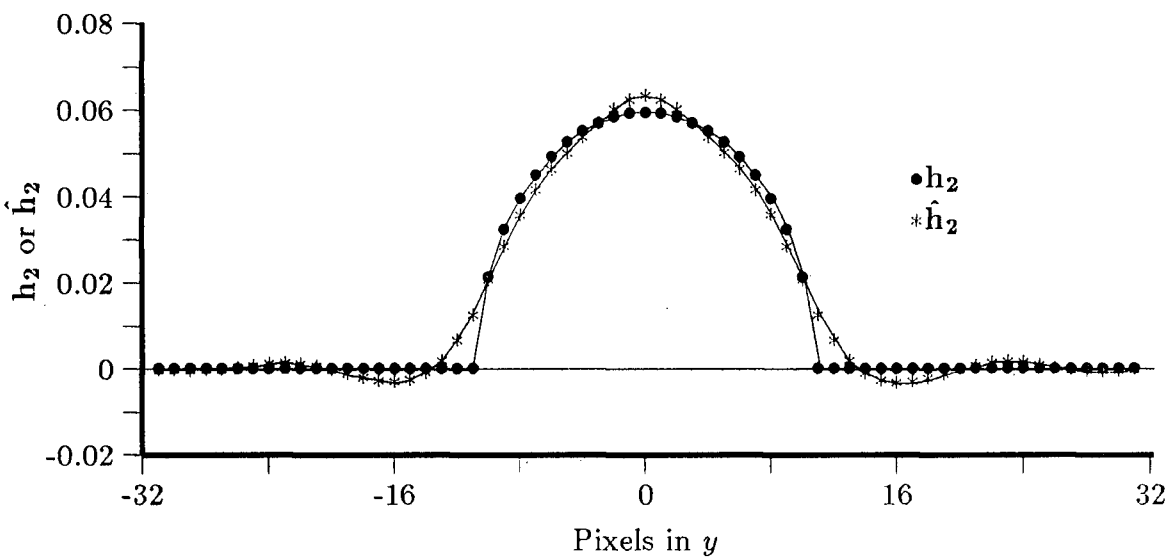


Figure 7.7: Comparison of $\hat{\mathbf{h}}_2$ and $\mathbf{h}_2$.

7.3.3 Table Of $h_3(x, y)$ Patterns For Geometric Optics

In a manner similar to the example shown in the last section, a table of $h_3(x, y)$ patterns was calculated for $D_o = 0.80$ to $D_o = 0.95$ *meters*. A total of 16 patterns were stored, corresponding to 1 *cm* increments. Patterns at distances in between stored patterns, can be calculated by simple linear interpolation.

7.4 ACQUIRING THE IMAGES

As explained in Section 7.2, two images of the same scene will be acquired at two different aperture settings. This section explains the details of image acquisition and correction.

The experimental equipment used for this research was

- A *JAVELIN* Model JE2062 black and white digital CCD camera with 384 pixels horizontal and 485 pixels vertical resolution, and RS-170 standard video output;
- A 50 *mm* and a 16 *mm* lens;
- A *PC VISION* frame grabber with 512×512 resolution at 8 bits; and
- An 8 *MHz IBM PC XT* for data acquisition control and processing.

The 50 *mm* lens was used for the experiments described in Chapter 8. As shown in Section 3.7, the long focal length allowed the active area to be farther away from the camera. The experiments were performed with the point of sharpest focus set at 1 *meter* from the camera, and objects were placed within the interval of 0.95 to 0.80 *meters* from the camera.

Keeping the active area far away from the camera also tended to minimize the perspective angle between objects at the border of the image and the optical axis.

This in turn minimized the difference between the tangent distance directly to the object and the distance along the optical axis. Therefore, it was reasonable to treat the acquired images as orthographic. This simplified the geometric calculation of the actual distances.

The two aperture settings chosen were $f_2 = 1.3$ for the large defocus operator and $f_1 = 2.0$ for the small defocus operator. Since these aperture settings were only one f-stop apart, the difference in illumination between the two images was minimized. However, the difference between the defocus operators was also slight, making recovery of $h_3(x, y)$ a more challenging problem.

To help minimize the noise, sixteen images were averaged to produce the image used for processing. To measure the noise, 16 images (64×64 pixels) of a white surface were averaged. As shown in Figure 7.8, the histogram of the noise was confirmed to have a Gaussian distribution with good symmetry, indicating the mean was zero as previously assumed in Section 6.3.

7.5 CORRECTING THE IMAGES

7.5.1 Radiometric Correction

Since depth from focus relies on the relative grey levels between the two acquired images, radiometric correction is critical. The intensity values are influenced by two main blocks of the system.

1. In addition to blurring the scene by convolution with the defocus operator, the optical system introduces the following undesirable changes:

    - Aberrations change the shape of the defocus operator from the ideal theoretical case [132, §2.4.2];

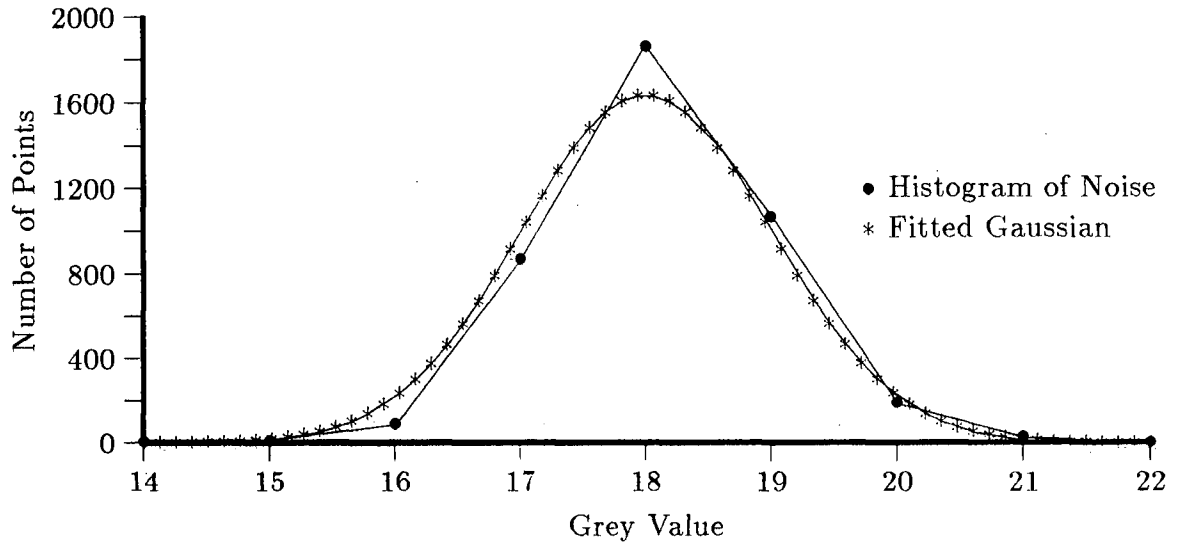


Figure 7.8: Histogram of noise with fitted Gaussian, after averaging 16 images (64 × 64) of a white surface.

- The intensity of the image with the smaller aperture setting is darkened in relation to the image with the larger aperture setting; and
- The illuminance off the axis is darkened by cosine effects which darkens the illumination proportional to $\cos^4 \alpha$, where α is the angle with the optical axis, and vignetting which further darkens the off-axis illumination. [75, §7.17 & §7.18][132, §2.4.1].

2. The sensor and data acquisition system may introduce further non-linearities into the grey values of the image.

The convolution of the scene with the defocus operator takes place as the image goes through the lens of the camera, *before* it it digitized by the sensor. Since the scene is to be deconvolved from the image *after* it has been digitized, it is important that the image acquisition process be made linear. Therefore, the two distortions above must

be *undone* in the opposite order which they occurred. Since intensity differences due to different apertures is spatially invariant, it is easier to bundle this together with correction of sensor non-linearities. Then the images will be corrected in the following two steps, before deconvolving the defocus operator:

1. Correct non-linearities in sensor and data acquisition system and intensity differences due to different apertures, which are spatially invariant.

2. Correct spatially variant cosine effects and vignetting.

Space-invariant corrections

The first correction, for spatially invariant factors was accomplished by a radiometric linearization to a known standard. The calibration was done on a small local area at the center of the sensor, so that space-variant errors would be negligible. The reference was a Kodak Gray Scale, which had 20 steps in 0.10 density increments, between a "white" density of $d = 0.0$ and a "black" density at $d = 1.90$. For each density level, an average image intensity was recorded at both aperture settings. The reflectance percentage p, is related to density d, by the formula [67][80, p. 389]

$$p = \frac{100}{10^d} \,. \tag{7.30}$$

The results are shown in Figure 7.9.[6] A polynomial was fitted to both lines to linearize them to the same reflectance values.

To corroborate these results, the pixel intensity values were also compared to values obtained with a radiometer mounted beside the camera. The radiometer measured the light intensity in $\mu\mathrm{W}/cm^2$. These units are related linearly to the light intensity

[6] The large non-linearity was puzzling, since the CCD has been reported to be fairly linear [67]. After some investigations, it was discovered that most of this non-linearity was due to the *PC VISION* frame grabber, which seems to be taking the square root of the intensity, before the image is digitized.

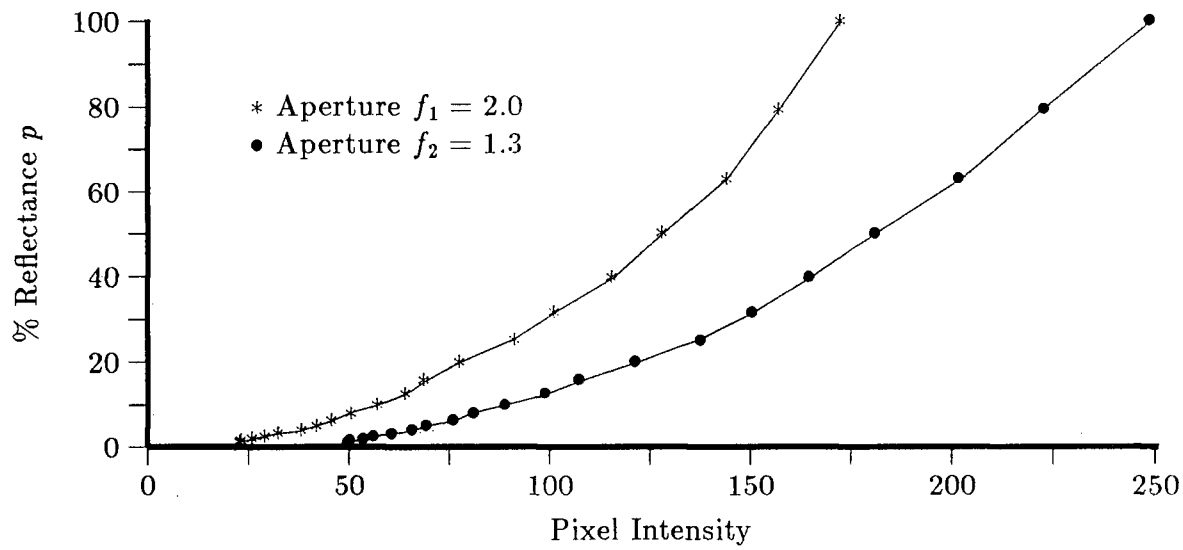


Figure 7.9: Radiometric calibration of camera sensor and data acquisition system to Kodak Grey Scale.

which according to Section 3.3, was necessary for deconvolution. The results shown in Figure 7.10, show a strong qualitative similarity to the Kodak calibration curves of Figure 7.9.[7]

Space-variant corrections

The correction for cosine effects and vignetting was performed by acquiring a series of images of a uniformly lit white sheet, at different light intensities. These images were radiometrically corrected as outlined above, and a polynomial fitted to each surface. The surfaces showed a peak close to the middle and less intensity toward all the borders, as shown in Figure 7.11. It was found that for any given set of coordinates, the surface peak was related linearly to the measured intensity. This result was quantified as a set

[7] These results were considered less accurate however, because the calibration information had to be obtained *beside* the camera sensor, and also the light level had to be changed for the various readings.

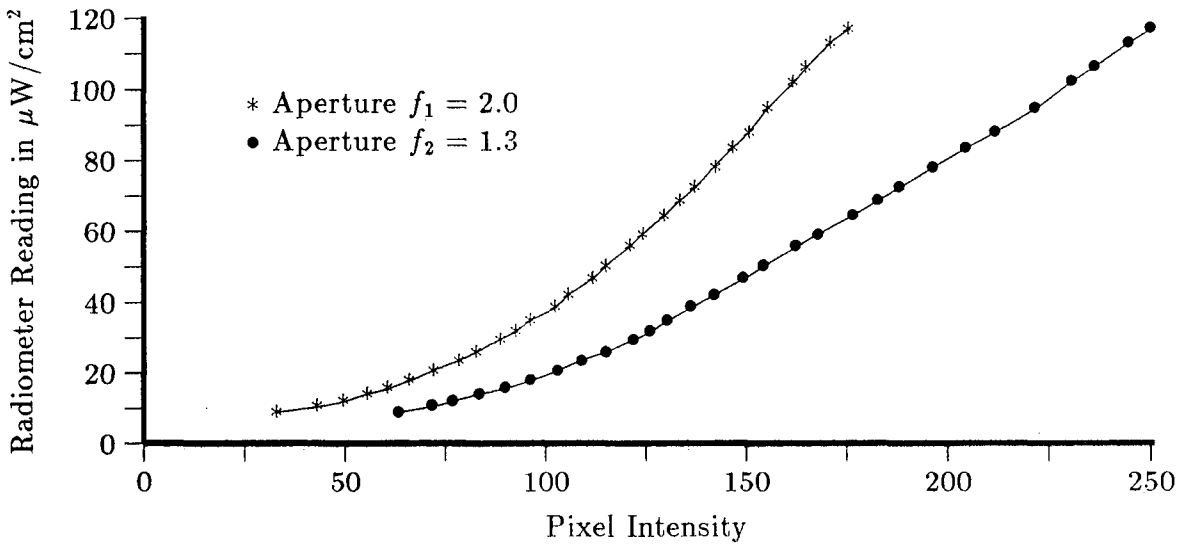


Figure 7.10: Radiometric calibration of camera sensor and data acquisition system to an external Radiometer.

of tables, which for all image coordinates gave a multiplier and an offset to drive the intensity level to the peak intensity. This correction effectively raised up all the sides of the surface to a flat plane at the same intensity as the peak value.

The experimental results were not compared to the theoretical model, since the geometric parameters of vignetting were unknown.

Resolving power of aperture

The resolving power of the aperture can be shown to be adequate for the lens geometry and CCD sensor used. Using Rayleigh's criterion, the minimum resolvable separation of geometrical images is [51, p. 130],

$$\delta_{\text{ideal}} = 1.22\frac{\lambda D_i}{L} . \tag{7.31}$$

For the images acquired, the worst case numbers are:

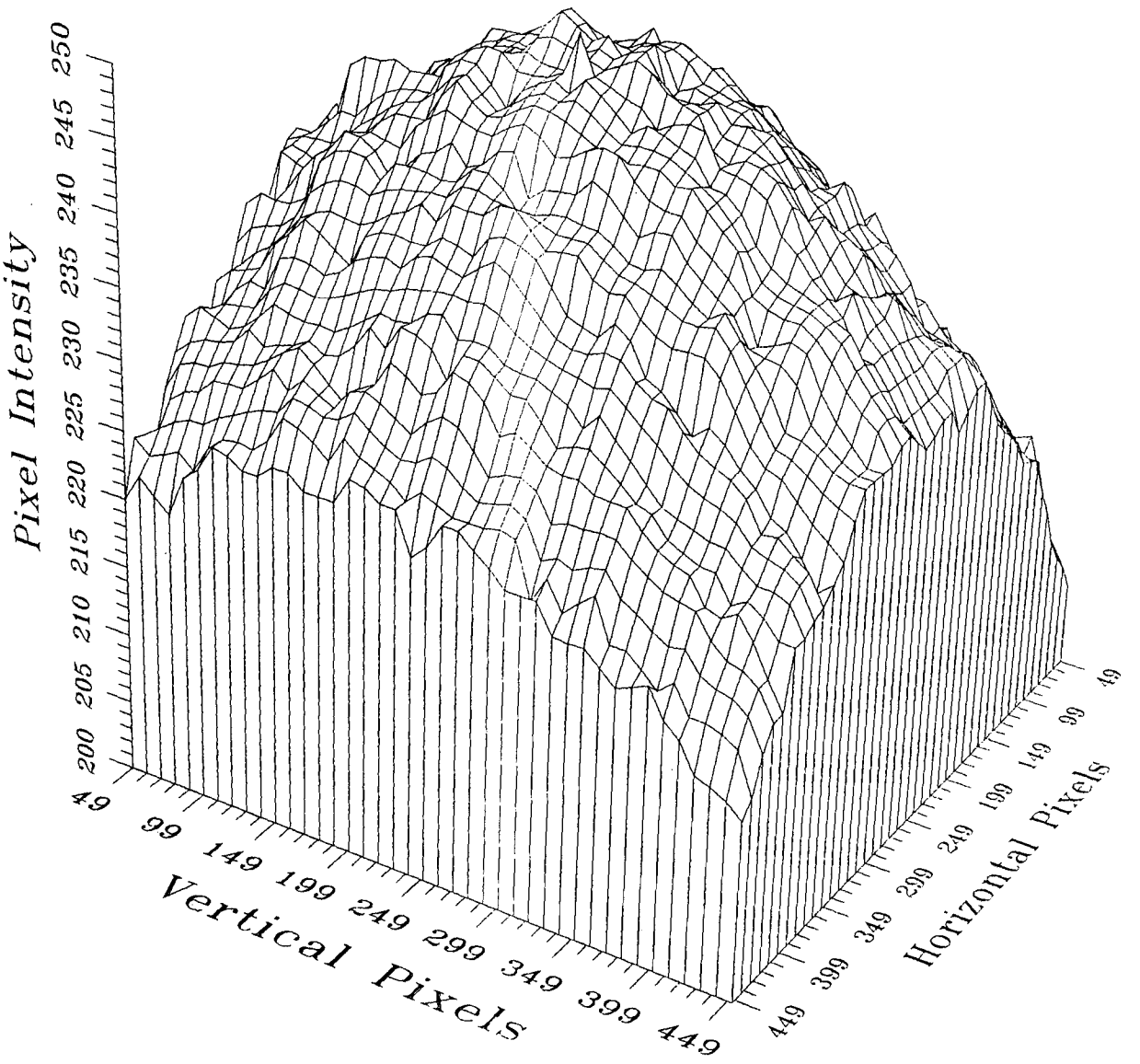


Figure 7.11: A profile of an image of an evenly illuminated background shows the spatially variant radiometric errors due to cosine effects and vignetting.

- $\lambda = 800\ nm$ for the peak spectral response of a CCD [13, p. 91][1][13];
- $D_i = 52.63\ mm$ is the distance from the lens to the sensor; and
- $L = 25\ mm$ is the diameter of the aperture at $f = 2.0$ for the 50 mm lens.

Then $\delta_{\text{ideal}} = 2.05 \times 10^{-6}\ m$. The sensor used for these experiments has a pixel size of $\delta_{\text{actual}} = 1.67 \times 10^{-5}\ m$. Since

$$\delta_{\text{actual}} > \delta_{\text{ideal}}\ , \tag{7.32}$$

then the resolving power of the aperture is adequate for the sensor used.

Chromatic effects

It was also shown experimentally that chromatic aberrations had a minimal affect on the defocus operator. According to the method outlined in Section 7.8, the point spread function was calculated for three images:

- One with an orange filter placed in front of the camera;
- One with a blue filter placed in front of the camera; and
- One with no filters.[8]

The results, after scaling amplitudes and smoothing, were virtually identical. On that basis, it was decided that the shape of the defocus operator could be estimated *independent* of the chromatic content of the illumination.

[8] Kodak Wratten gelatin filters were used. Orange and blue were chosen since they are toward the opposite ends of the colour spectrum (although red would have been better, if it was available). Although the exact frequency band-pass characteristics of the filters were not known, it was of secondary importance, since the experiment was first trying to establish if the filters made a *difference* or not.

7.5.2 Geometric Correction

The experimental data was not corrected for any geometric distortion. Correction for geometric distortion is well documented in the literature [149][98], and was considered beyond the scope of this work. More work could be done in this area, as discussed in Section 9.4.1.

7.6 FINDING EDGES

After the images had been acquired and corrected, the next step in the implementation of the new method for depth from focus, was to find edges in the images. Edges were located, because they were a prime source of information concerning the nature of the defocus operator. The sharper image i_1, was used as the input to the edge detector.

To find the edges, the $\nabla^2 G$ operator of Marr and Hildreth [106][107] was used,

$$\nabla^2 G(x,y) = -\frac{1}{\pi\sigma^4}\left(1 - \frac{x^2+y^2}{2\sigma^2}\right) e^{-\frac{x^2+y^2}{2\sigma^2}} , \tag{7.33}$$

where decreasing σ increases the sensitivity. This filter is rotationally invariant and easy to implement in one convolution pass. Although this method suffers from localization problems [147], which may be ameliorated by a directional filter such as Canny's [25] and other edge detectors, the location of edges was instead corrected by checking the original image. Local regions for depth processing were identified by the following steps in software:

1. The $\nabla^2 G$ filtered image was scanned for zero crossings;

2. The center of the edge was found by a search in the original image $\mathbf{i}_1$; and

3. The size of the local region $i_1(x,y)$ was set to $N \times N$, to include the complete

width of the edge.[9]

Each of these local regions can now be processed to identify the distance from the camera to that local region in the scene. Potentially, all of these local regions can be processed in parallel, however in this implementation they were processed sequentially.

7.7 FINDING DISTANCE TO SCENE

To find the distance to a local region in the scene, the best $\hat{h}_3(x,y)$ pattern which minimized (6.67),

$$\sum_{x=0}^{N-k}\sum_{y=0}^{N-k}\left[i_1(x,y)\ [\otimes]\ \hat{h}_3(x,y) - i_2(x,y)\right]^2 = \text{minimum}\ , \tag{7.34}$$

had to be found. This minimization was performed iteratively by the following steps:

1. One $h_3(x,y)$ pattern was picked, called $\hat{h}_3(x,y)$ with dimensions $k \times k$. Intuitively, the best place to start was with the pattern in the middle of the table, which represented the mid-point of expected distances in the scene.

2. The local region $i_1(x,y)$ of size $N \times N$, was convolved with $\hat{h}_3(x,y)$ to obtain $\hat{i}_2(x,y)$. This was a restricted convolution, where the borders of $\hat{h}_3(x,y)$ stayed within the borders of $i_1(x,y)$. The area of $\hat{i}_2(x,y)$ was therefore $N-k+1 \times N-k+1$, which was smaller than the area of $i_1(x,y)$. The area of $i_2(x,y)$ was chosen to match the area of $\hat{i}_2(x,y)$.

3. Final radiometric correction was performed on $\hat{i}_2(x,y)$. Although both images were radiometrically corrected as indicated in Section 7.5.1, slight variations still existed. Therefore, the final radiometric adjustment was performed by finding

[9] For simplicity in this implementation, the scanning for these three steps was done horizontally, which favoured vertical edges. Extensions to include edges of any orientation are straightforward.

a scale m and offset b which best matched $\hat{i}_2(x,y)$ to $i_2(x,y)$. That is, χ^2 was minimized, according to

$$\chi^2 = \sum_{x=0}^{N-k}\sum_{y=0}^{N-k}\left(m\hat{i}_2(x,y) + b - i_2(x,y)\right)^2 = \text{minimum} . \tag{7.35}$$

The optimal m and b were easily found, by solving the Euler equations of (7.35),

$$m = \frac{S_{\hat{i}i} - S_{\hat{i}}S_i/(N-k+1)^2}{S_{\hat{i}\hat{i}} - S_{\hat{i}}^2 S_i/(N-k+1)^2} \text{ and} \tag{7.36}$$

$$b = \frac{S_i - mS_{\hat{i}}}{(N-k+1)^2} , \tag{7.37}$$

where

$$S_{\hat{i}\hat{i}} = \sum_x\sum_y\left(\hat{i}_2(x,y)\right)^2 , \tag{7.38}$$

$$S_{\hat{i}i} = \sum_x\sum_y \hat{i}_2(x,y)i_2(x,y) , \tag{7.39}$$

$$S_{\hat{i}} = \sum_x\sum_y \hat{i}_2(x,y) , \tag{7.40}$$

$$S_i = \sum_x\sum_y i_2(x,y) , \tag{7.41}$$

$$\tag{7.42}$$

and $(N-k+1)^2$ was the total number of points in the local region. Experimentally it was found that some reasonable upper and lower bounds had to be placed on m and b. This substantially reduced the probability of the minimization process getting caught in a local minimum. If the original radiometric correction was good, then

$$m \approx 1 \quad \text{and} \quad b \approx 0 . \tag{7.43}$$

4. If χ^2 was less than a minimum threshold, then the best $\hat{h}_3(x,y)$ had been found, and the process was complete. Otherwise, the next step was executed.

5. The values of χ^2 from previous iterations and Brent's method [20][123, §10.2] was used to determine the next guess for $\hat{h}_3(x,y)$. Brent's method was a technique which determined the best guess for $\hat{h}_3(x,y)$ based on up to six previous values of χ^2, and no knowledge of the slope of the function being minimized.

6. Goto step 2.

It is important to note that the value of k varied dynamically, depending on the choice of $\hat{h}_3(x,y)$ for each iteration. Therefore the area of $i_2(x,y)$, which was $N-k+1 \times N-k+1$, may also vary for each iteration. The area of $i_1(x,y)$, which was $N \times N$, had been previously fixed, according to Section 7.6. After the best $\hat{h}_3(x,y)$ was found, its location in the table of patterns yielded the estimated depth to the scene.

7.8 CALCULATING AN OPTIMIZED TABLE OF $h_3(x,y)$ PATTERNS

In Section 7.3 the geometric optics model of defocus was assumed to determine $h_1(x,y)$ and $h_2(x,y)$, and hence derive the patterns for $h_3(x,y)$. This table can be optimized to a particular lens by experimentally deriving $h_1(x,y)$ and $h_2(x,y)$.

The point spread function of defocus can be derived, by imaging a scene of an impulse function. An impulse function can be formed with a black dot on a white background, or if necessary, a sharper contrast can be obtained by placing a light source behind a black background containing a pinhole. Figure 7.12 shows a radiometrically corrected, experimental point spread function of defocus, obtained by imaging a black dot.[10] It can be seen that this experimentally obtained defocus operator is significantly different than the theoretical defocus operators shown in Chapter 3. Figure 7.12 also shows some rotational variance, which is probably due to an off-axis point source [4, p.

[10]The intensity values have been reversed (ie higher numbers represent darker intensity), to better illustrate the function.

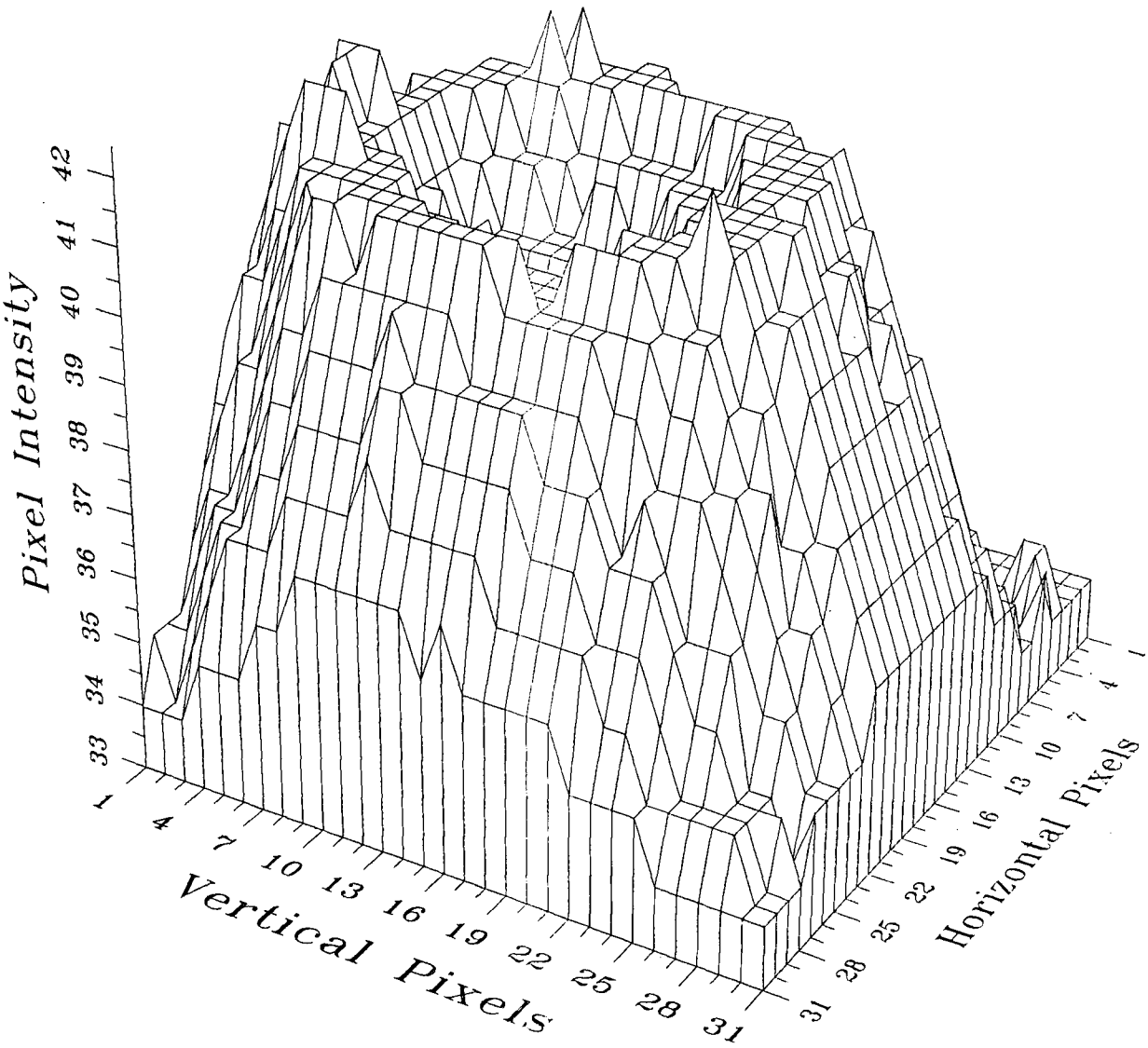


Figure 7.12: Example of a defocus operator formed with a lens having aberrations.

76]. This rotational variance is slight and will be ignored in the remaining analysis.

Since the method for deriving $h_3(x, y)$, as described in Section 7.3, uses integrated functions $h_{1y}(y)$ and $h_{2y}(y)$ in one dimension, it is convenient to derive these directly. They can be obtained by imaging a scene $s(y)$ which has no variation in x. This can be shown by working out the convolution equation

$$\begin{aligned} s(y) \otimes h_1(x,y) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} s(y-\eta)h_1(\xi,\eta)d\xi d\eta \text{ and} \\ s(y) \otimes h_1(x,y) &= \int_{-\infty}^{\infty} s(y-\eta)\int_{-\infty}^{\infty} h_1(\xi,\eta)d\xi d\eta \ . \end{aligned} \tag{7.44}$$

Using the definition of $h_{1y}(y)$ from (7.16), yields

$$\begin{aligned} s(y) \otimes h_1(x,y) &= \int_{-\infty}^{\infty} s(y-\eta)h_{1y}(\eta)d\eta \text{ and} \\ s(y) \otimes h_1(x,y) &= s(y) \otimes h_{1y}(y) \ . \end{aligned} \tag{7.45}$$

Therefore, convolution of the defocus operator $h_1(x, y)$, with a one-dimensional scene $s(y)$, is identical to convolution of the same scene with an integrated defocus operator $h_{1y}(y)$.

Furthermore, if $s(y)$ is an ideal black to white transition with amplitude of β, then $h_{1y}(y)$ can be isolated by simple differentiation [97]. If

$$i_1(y) = h_{1y}(y) \otimes s(y) \ , \tag{7.46}$$

and both sides are differentiated, then

$$\frac{d}{dy} i_1(y) = \frac{d}{dy}\left(h_{1y}(y) \otimes s(y)\right) \ . \tag{7.47}$$

But the right hand side can be rewritten [95, p. 416],

$$\frac{d}{dy} i_1(y) = h_{1y}(y) \otimes \left(\frac{d}{dy} s(y)\right) \ . \tag{7.48}$$

Since the scene is a black to white transition of amplitude β, its derivative is simply β. Therefore

$$h_{1y}(y) = \frac{1}{\beta}\left(\frac{d}{dy} i_1(y)\right) . \tag{7.49}$$

This means that $h_{1y}(y)$ can be obtained directly, by taking the derivative of a defocused black to white transition. A similar reasoning also applies to $h_{2y}(y)$.

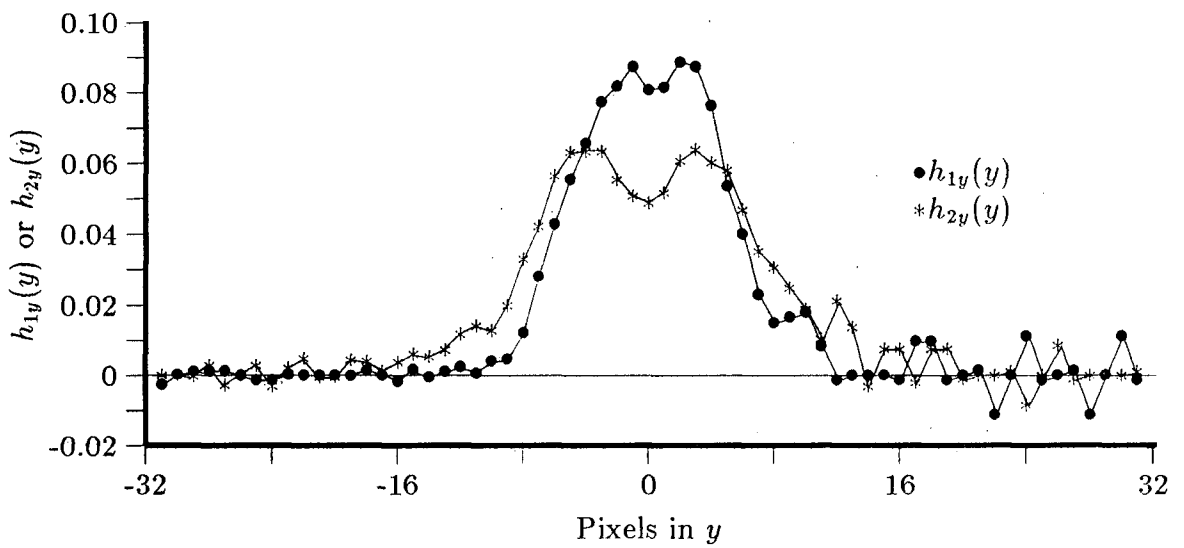


Figure 7.13: Experimental examples of functions $h_{1y}(y)$ and $h_{2y}(y)$ with the optical parameters given in the text.

Examples of experimentally obtained $h_{1y}(y)$ and $h_{2y}(y)$ are shown in Figure 7.13. They can be directly compared to Figure 7.3, the theoretically predicted defocus operators. This shows that the geometric optics model is a close approximation, however it doesn't completely model the actual defocus operators. The actual defocus operators were modelled with a trapezoidal shape as shown in Figure 7.14.

A number of optimized patterns for $h_{1y}(y)$ and $h_{2y}(y)$ were obtained experimentally, and modelled with a trapezoidal shape, using the same optical parameters that the

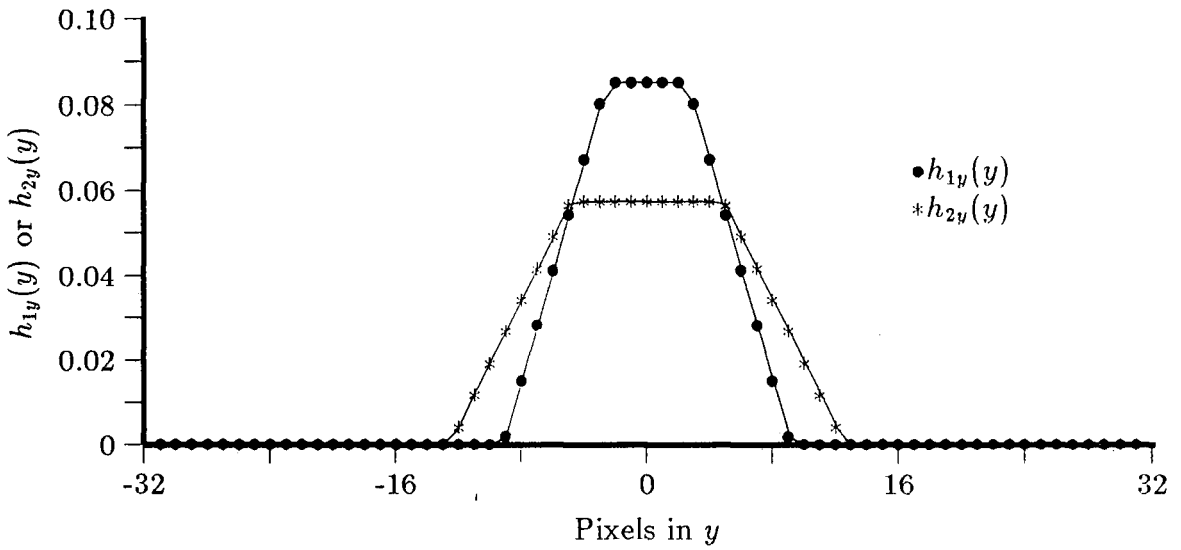


Figure 7.14: Models of the experimental functions $h_{1y}(y)$ and $h_{2y}(y)$ shown in the previous Figure.

theoretical patterns were calculated in Section 7.3. That is $f_1 = 2.0$, $f_2 = 1.3$, the point of perfect focus was set to 1 *meter*, and the distance to the scene ranged from $D_o = 0.80$ to $D_o = 0.95$ *meters*. A total of 16 patterns each for $h_{1y}(y)$ and $h_{2y}(y)$ were derived, corresponding to 1 *cm* increments. From these, trapezoidal models were fitted, and an optimized table of $h_3(x, y)$ patterns was calculated using the regularization method outlined in Section 7.3.

An example $\mathbf{h_3}$ pattern for $D_o = 0.85$ *meters* is shown in Figure 7.15. This can be directly compared to the theoretical $\mathbf{h_3}$ shown in Figure 7.6. It can be seen that the patterns are quite different, since the geometric optics model does not adequately model the defocus operator of the camera used in the experiments.

The optimized table of $h_3(x, y)$ patterns, along with the table of $h_3(x, y)$ patterns derived assuming geometric optics will be used in the next chapter. Chapter 8 will outline the experimental results of implementing the new method of depth from focus

described in this chapter.

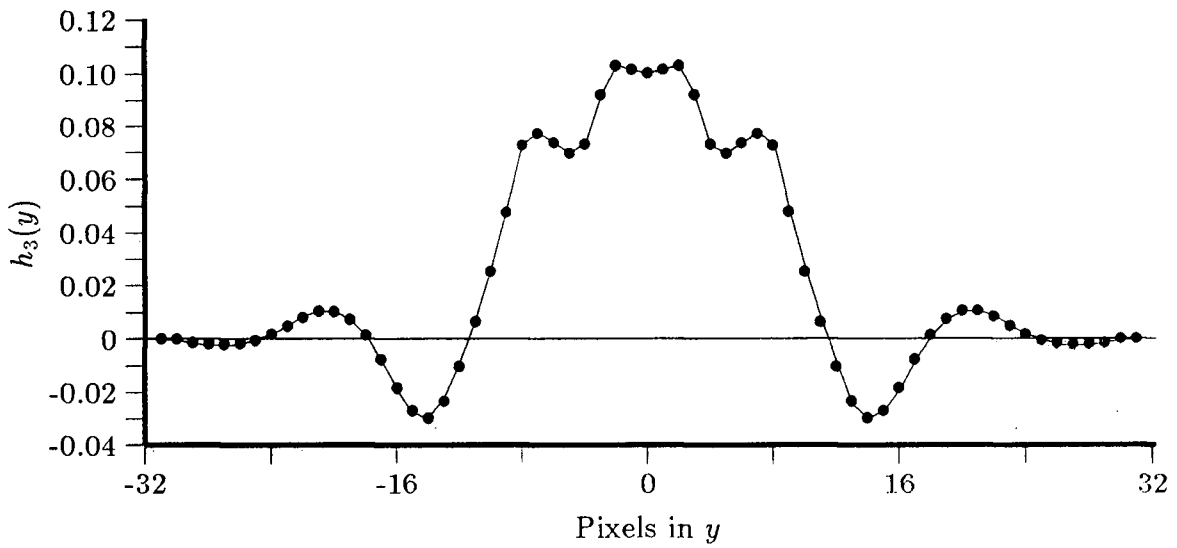


Figure 7.15: Example of experimental $\mathbf{h}_3$ solved by regularization with smoothing and suppression of ends.

7.9 CONCLUSIONS

This chapter presented the steps necessary to implement the author's new iterative matrix solution for calculating depth from focus. Two tables of $h_3(x, y)$ patterns were calculated:

- One assumed geometric optics; and
- One was based on the experimentally derived defocus operator.

Differences in the tables showed that the geometric optics model is a reasonable approximation, however it does not exactly model the experimentally derived defocus operator.

The implementation of the iterative matrix solution can be summarized by the following steps:

1. Acquire two images i_1 and i_2 of the same scene with different camera optical parameters, such as a change in aperture.

2. Radiometrically correct images i_1 and i_2.

3. Locate local regions at edges in i_1.

4. Select a local region.

5. Iteratively select $\hat{h}_3(x, y)$ from the table, to find the one which minimizes

$$\sum_{x=0}^{N-k} \sum_{y=0}^{N-k} \left[i_1(x, y) \; [\otimes] \; \hat{h}_3(x, y) - i_2(x, y) \right]^2 = \text{minimum} \; . \tag{7.50}$$

6. The position of $\hat{h}_3(x, y)$ in the table yields the distance to the local region in the scene.

7. Goto step 4.

Chapter 8

EXPERIMENTAL RESULTS

8.1 INTRODUCTION

This chapter presents the experimental results using the author's new iterative matrix based method for calculating depth from focus. These results were obtained by following the steps detailed in Chapter 7. Conclusions and possibilities for further research work are given in Chapter 9.

8.2 OVERVIEW

Using four different experimental scenes, this chapter details the results of solving the depth from focus problem using the author's new iterative matrix based method as well as constrained inverse filtering. Each scene contains features at known distances from the camera so that explicit accuracy results will be given. The scenes processed are:

1. A flat sheet of paper with various letters of the alphabet. The paper is sloped so that the distance to the camera ranges linearly from 0.95 to 0.80 *meters*.

2. Two blocks which show a shadow edge, an occlusion edge, and a printed grey edge.

3. A flat sheet of paper with printed, grey, ramp edges of increasing width.

4. A sloped piece of paper with equally spaced, vertical, black bars which produces a phase inversion in the acquired images.

The experimental equipment used for these tests has already been described in Section 7.4. The focal length of the camera lens is 50 *mm*, and it is adjusted so that scene features which are one *meter* from the camera, are in perfect focus. Two images are acquired with apertures set at $f_1 = 2.0$, and $f_2 = 1.3$. The images are radiometrically corrected according to Section 7.5. As described in Section 7.6, edges are found in the sharper image ($f = 2.0$), which identifies the local regions used for processing of depth information.

Experimental results of obtaining depth from focus using images obtained with two different apertures, where one is not a pinhole, have not previously been reported in the literature.

8.3 SCENE WITH TWO-DIMENSIONAL FEATURES

The first scene imaged was a flat sheet of paper with various letters of the alphabet. The paper was arranged so that the distance to the camera ranged linearly from 0.95 *meters* at the left hand side of the image, to 0.80 *meters* at the right hand side of the image. The acquired images are shown in Figures 8.1 and 8.2.

8.3.1 Constrained Inverse Filtering

The first solution to the depth from focus problem is obtained using constrained inverse filtering, as described in Section 6.7.3. A total of 242 local regions, of size 64×64 pixels were processed,[1] and one depth point was calculated for each local region. A mesh plot of the results is shown in Figure 8.3. The dashed line on the side of the plot shows the expected depth values for the scene being imaged.

[1] The problem of providing a limited circular window for the quadratic, mentioned in Section 6.5.2, is avoided by calculating the *expected* width of the quadratic based on known depth values. Although this is not possible in practice (since the depth values are not known *a priori*), it was done in these experiments to obtain a *best case* performance of the constrained inverse filtering method.

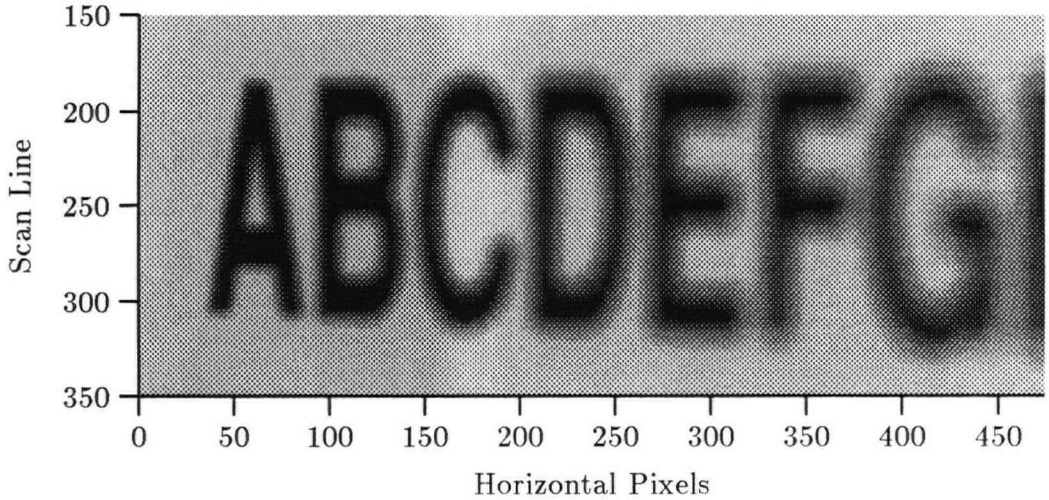


Figure 8.1: Slanted image of alphabet at $f_1 = 2.0$.

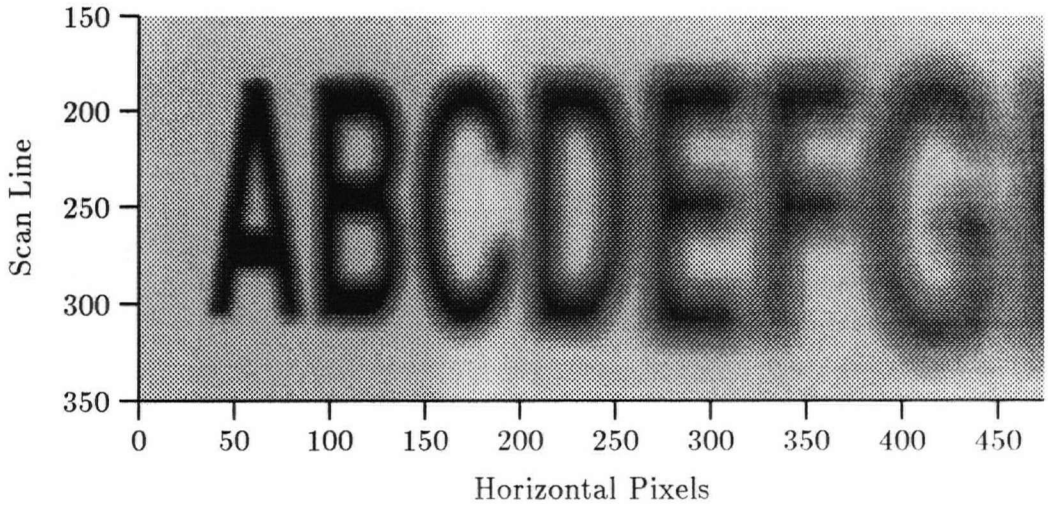


Figure 8.2: Slanted image of alphabet at $f_2 = 1.3$.

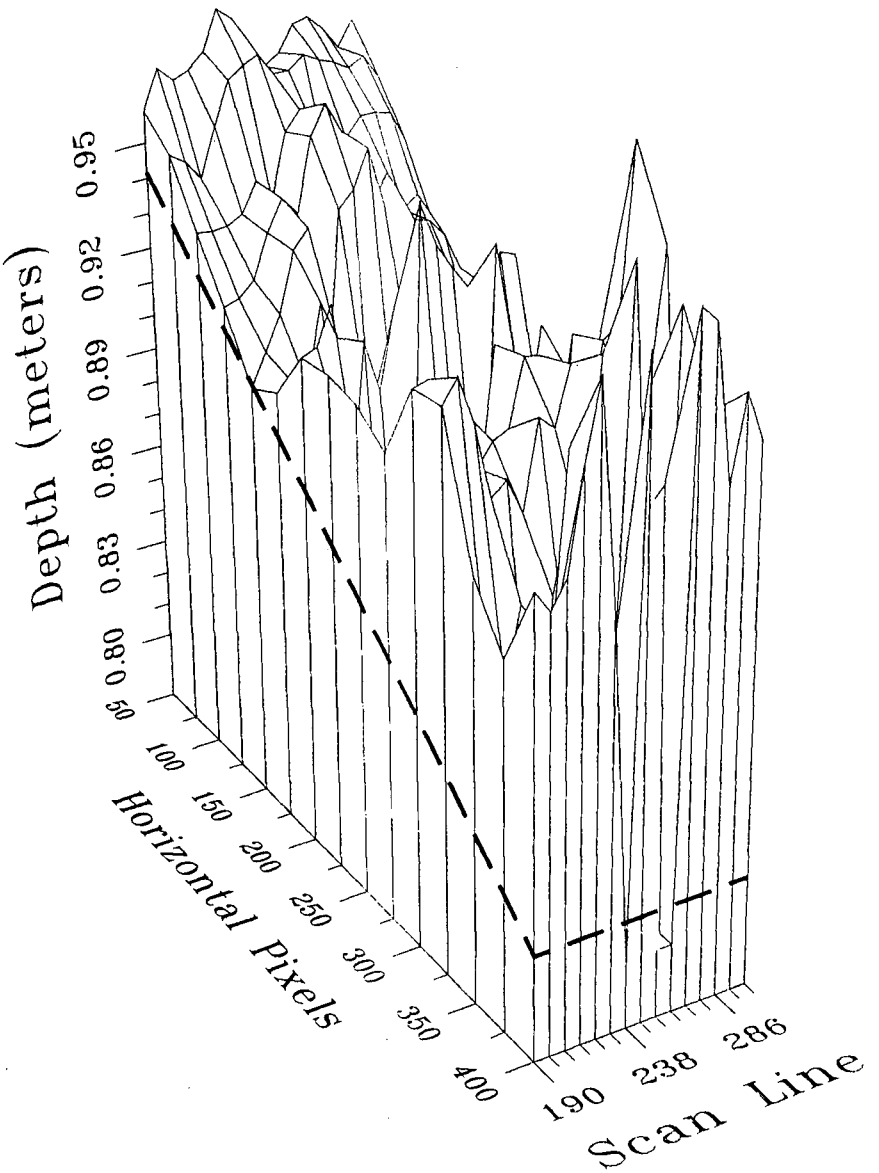


Figure 8.3: Mesh plot of calculated depth values from using constrained inverse filtering on the "alphabet" images. The dashed line shows the expected depth values.

The experimental results are consistent with the theoretical observations of Section 6.5.3. The most consistently accurate depth values in the plot are found on the left hand side, for both large and small scan lines. In the images of Figures 8.1 and 8.2, this corresponds to the bottom and top of the letters "A" and "B". Here the errors associated with windowing and border effects are minimal. Windowing errors are small because a majority of the outside perimeter of the local areas, are about the same intensity. Border effects are also slight, because the width of the defocus operator is relatively small.

The least accurate depth values are on the right hand side of the plot, towards the center. Here windowing effects are large, since a local area can likely have a dark intensity on one side and a light intensity on the other side. The border effects are also large, since the defocus operators are wide and edges from adjacent local areas "bleed" into each other.

Using an 8 *MHz* IBM PC with *no* hardware acceleration or math coprocessor, and floating pointing point arithmetic implemented in Pascal, the average computational time was about 4.4 minutes per depth point.[2] The RMS errors over the scene are shown in Table 8.1. It can be clearly seen that the error increases for distances closer to the camera, which is also increasing defocus. The average RMS error for the scene was 42% over the *expected range*, and 6.8% in terms of *distance from the camera*.

[2]Benchmarks performed by the author indicate that this could be reduced to 15 seconds, by using integer arithmetic in machine language, although it was not implemented. This number allows a fair comparison to be made with the processing times taken by the iterative matrix based method in Section 8.3.2.

Expected depth (*meters*)	RMS error over expected range (%)	RMS error in distance to camera (%)	Number of points
$0.95 \geq D_o > 0.90$	23.7	3.88	89
$0.90 \geq D_o > 0.85$	40.5	6.78	92
$0.85 \geq D_o \geq 0.80$	59.9	9.57	61
$0.95 \geq D_o \geq 0.80$	41.7	6.79	242

Table 8.1: RMS errors in depth values from using constrained inverse filtering on the "alphabet" images.

8.3.2 Iterative Matrix Solution

Using geometric optics

The table of $h_3(x, y)$ values calculated in Section 7.3, assuming geometric optics, was used for this test. Using the steps described in Section 7.7, 242 depth points were calculated with the author's new iterative matrix based solution.[3] A mesh plot of the results is shown in Figure 8.4. Smoothing of the data was not required.[4]

The iterative process of finding the best $\hat{h}_3(x, y)$, described in Section 7.7, converged after about six iterations. The algorithm always converged, for this scene as well as all the others in this chapter. The best results were obtained when the parameters m and b, described in Section 7.7, were constrained to the limits

$$0.98 \leq \quad m \quad \leq 1.02 \text{ and} \tag{8.1}$$

$$-5.00 \leq \quad b \quad \leq 5.00\,. \tag{8.2}$$

Using an 8 *MHz* IBM PC with *no* hardware acceleration, the average computational

[3]The center of the local areas were located at the same point as the constrained inverse filtering method, however for the iterative matrix based method, the size of $i_1(x, y)$ varied according to the width of the edges in the scene, and the size of $i_2(x, y)$ was additionally determined by the size of $\hat{h}_3(x, y)$. The average size of $i_1(x, y)$ was about 64×64 pixels.

[4]The mesh program required a regular grid to be calculated. Grid values close to data points were simply assigned the value of the data point, and other grid points were interpolated by minimizing the curvature of the grid. This method should yield a faithful representation of the original data.

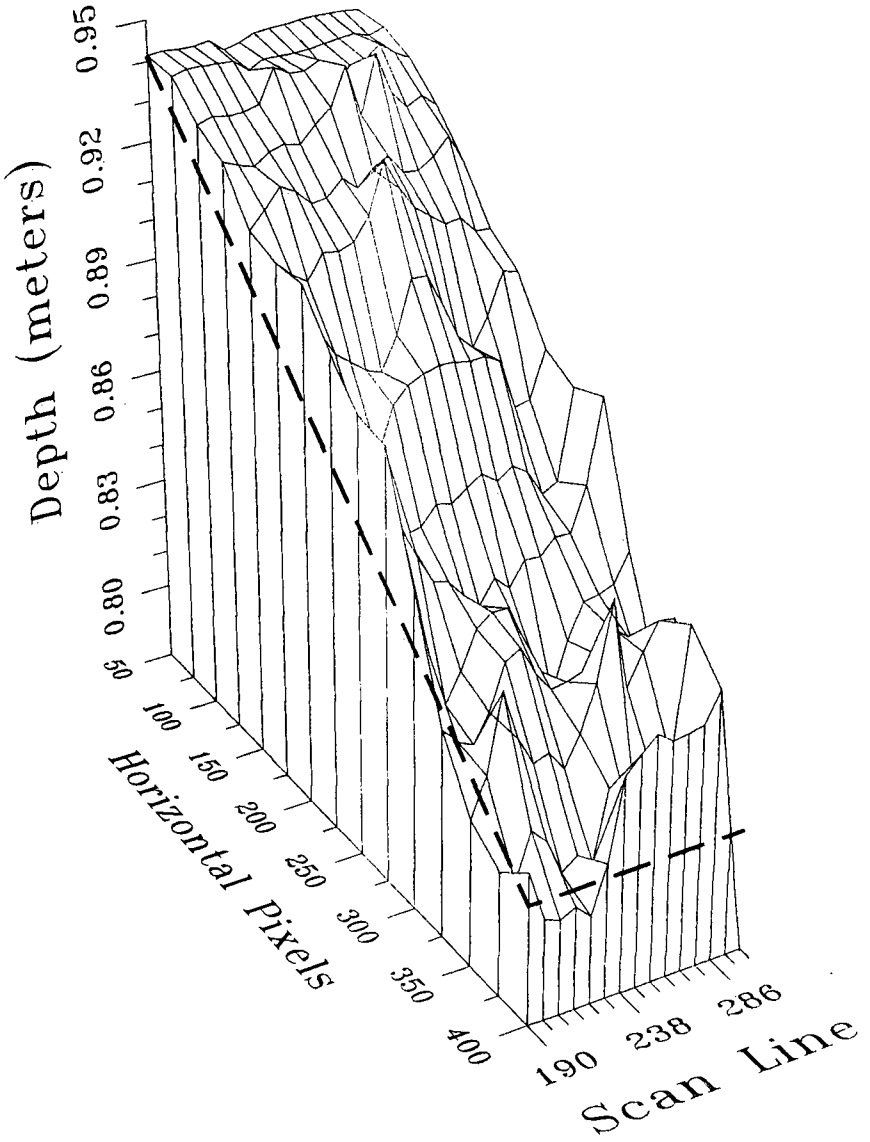


Figure 8.4: Mesh plot of calculated depth values from using iterative matrix solution on the "alphabet" images, and assuming the geometric optics model of defocus. The dashed line shows the expected depth values.

time per depth point, was 3.3 minutes. The RMS errors over the scene are shown in Table 8.2. The RMS error over the *expected range* was 10%, and it was 1.7% in terms

Expected depth (*meters*)	RMS error over expected range (%)	RMS error in distance to camera (%)	Number of points
$0.95 \geq D_o > 0.90$	10.67	1.71	89
$0.90 \geq D_o > 0.85$	8.67	1.45	92
$0.85 \geq D_o \geq 0.80$	10.69	1.89	61
$0.95 \geq D_o \geq 0.80$	9.96	1.67	242

Table 8.2: RMS errors in depth values from using iterative matrix solution on the "alphabet" images, and assuming the geometric optics model of defocus.

of *distance from the camera*. These values represent a substantial improvement over the errors obtained using the constrained inverse filtering technique.

Using (3.15) and (3.17), the different values of R_{1p}, the radius of $h_1(x, y)$ and R_{2p}, the radius of $h_2(x, y)$ are calculated and shown in Table 8.3. For distances close to

D_o (m)	R_{1p} (pixels)	R_{2p} (pixels)
0.95	2.1	3.2
0.80	9.9	15.2

Table 8.3: Values of R_{1p} and R_{2p} for the range of distances D_o in the experimental scene.

0.95 m, it can be seen that the difference between R_{1p} and R_{2p} is slight, yet by looking at Figure 8.4 the results are still comparatively good in this region. Therefore it can be seen that the algorithm performs well, even though the two images were acquired with a difference of only one *f*-stop.

Using experimental defocus operator

Using the table of $h_3(x, y)$ values found in Section 7.8, another set of depth values was calculated from the same images shown in Figures 8.1 and 8.2. A mesh plot of the results is shown in Figure 8.5.

The RMS errors over the scene are shown in Table 8.4. The table shows a progres-

Expected depth (*meters*)	RMS error over expected range (%)	RMS error in distance to camera (%)	Number of points
$0.95 \geq D_o > 0.90$	4.89	0.79	89
$0.90 \geq D_o > 0.85$	7.85	1.36	92
$0.85 \geq D_o \geq 0.80$	9.57	1.72	61
$0.95 \geq D_o \geq 0.80$	7.44	1.29	242

Table 8.4: RMS errors in depth values from using iterative matrix solution on the "alphabet" images, and an experimentally derived defocus operator.

sion of increasing error for distances closer to the camera, where the defocus operator is larger. This seems intuitive, since a larger defocus operator limits the bandwidth of information about the images. The larger errors for distances close to 0.80 m seem to indicate that the 64×64 region used to calculate $h_3(x, y)$ in Section 7.3.2, is probably too small. The RMS error over the *expected range* was 7.4%, and it was 1.3% in terms of *distance from the camera*. This is a reduction, from the error found in the last section, while assuming the geometric optics model of defocus.

8.4 SCENE WITH VARIOUS EDGES

Although the images of the alphabet contained an assortment of two-dimensional features, they had only sharp black to white printed edges. The next scene contains three other types of edges:

1. A soft edge caused by a shadow (penumbra);

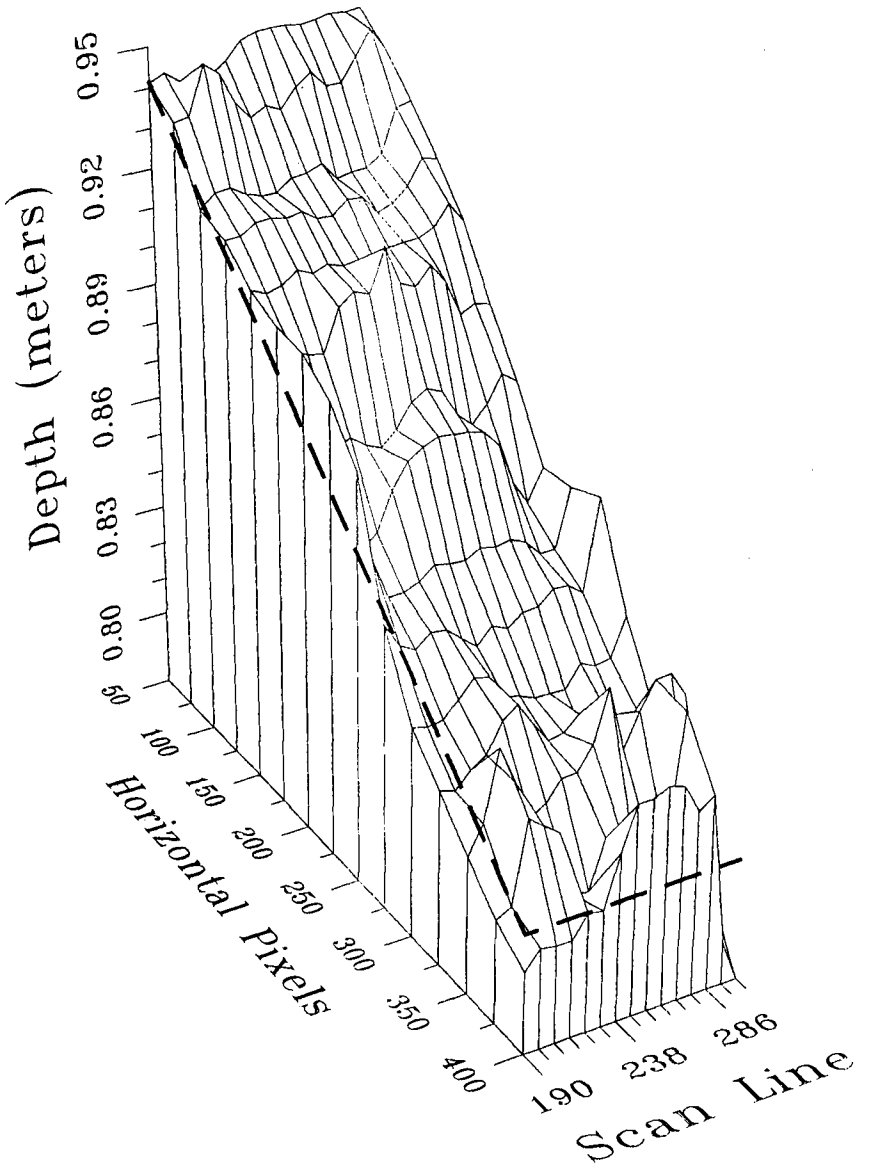


Figure 8.5: Mesh plot of calculated depth values from using iterative matrix solution on the "alphabet" images, and an experimentally derived defocus operator. The dashed line shows the expected depth values.

2. An occlusion edge; and

3. A printed, grey, ramp edge.

Conditions under which depth from focus will not work, and features that will yield inaccurate depth values have already been discussed in Sections 2.5 and 2.6, and are deliberately excluded from the scene. They are

- Featureless areas;
- Specularity and mirrored surfaces;
- Scene motion;
- Features outside of the active range;
- Excessive image noise;
- Semi-transparent surfaces, such as dirty windows; and
- Illumination causing camera saturation or black-out.

To construct the three edges mentioned at the beginning of this section, a top view of the scene geometry is shown in Figure 8.6. The resulting images are shown in Figure 8.7 and 8.8.

8.4.1 Constrained Inverse Filtering

Using constrained inverse filtering, eleven local regions on each of the three edges were processed. The errors in the results are shown in Table 8.5. These errors are somewhat greater than the errors found in Section 8.3.1, when processing the "alphabet" images. The occlusion edge yielded invalid results at all local regions tested.[5]

[5] Referring to the method described in Section 6.7.3, the modelled quadratic was up-turned, resulting in invalid depth points.

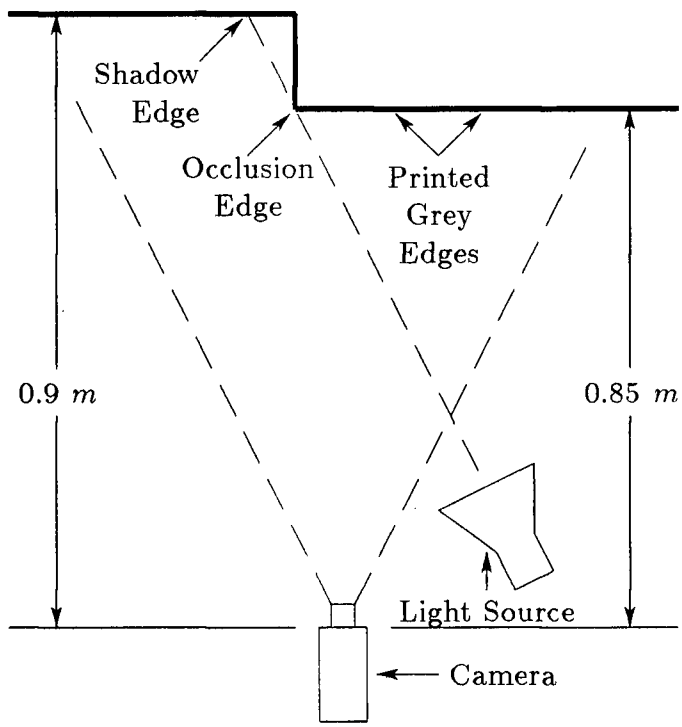


Figure 8.6: Top view of scene geometry to obtain images with various edges.

Scene edges	RMS error over expected range (%)	RMS error in distance to camera (%)	Number of points
Shadow edge	34.8	5.86	11
Occlusion edge	—	—	11
Printed grey edge	71.3	11.16	11
Total	49.7	8.1	33

Table 8.5: RMS errors in depth values from using constrained inverse filtering on the images of various edges.

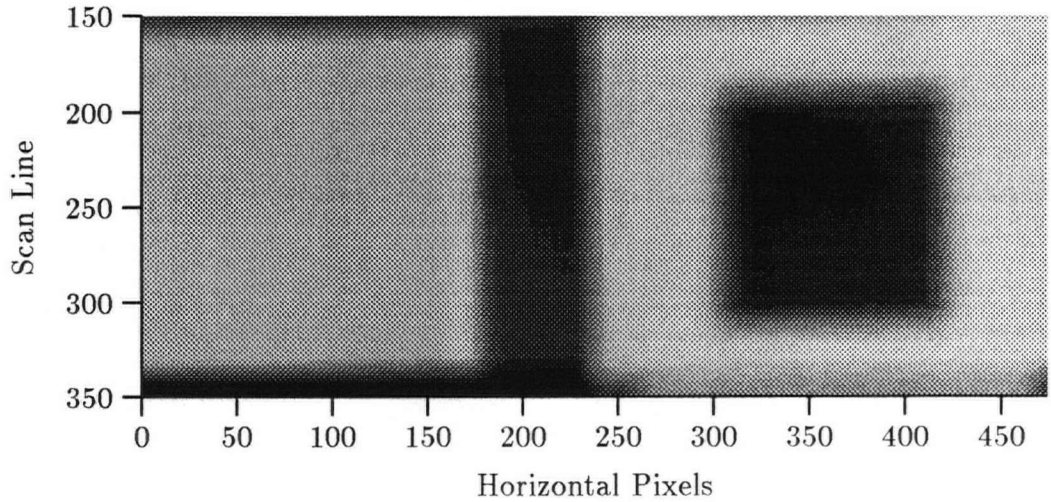


Figure 8.7: Image with various edges at $f_1 = 2.0$.

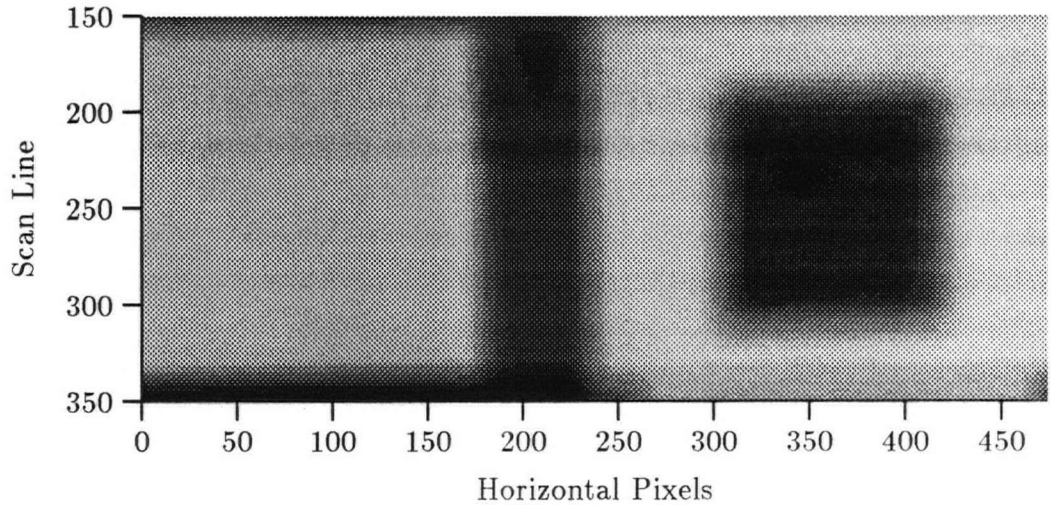


Figure 8.8: Image with various edges at $f_2 = 1.3$.

8.4.2 Iterative Matrix Solution

The iterative matrix solution was also implemented on the images of various edges. For this test and all successive tests in this chapter, the experimentally derived table of $h_3(x, y)$ was used, since Section 8.3.2 showed that it yielded superior results to the table formed assuming geometric optics. The results are shown in Table 8.6. The

Scene Feature	RMS error over expected range (%)	RMS error in distance to camera (%)	Number of points
Shadow edge	2.99	0.50	11
Occlusion	2.83	0.50	11
Printed grey edge	4.90	0.86	11
Total	3.70	0.64	33

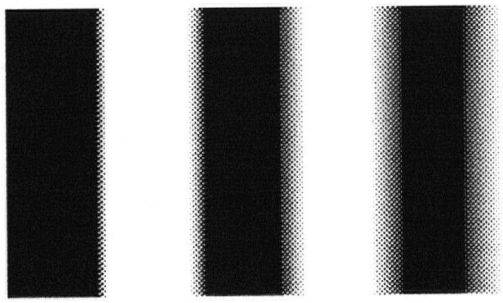
Table 8.6: RMS errors in depth values from using iterative matrix solution on the images of various edges.

errors shown are even lower than those found while processing the "alphabet" images. This seems somewhat counter-intuitive, since the "alphabet" images had sharp edges, whereas all three of the edges in the current images are softer. The answer is found in the next section, which will consider a series of grey, ramped edges of various widths.

8.5 SCENE WITH RAMPED EDGES

The third scene imaged, is a flat sheet of paper with printed grey edges of increasing width. The scene is shown in Figure 8.9.[6] The images acquired of this scene are shown in Figure 8.10 and 8.11.

[6]Note the sharp white to black transition has a measured ramp edge of four pixels because of bandwidth limitations in the camera and frame grabber board.



4 8 12 16 20 24

Width of edge in pixels, when in focus

Figure 8.9: Actual scene of ramped edges.

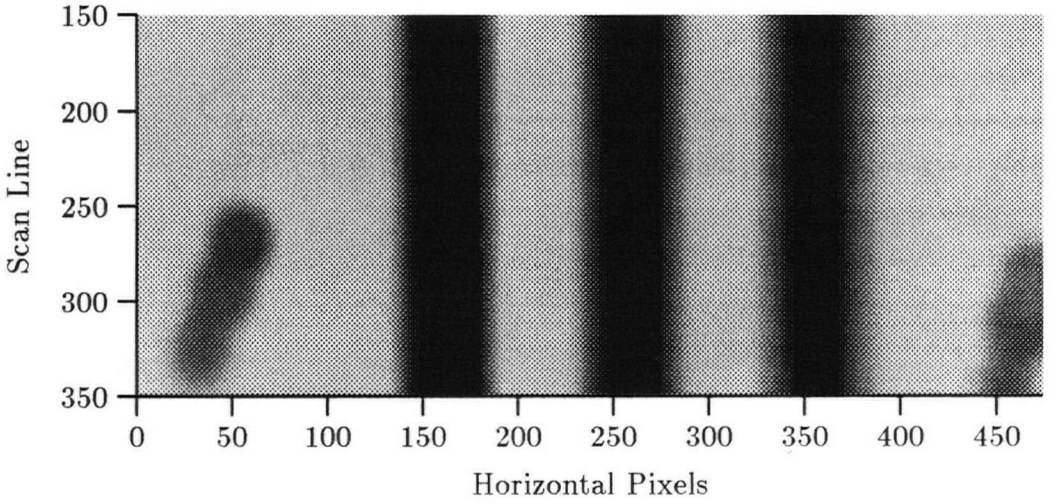


Figure 8.10: Image with various ramped edges at $f_1 = 2.0$.

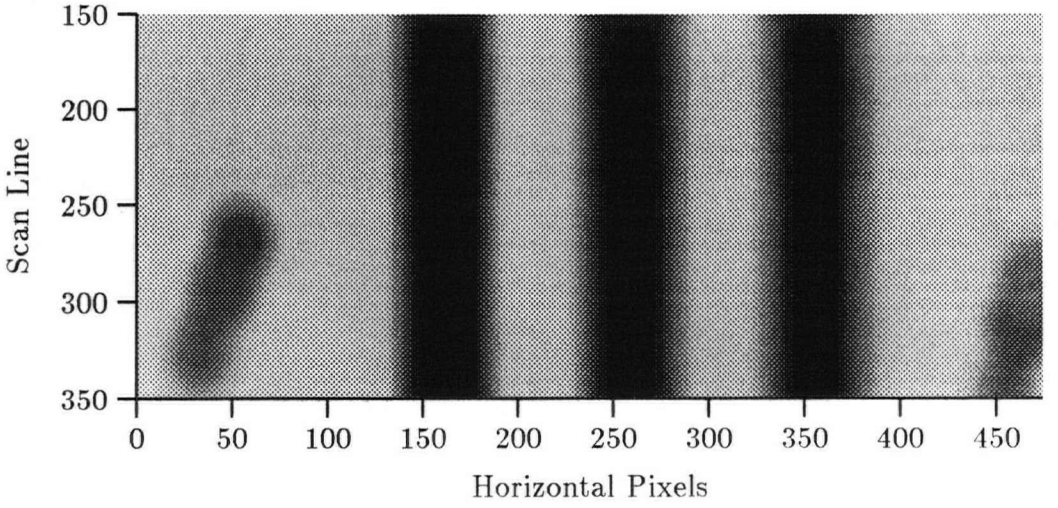


Figure 8.11: Image with various ramped edges at $f_2 = 1.3$.

8.5.1 Constrained Inverse Filtering

Using constrained inverse filtering, eleven local regions on each ramp edge were processed. The errors in the results are shown in Table 8.7. These errors are also shown

Ramp edge width (pixels)	RMS error over expected range (%)	RMS error in distance to camera (%)	Number of points
4	42.4	6.51	11
8	38.8	6.07	11
12	25.0	3.99	11
16	56.0	9.09	11
20	24.0	3.85	11
24	53.5	9.04	11
Total	41.8	6.76	66

Table 8.7: RMS errors in depth values from using constrained inverse filtering on the images of ramp edges.

graphed in Figure 8.12. The error is expected to rise as the ramp in the grey edge becomes wider, since the difference between the two acquired images becomes smaller. However, the results are not consistent enough to draw any firm conclusions.

8.5.2 Iterative Matrix Solution

The iterative matrix solution was also implemented on the images of the ramp edges. The results are shown in Table 8.8 and graphed in Figure 8.13. The graph shows a result which may not be intuitively obvious. That is, the minimum error is achieved with ramp edges of about 8 pixels, and not with sharp black to white transitions. The reason for this can be found in the example given in Section 6.6. Here it was seen that sharp edges require $h_3(x, y)$ to have an infinite extent, whereas the $h_3(x, y)$ used for the experiments has been smoothed and windowed, as explained in Section 7.3.2.

After the minimum is reached, the error continues to rise as the width of the ramp

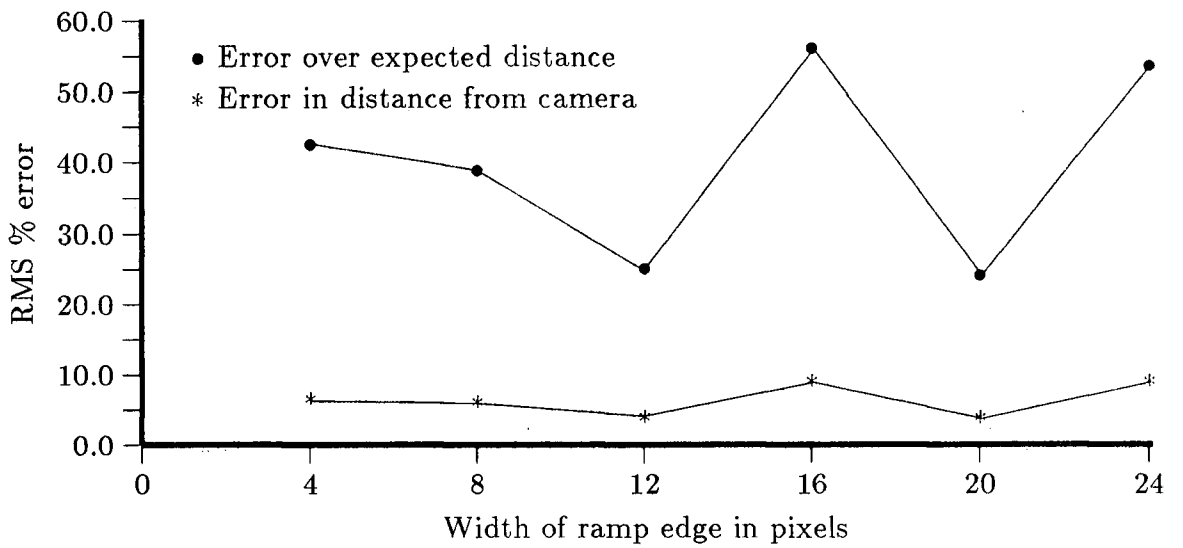


Figure 8.12: RMS errors in depth values from using constrained inverse filtering on the images of ramp edges.

Ramp edge width (pixels)	RMS error over expected range (%)	RMS error in distance to camera (%)	Number of points
4	5.75	0.97	11
8	3.10	0.52	11
12	3.22	0.54	11
16	5.89	0.99	11
20	7.74	1.31	11
24	13.68	2.34	11
Total	7.47	1.27	66

Table 8.8: RMS errors in depth values from using iterative matrix solution on the images of ramp edges.

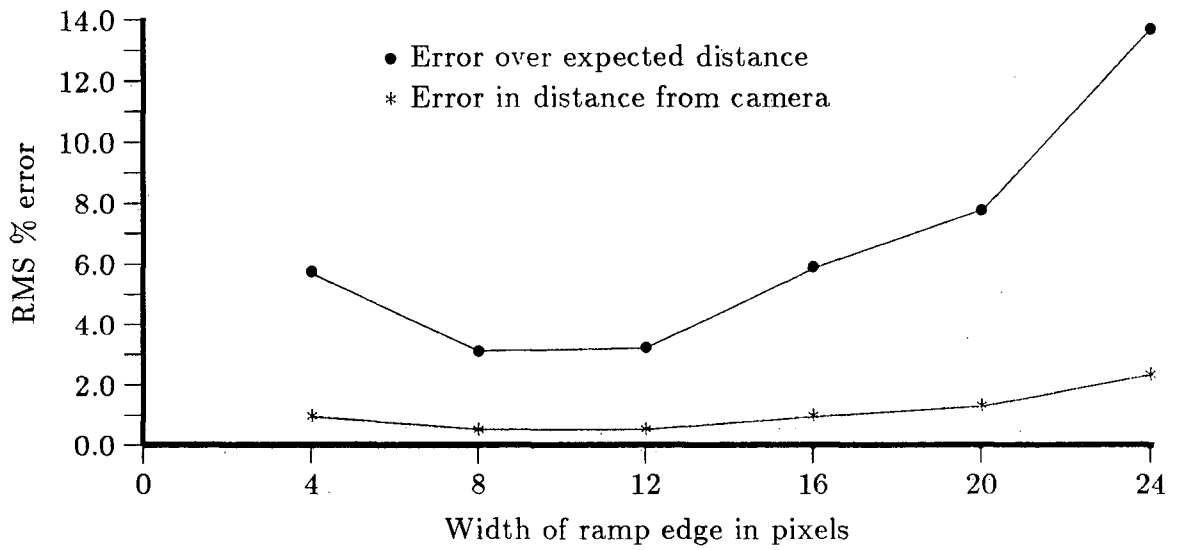


Figure 8.13: RMS errors in depth values from using iterative matrix solution on the images of ramp edges.

increases, which is the expected result.

8.6 PHASE INVERSION

Image restoration is a difficult problem, but image restoration after phase reversal is an even more difficult problem [153]. As previously shown in Figure 3.4, the higher spatial frequencies of the optical transfer function for defocus can swing below zero. This implies a phase reversal in the spatial domain [51, §6-4] when features are close together. The effect of this phase reversal is illustrated by Figure 8.14. The scene viewed is a sheet of paper with a series of equally spaced, vertical, black bars. The paper was arranged in the manner described in Section 8.3, so that the distance to the camera varied linearly from 0.95 *meters* on the left hand side, to 0.80 *meters* on the right hand side.

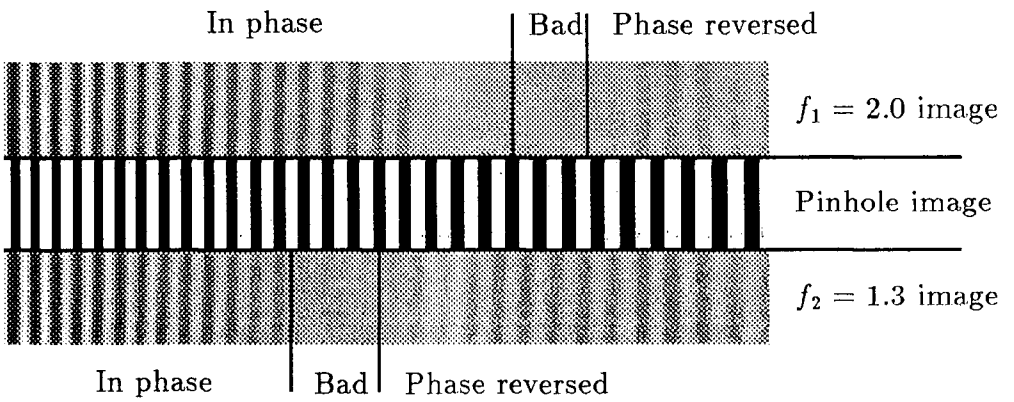


Figure 8.14: The center strip shows a pinhole image of the scene, while the upper strip is the image acquired at $f_1 = 2.0$ and the lower strip is the image acquired at $f_2 = 1.3$.

The center strip of Figure 8.14 shows the scene in perspective, as seen through a pinhole camera. The top strip is an image blurred at $f_1 = 2.0$ and the bottom strip is an image blurred at $f_2 = 1.3$. It can be seen that the bars on the right hand side of both blurred images are out of phase with the unblurred scene. When the two blurred images are compared to each other, they are in phase on the left, out of phase in the center, and in phase again on the right hand side.

8.6.1 Constrained Inverse Filtering

The constrained inverse filtering technique was attempted on these images, with the depth results of points along a horizontal line shown in Figure 8.15, and the errors shown in Table 8.9. The graph shows constrained inverse filtering working reasonably well before the phase reversal, however after one image is phase reversed, the results are quite unstable.

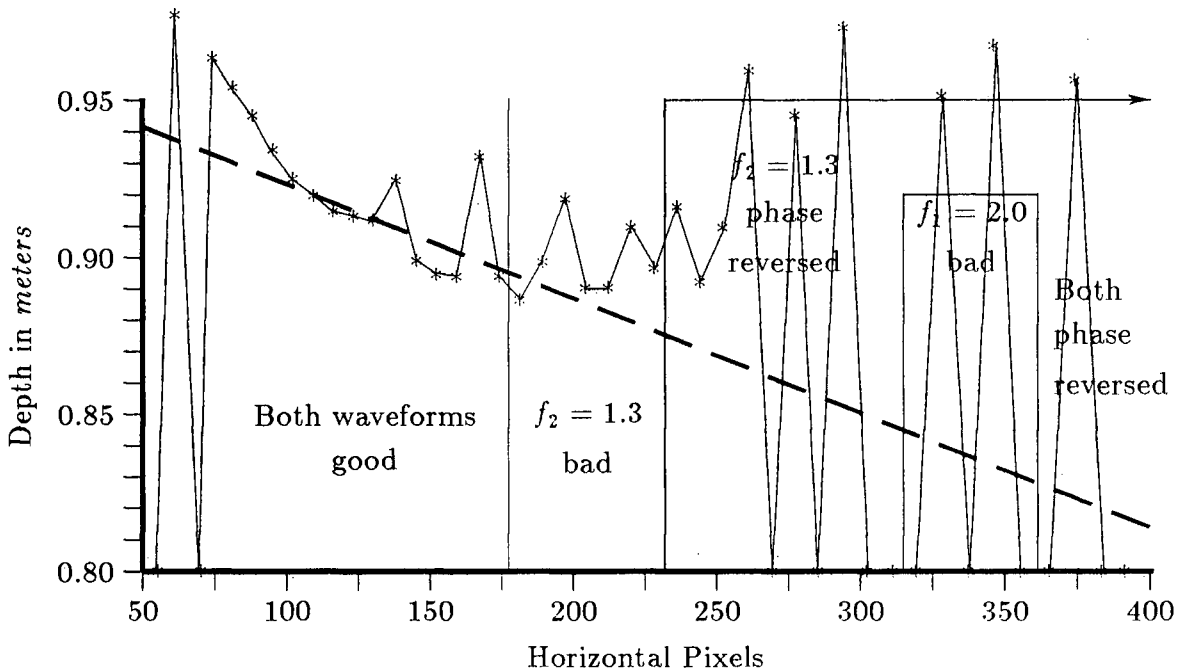


Figure 8.15: Depth results from using constrained inverse filtering on the images of vertical bars with phase reversal. The dashed line shows the expected depth values.

Area	RMS error over expected range (%)	RMS error in distance to camera (%)	Number of points
Both images good	32.4	5.97	18
f_2 bad	12.4	2.05	7
f_2 phase reversed	45.3	7.49	10
f_1 bad	55.2	8.85	5
Both phase reversed	46.0	7.33	4
Total	38.2	6.46	44

Table 8.9: RMS errors from using constrained inverse filtering on the images of vertical bars with phase reversal.

8.6.2 Iterative Matrix Solution

The author's iterative matrix solution was also applied to the same horizontal line of points from both blurred images, with the range results shown in Figure 8.16 and the errors shown in Table 8.10. The graph shows excellent results when both waveforms

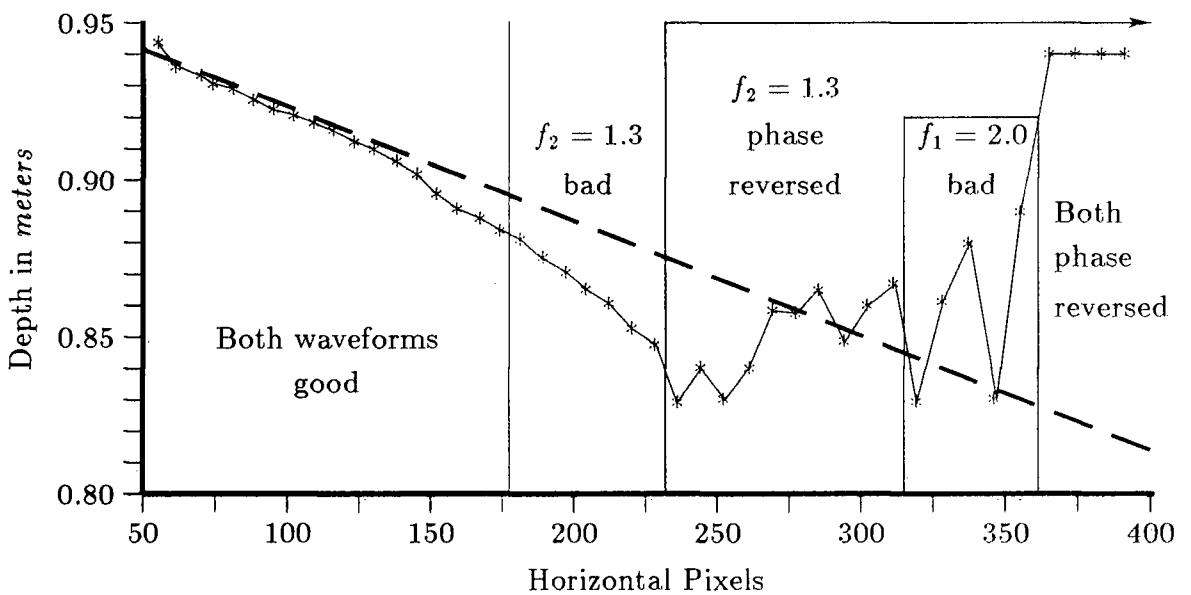


Figure 8.16: Depth results from using iterative matrix solution on the images of vertical bars with phase reversal. The dashed line shows the expected depth values.

are good, with some degradation when the $f_2 = 1.3$ image fades out. During phase reversal of one image, the results are still fair. The algorithm does fail, however, when both images are phase reversed, since this condition closely mimics a small amount of defocus. This test shows that the author's iterative matrix solution is robust enough to handle phase inversion of *one* image relative to the other.

Area	RMS error over expected range (%)	RMS error in distance to camera (%)	Number of points
Both images good	3.83	0.64	18
f_2 bad	14.2	2.49	7
f_2 phase reversed	15.7	2.82	10
f_1 bad	23.1	3.93	5
Both phase reversed	78.8	12.6	4
Total	26.8	4.37	44

Table 8.10: RMS errors from using iterative matrix solution on the images of vertical bars with phase reversal.

8.7 CONCLUSIONS

The following four scenes were processed with the author's iterative matrix solution and constrained inverse filtering:

1. A sloped sheet of paper with various letters of the alphabet.
2. Two blocks which show a shadow edge, an occlusion edge, and a printed grey edge.
3. A flat sheet of paper with printed, grey, ramp edges of increasing width.
4. A sloped piece of paper with equally spaced, vertical, black bars which produces a phase inversion in the acquired images.

On all of the four experimental scenes processed, the iterative matrix solution was consistently more accurate than constrained inverse filtering. The following conclusions can also be made about the iterative matrix solution:

1. For the camera lens employed, geometric optics provides a good first approximation;

2. The most accurate depth results can be achieved when the actual defocus operator is experimentally measured and modelled;

3. The accuracy of the algorithm increases as the defocus operators become smaller, and the images are less blurred.

4. The algorithm is most accurate when the scene contains edges with a small ramped, grey transition area.

5. The best features processed, had an RMS error of 2.8% in terms of *expected range*, and 0.5% in terms of *distance from the camera*; and

6. The algorithm is able to cope with phase reversal in one image.

Experimental results of obtaining depth from focus using images obtained with two different apertures, where one is not a pinhole, have not previously been reported in the literature. However, the author's results still compare favorably with Pentland's measured error of 2.5%, where one image was acquired with a pinhole aperture.

Chapter 9

SUMMARY, CONCLUSIONS AND FURTHER WORK

9.1 OVERVIEW

This chapter is divided into three sections. Section 9.2 gives a summary of Chapters 2 to 4 which describe the current state of depth from focus. Section 9.3 presents the conclusions based on the author's work in Chapters 5 to 8. These conclusions summarize the original contribution of the thesis. Opportunities for further work are described in Section 9.4.

9.2 SUMMARY OF DEPTH FROM FOCUS

The method of depth from focus and its current state can be summarized as follows:

1. The concept of depth from focus involves calculating distances to points in an observed scene, by modelling the effect that the camera's focal parameters have on images acquired with a small depth of field. This technique is passive and requires only a single camera.

2. Depth from focus is a viable technique for depth perception, since it has several key advantages over other passive techniques of depth perception.

    - It does not suffer from the *correspondence problem*, since depth is shown by a blur in the images rather than a parallax.

- It is suitable for hardware implementation, since the problem has a predictable computational path of deterministic sub-problems.
- It is suitable for parallel processing, since the depth calculation for any locality can be done *independently* from neighbouring local regions.
- It requires only one camera.
- It has no *occlusion problem*, since all images are taken from the same viewing angle and projection.

Depth from focus has some disadvantages, such as

- Error increases with distance squared;
- Problem is ill-posed;
- Ambiguity about point of focus; and
- No scene motion is permitted;

however in most controlled applications, these can usually be overcome, or sufficiently constrained.

3. Defocus can be modelled as a convolution by a point spread function. The nature of this point spread function can be modelled in two traditional ways: geometric optics and diffraction optics. For the optical camera parameters which were used in the experimental analysis, it was shown that the diffraction optics model yields a similar result to geometric optics. Considering the complexity of the diffraction optics model, geometric optics provides an adequate theoretical representation.

4. Early researchers trying to solve the depth from focus problem, developed methods for automatically focusing a camera on the point of interest. The distance

to the point in the scene could then be determined from the position of the lens. However, a key limitation of this technique was that it measured depth, *one* point at a time and required an adjustment of the lens setting for each point.

More recently, researchers have attempted to measure the amount of defocus, called *focal gradient* in different parts of an image, and therefore infer distance to a variety of points in the scene. Since one unfocused image of an unknown scene was an underconstrained problem (soft edges in the images may be either unfocused hard edges in the scene, *or* focused soft edges in the scene), three approaches were presented:

- Introduce extra cues into the scene;
- Make assumptions about the scene; or
- Take several images with different defocus operators.

9.3 CONCLUSIONS

The following original conclusions can be made, based on the contents of this thesis:

1. The general solution to the depth from focus problem can be condensed into the computational goal of finding a convolution ratio $h_3(x,y)$, which is a solution to the equation

$$i_1(x,y)\ [\otimes]\ h_3(x,y) = i_2(x,y)\ . \tag{9.1}$$

The operator $[\otimes]$ designates restricted convolution, where the borders of the kernel $h_3(x,y)$ are *not* convolved past the borders of the image $i_1(x,y)$. The local image region $i_1(x,y)$ was acquired with the defocus operator $h_1(x,y)$, and $i_2(x,y)$ was a local image region acquired with the defocus operator $h_2(x,y)$. It was

also shown that with no noise, $h_3(x,y)$ can be expressed in terms of $h_1(x,y)$ and $h_2(x,y)$, as

$$h_1(x,y) \otimes h_3(x,y) = h_2(x,y) \ . \tag{9.2}$$

Using the geometric optics model for defocus, it was shown that $h_3(x,y)$ is a unique indicator of depth.

The solution for $h_3(x,y)$ can be formulated with inverse filtering or with a matrix based approach. When only one of the images was blurred, in a noiseless environment, it was shown that the inverse filtering solution in the frequency domain had two fundamental limits to accuracy, over the matrix based approach:

- Finding the Fourier domain representation of a local signal requires windowing the spatial domain signal, however it introduces residual distortions in the spatial frequency domain; and
- Border effects from neighbouring regions can introduce spurious data.

When both of the images are blurred, the problem of border effects still remains.

2. In a noiseless environment, given signals $s(x) \in \Re(x)$ and $i(x) \in \Re(x)$, and the convolution kernel $h(x) \in \Re(x)$, where $\Re(x)$ is the set of all real signals, and the relation

$$s(x) \ [\otimes] \ h(x) = i(x) \ , \tag{9.3}$$

where the symbol $[\otimes]$ represents restricted convolution such that the ends of $h(x)$ do not convolve beyond the ends of $s(x)$. If $I\{s(x), i(x)\}$ is the set of all signals $s(x)$ and $i(x)$ where the exact recovery of $h(x)$ is possible using inverse filtering in the spatial frequency domain, and $M\{s(x), i(x)\}$ is the set of all signals $s(x)$ and $i(x)$ where the exact recovery of $h(x)$ is possible using a matrix based approach

in the spatial domain, then

$$I\{s(x), i(x)\} \subset M\{s(x), i(x)\} \ . \tag{9.4}$$

That is, the exact recovery of $h(x)$ can be performed for a larger family of local signals $s(x)$ and $i(x)$ by using a matrix based approach in the spatial domain, than by using inverse filtering in the spatial frequency domain.

3. To deal with noise present in acquired images, the technique of regularization has been used to augment the inverse filter method into a constrained inverse filtering solution. Regularization has also been used to augment the matrix based approach into an iterative matrix based solution.

4. A novel application of a general iterative matrix solution has been theoretically justified, which circumvents the problem of border effects encountered with inverse filtering. This new method is general for the reasons below.

   Minimal constraints on the scene: The two images can be obtained under any differing conditions of defocus (change of aperture, position of image plane, or focal length), one image need not be obtained with a pinhole aperture, and no assumptions are made about the nature of the edges in the scene.

   Independent of the defocus operator model: The defocus operator may be assumed to be a parametric shape, or for more accuracy it can be experimentally measured.

   Selective trade-offs: The trade-offs in forming the convolution ratio $h_3(x, y)$; fit to data and fit to expected models, are resolved through regularization.

5. The following four scenes were processed with the author's iterative matrix solution and constrained inverse filtering:

   (a) A sloped sheet of paper with various letters of the alphabet.

   (b) Two blocks which show a shadow edge, an occlusion edge, and a printed grey edge.

   (c) A flat sheet of paper with printed, grey, ramp edges of increasing width.

   (d) A sloped piece of paper with equally spaced, vertical, black bars which produces a phase inversion in the acquired images.

   On all of the four experimental scenes processed, the iterative matrix solution was consistently more accurate than constrained inverse filtering. The following conclusions can also be made about the iterative matrix solution:

   (a) For the camera lens employed, geometric optics provides a good first approximation;

   (b) The most accurate depth results can be achieved when the actual defocus operator is experimentally measured and modelled;

   (c) The accuracy of the algorithm increases as the defocus operators become smaller, and the images are less blurred.

   (d) The algorithm is most accurate when the scene contains edges with a small ramped, grey transition area.

   (e) The best features processed, had an RMS error of 2.8% in terms of *expected range*, and 0.5% in terms of *distance from the camera*; and

   (f) The algorithm is able to cope with phase reversal in one image.

Experimental results of obtaining depth from focus using images obtained with two different apertures, where one is not a pinhole, have not previously been reported in the literature. However, the author's results still compare favorably with Pentland's measured error of 2.5%, where one image was acquired with a pinhole aperture.

9.4 OPPORTUNITIES FOR FURTHER WORK

This section outlines some suggestions for further work, which stem from the research work described in this thesis.

9.4.1 Geometric Correction

Since the geometric correction of images acquired with a camera is well documented in the literature [149][98], it was considered unlikely that a peripheral effort could result in any unique contribution. Therefore it was not attempted as part of this thesis research.

It is very likely, however, that including geometric corrections into the depth from focus algorithm will improve the experimental results. For example, it was observed that the aspect ratio of the digitization process was 20% from unity, whereas the construction of the table of $h_3(x, y)$ patterns assumed that it was unity.

9.4.2 Speed Up Implementation

The greatest disadvantage of the iterative matrix based solution presented in this thesis, is the time required to calculate the depth values. Almost all of the computational time is spent convolving the local region $i_1(x, y)$ with the estimated convolution ratio $\hat{h}_3(x, y)$ to obtain $\hat{i}_2(x, y)$. Since finding the best $\hat{h}_3(x, y)$ is an iterative process, several convolutions must be performed. Currently all the convolutions are done in software,

and the resulting $\hat{i}_2(x, y)$ is discarded after each iteration.

There are two ways of speeding up these calculations.

1. Implement convolutions in hardware.

2. Make some approximations to the iterative process, so that the results of the previous convolution are used to obtain the results for next convolution.

Another possible way to speed up this algorithm, may be to implement part of the spatial domain convolution in the frequency domain in the manner introduced in Section 6.5.3. This differs from the inverse filtering solution offered by other researchers, in that *different* sized pieces are taken in spatial domain. At this point however, the exact algorithm has not yet been identified.

9.4.3 Further Experimental Tests

A variety of further experimental tests are possible, which would give further insight into the strengths and limitations of the current algorithm. Tests could be performed on a variety of featured scenes, including textured scenes. The most interesting test would be to take the two images i_1 and i_2, with a larger aperture separation, since the tests in Chapter 8 were performed with the images differing by only one *f*-stop. Intuitively, it should be possible to gain greater accuracy this way, although this constrains the scene to be more illuminated.

9.4.4 Space-variant Solution

This is perhaps the most challenging future course of research, which involves wrapping another layer around the current algorithm. The depth points from the present algorithm would act as a starting point and then treating the image as a whole, with

the defocus operator space-variant, another iteration could take place. This fine tuning should help to eliminate errors where multiple and conflicting depths have been present in a previously chosen local area. This algorithm may also want to take depth continuity into consideration, however step discontinuities, which often occur in range images would also have to be taken into account.

Appendix A

GLOSSARY OF COMMON SYMBOLS

D_{fi}	The distance from the lens to the image, when the image is *guaranteed* to be in focus.
D_i	The distance from the lens to the image, which may or may *not* be in focus.
D_o	The distance from the lens to an observed object.
f	The f-number of an optical system.
f_x, f_y	The spatial frequency domain components (cycles per spatial unit).
F	The focal length of a lens.
$\mathcal{F}$	The Fourier operator.
$\mathcal{F}^{-1}$	The inverse Fourier operator.
$h(x,y)$	A defocus operator.
$h_1(x,y)$	The small defocus operator.
$h_2(x,y)$	The large defocus operator.
$h_3(x,y)$	The convolution ratio of the large and small defocus operators.
$\mathbf{h_3}$	The vector formed from $h_3(x)$.
$\hat{h}_3(x,y)$	A guess at $h_3(x,y)$.
$H(f_x,f_y)$	A defocus operator in the spatial frequency domain.
$H_1(f_x,f_y)$	The small defocus operator in the spatial frequency domain.
$H_2(f_x,f_y)$	The large defocus operator in the spatial frequency domain.
$H_3(f_x,f_y)$	The frequency domain division of the large and small defocus operators.
i	An acquired image, which may be corrupted with noise or blurred in some areas.
$i(x,y)$	A local region of an acquired image.

$i_1(x,y)$	The local image region acquired with defocus operator $h_1(x,y)$.
$i_2(x,y)$	The local image region acquired with defocus operator $h_2(x,y)$.
$\hat{i}_2(x,y)$	A guess at $i_2(x,y)$.
$\mathbf{i_2}$	A vector formed from $i_2(x)$.
$I(f_x,f_y)$	A local image function in the spatial frequency domain.
$I_1(f_x,f_y)$	The local image function $i_1(x,y)$ in the spatial frequency domain.
$I_2(f_x,f_y)$	The local image function $i_2(x,y)$ in the spatial frequency domain.
$[\mathbf{i_1}]$	A matrix formed from $i_1(x)$.
$J_n(x)$	The Bessel operator of order n.
L	The limiting diameter of the lens or aperture.
$n_1(x,y)$	First noise distribution in the spatial domain.
$n_2(x,y)$	Second noise distribution in the spatial domain.
N	The number of points in a vector or one dimension of a matrix.
P	The digitization ratio of the camera.
R	The radius of the blur circle.
R_p	The radius of the blur circle in pixels.
$s(x,y)$	The spatial domain description of the scene being imaged.
$S(f_x,f_y)$	The scene function in the spatial frequency domain.
$w_1(x,y)$	First windowing function in the spatial domain.
$w_2(x,y)$	Second windowing function in the spatial domain.
$w_3(x,y)$	Third windowing function in the spatial domain.
$W_1(f_x,f_y)$	First windowing function in the spatial frequency domain.
$W_2(f_x,f_y)$	Second windowing function in the spatial frequency domain.
x,y	The spatial domain components in a local area.
δ	The distance from the image plane to the point of perfect focus.
ϵ_i	Errors due to inverse filtering in the frequency domain.
ϵ_m	Errors due to matrix based solution in the spatial domain.

θ The angular component in the spatial domain.

λ A scalar parameter in a regularization equation.

ρ The radial component in the spatial frequency domain.

ρ_p The radial component in the spatial frequency domain measured in pixel units.

σ The spatial constant for the Gaussian function.

ϕ The angular component in the spatial frequency domain.

$\otimes$ The convolution operator.

$[\otimes]$ A convolution operation such that the borders of the convolution kernel do not extend beyond the borders of the signal.

Appendix B

DECONVOLUTION OF LOCAL SIGNALS

B.1 INTRODUCTION

Given a signal $s(x)$ and a convolution kernel $h(x)$, the result $i(x)$ in a noiseless environment, can be expressed as,

$$s(x) \otimes h(x) = i(x) \ . \tag{B.5}$$

The *convolution theorem* [50, p. 84] states that the dual of this operation can be represented in the spatial frequency domain as

$$S(f_x) \cdot H(f_x) = I(f_x) \ , \tag{B.6}$$

where the following are Fourier pairs

$$s(x) \iff S(f_x) \ , \tag{B.7}$$

$$h(x) \iff H(f_x) \ \text{ and} \tag{B.8}$$

$$i(x) \iff I(f_x) \ . \tag{B.9}$$

The *deconvolution problem* states that given $s(x)$ and $i(x)$, find $h(x)$. A simple way of solving the deconvolution problem is by inverse filtering. This is usually implemented in the frequency domain. Solving (B.6) for $H(f_x)$ yields

$$H(f_x) = \frac{I(f_x)}{S(f_x)} \ . \tag{B.10}$$

If the signal $s(x)$ is part of a larger signal s, then convolution of a local region can be written as

$$s(x) \; [\otimes] \; h(x) = i(x) \; . \tag{B.11}$$

The operator $[\otimes]$ designates restricted convolution, where the ends of the kernel $h(x)$ are *not* convolved past the ends of the signal $s(x)$. This restricted definition is necessary, because in this case the result of convolving the kernel past the ends of the local region of the signal is *unknown*.

Under these conditions, for most signals $s(x)$ and $i(x)$, the deconvolution problem *cannot* be solved exactly by using inverse filtering, even in a noiseless environment. This appendix will present a matrix based approach as a more appropriate and accurate solution. The matrix based approach exhaustively specifies and solves the linear equations formed by the restricted convolution of (B.11).

B.2 OBJECTIVE

This appendix presents a proof for the following theorem.

Theorem 1 *In a noiseless environment, given signals $s(x) \in \Re(x)$ and $i(x) \in \Re(x)$, and the convolution kernel $h(x) \in \Re(x)$, where $\Re(x)$ is the set of all real signals, and the relation*

$$s(x) \; [\otimes] \; h(x) = i(x) \; , \tag{B.12}$$

where the symbol $[\otimes]$ represents restricted convolution such that the ends of $h(x)$ do not convolve beyond the ends of $s(x)$. If $I\{s(x), i(x)\}$ is the set of all signals $s(x)$ and $i(x)$ where the exact recovery of $h(x)$ is possible using inverse filtering in the spatial frequency domain, and $M\{s(x), i(x)\}$ is the set of all signals $s(x)$ and $i(x)$ where the exact recovery of $h(x)$ is possible using a matrix based approach in the spatial domain,

then

$$I\{s(x), i(x)\} \subset M\{s(x), i(x)\} \ . \tag{B.13}$$

That is, the exact recovery of $h(x)$ can be performed for a larger family of local signals $s(x)$ and $i(x)$ by using a matrix based approach in the spatial domain, than by using inverse filtering in the spatial frequency domain.

This theorem will be proved in one dimension for simplicity, however the results can be directly extended to two dimensions.

B.3 OVERVIEW

Proof of Theorem 1 will come as a result of four lemmas. The first two lemmas establish some necessary representations. Lemma 1 will establish that the multiplication of two signals in the discrete Fourier domain can be represented by multiplication of a circulant matrix with a vector, in the spatial domain. Then Lemma 2 will show that, more generally, the restricted convolution of a local region of a signal with a kernel, can be represented by the multiplication of a Toeplitz matrix and a vector.

The next two lemmas build on the last two lemmas, to draw some conclusions regarding the solution of the deconvolution problem using both methods. Lemma 3 will prove that all deconvolution problems that can be solved using inverse filtering, can also be solved with a matrix based approach. Next, Lemma 4 will prove that some deconvolution problems which can be solved using the matrix based approach cannot be solved by inverse filtering.

The logical conclusion from Lemmas 3 and 4, is that more deconvolution problems can be solved exactly using the matrix based approach than by inverse filtering, which is the proof of Theorem 1.

B.4 REPRESENTATION

B.4.1 Inverse Filtering Approach

Lemma 1 *Given complex signals $S(f_x)$, $H(f_x)$ and $I(f_x)$ in the discrete Fourier domain, and the relation*

$$S(f_x) \cdot H(f_x) = I(f_x) \ , \tag{B.14}$$

this can be equivalently represented in the spatial domain by the matrix equation

$$[\mathbf{s}_\mathbf{C}] \cdot \mathbf{h} = \mathbf{i} \ , \tag{B.15}$$

where $[\mathbf{s}_\mathbf{C}]$ is a circulant matrix formed from $s(0), s(1), s(2), \ldots, s(N-1)$ according to

$$\mathbf{s}_\mathbf{C}(k,i) = \begin{cases} s(k-i) & \textit{for} \quad i \leq k \ , \quad \textit{and} \\ s(k-i+N) & \textit{for} \quad i > k \ , \end{cases} \tag{B.16}$$

where $k, i = 0, 1, 2, \ldots, N-1$ and $s(x)$ is the inverse Fourier transform of $S(f_x)$. Also, $\mathbf{h}$ and $\mathbf{i}$ are vectors formed from the inverse Fourier transforms of $H(f_x)$ and $I(f_x)$ respectively. That is, the multiplication of signals in the discrete Fourier domain, can be equivalently represented by multiplication of a circulant matrix with a vector in the spatial domain.

Before the proof of Lemma 1 is presented, a number of matrix equivalencies must first be established. The expressions below follow Gonzalez and Wintz [50, §5.2], and are also presented by Hunt [68].

Consider an $N \times N$ circulant matrix $[\mathbf{s}_\mathbf{C}]$, where

$$[\mathbf{s}_\mathbf{C}] = \begin{bmatrix} s(0) & s(N-1) & s(N-2) & \cdots & s(1) \\ s(1) & s(0) & s(N-1) & \cdots & s(2) \\ s(2) & s(1) & s(0) & \cdots & s(3) \\ \vdots & \vdots & \vdots & & \vdots \\ s(N-1) & s(N-2) & s(N-3) & \cdots & s(0) \end{bmatrix} , \tag{B.17}$$

defined for scalars $s(0), s(1), s(2), \ldots, s(N-1)$ and $k, i = 0, 1, 2, \ldots, N-1$, as

$$\mathbf{s_C}(k,i) = \begin{cases} s(k-i) & \text{for} \quad i \leq k \\ s(k-i+N) & \text{for} \quad i > k \ . \end{cases} \tag{B.18}$$

Also given, is a scalar function $\lambda(k)$, of the form

$$\begin{aligned} \lambda(k) &= s(0) + s(N-1)\exp\left(j\frac{2\pi}{N}k\right) + s(N-2)\exp\left(j\frac{2\pi}{N}2k\right) + \cdots \\ &\quad + s(1)\exp\left(j\frac{2\pi}{N}(N-1)k\right) \ , \end{aligned} \tag{B.19}$$

where $j = \sqrt{-1}$, and a vector $\mathbf{w}(k)$, where

$$\mathbf{w}(k) = \begin{bmatrix} 1 \\ \exp\left(j\frac{2\pi}{N}k\right) \\ \exp\left(j\frac{2\pi}{N}2k\right) \\ \vdots \\ \exp\left(j\frac{2\pi}{N}(N-1)k\right) \end{bmatrix} \ , \tag{B.20}$$

for $k = 0, 1, 2, \ldots, N-1$. It is evident, by matrix multiplication, that

$$[\mathbf{s_C}]\,\mathbf{w}(k) = \lambda(k)\mathbf{w}(k) \ . \tag{B.21}$$

Equation (B.21) indicates that $\mathbf{w}(k)$ is an eigenvector of the circulant matrix $[\mathbf{s_C}]$, and $\lambda(k)$ is its corresponding eigenvalue [123, §11.0][49, p. 333].

It is now possible to form an $N \times N$ matrix $[\mathbf{W}]$, by arranging the N eigenvectors of $[\mathbf{s_C}]$ as columns of $[\mathbf{W}]$. Then, the kith element of the matrix $[\mathbf{W}]$ is expressed as

$$\mathbf{W}(k,i) = \exp\left(j\frac{2\pi}{N}ki\right) \ , \tag{B.22}$$

where $k, i = 0, 1, 2, \ldots, N-1$. The inverse matrix $[\mathbf{W}]^{-1}$ can be written by inspection as

$$\mathbf{W}(k,i) = \frac{1}{N}\exp\left(-j\frac{2\pi}{N}ki\right) \ , \tag{B.23}$$

and it can be verified using (B.22) and (B.23), that

$$[\mathbf{W}] \cdot [\mathbf{W}]^{-1} = [\mathbf{W}]^{-1} \cdot [\mathbf{W}] = [\mathbf{I}] \ , \tag{B.24}$$

where $[\mathbf{I}]$ is the $N \times N$ identity matrix. It follows that

$$[\mathbf{s_C}] = [\mathbf{W}] \cdot [\mathbf{S_D}] \cdot [\mathbf{W}]^{-1} \ , \tag{B.25}$$

if $[\mathbf{S_D}]$ is a diagonal matrix, whose elements,

$$\mathbf{S_D}(k,k) = \lambda(k) \ , \tag{B.26}$$

are the eigenvalues of $[\mathbf{s_C}]$ [123, §11.0]. Finally, using (B.24), (B.25) becomes

$$[\mathbf{S_D}] = [\mathbf{W}]^{-1} \cdot [\mathbf{s_C}] \cdot [\mathbf{W}] \ . \tag{B.27}$$

After the key matrix identities above have been established, the proof of Lemma 1 follows. The expression

$$S(f_x) \cdot H(f_x) = I(f_x) \ , \tag{B.28}$$

where $S(f_x)$, $H(f_x)$ and $I(f_x)$ are complex signals in the discrete Fourier domain, can be represented in matrix form as

$$[\mathbf{S_D}] \cdot \mathbf{H} = \mathbf{I} \ , \tag{B.29}$$

where $[\mathbf{S_D}]$ is made up of the elements of $S(f_x)$ placed on the diagonal, and $\mathbf{H}$ and $\mathbf{I}$ are vectors formed from $H(f_x)$ and $I(f_x)$ respectively. Then using (B.27), (B.29) becomes

$$[\mathbf{W}]^{-1} \cdot [\mathbf{s_C}] \cdot [\mathbf{W}] \cdot \mathbf{H} = \mathbf{I} \text{ and} \tag{B.30}$$

$$[\mathbf{s_C}] \cdot [\mathbf{W}] \cdot \mathbf{H} = [\mathbf{W}] \cdot \mathbf{I} \ . \tag{B.31}$$

It can be recognized from (B.22) that the matrix $[\mathbf{W}]$, when multiplied by a vector, performs the inverse discrete Fourier transform on that vector. Therefore, let

$$[\mathbf{W}] \cdot \mathbf{H} = \mathbf{h} \ , \tag{B.32}$$

where $\mathbf{h}$ is a vector representing the inverse Fourier transform of $\mathbf{H}$. Similarly, let

$$[\mathbf{W}] \cdot \mathbf{I} = \mathbf{i} \,, \tag{B.33}$$

where $\mathbf{i}$ is a vector representing the inverse Fourier transform of $\mathbf{I}$. Then, using (B.32) and (B.33), (B.31) can be rewritten as

$$[\mathbf{s}_\mathbf{C}] \cdot \mathbf{h} = \mathbf{i} \,. \tag{B.34}$$

Therefore, it has been shown that the multiplication of signals in the discrete Fourier domain, can be equivalently represented by multiplication of a circulant matrix with a vector in the spatial domain, which completes the proof of Lemma 1.

B.4.2 Matrix Based Approach

Lemma 2 *Given spatial domain signals $s(x)$ and $i(x)$, a convolution kernel $h(x)$, and the relation*

$$s(x) \; [\otimes] \; h(x) = i(x) \,, \tag{B.35}$$

where the symbol $[\otimes]$ represents restricted convolution such that the ends of $h(x)$ do not convolve beyond the ends of $s(x)$. This can be represented by the matrix equation

$$[\mathbf{s}_\mathbf{T}] \cdot \mathbf{h} = \mathbf{i} \,, \tag{B.36}$$

where $[\mathbf{s}_\mathbf{T}]$ is a Toeplitz matrix formed from $s(1-N), s(2-N), \ldots, s(0), \ldots, s(N-1)$ according to

$$\mathbf{s}_\mathbf{T}(k,i) = s(k-i) \,, \tag{B.37}$$

where $k, i = 0, 1, 2, \ldots, N-1$. Also $\mathbf{h}$ and $\mathbf{i}$ are vectors formed from $h(x)$ and $i(x)$ respectively. That is, convolution of a signal by a kernel, where the ends of the kernel do not convolve past the ends of the signal, can be represented by a multiplication of a Toeplitz matrix with a vector.

The proof for Lemma 2 is similar to the presentation by Andrews and Hunt [4, Appendix A]. Consider Figure B.1 which shows $s(x)$ and $i(x)$ as two segments of larger waveforms s and i. Let $i(x)$ and $h(x)$ be N points long. Since $s(x)$ is part of a larger waveform, the ends of the kernel $h(x)$ cannot convolve past the ends of $s(x)$. Then by looking at (B.35), it is easily seen that $s(x)$ will need to be $2N-1$ points long, to fully recover $h(x)$.[1]

A critical point to note, is that this representation is *not* the conventional form of convolution. Conventionally the convolution kernel $h(x)$ moves *past* the ends of $s(x)$ so that the resulting $i(x)$ is longer than $s(x)$. In the representation above, the kernel moves *within* $s(x)$ so that the resulting $i(x)$ is smaller than $s(x)$. This allows the deconvolution problem to be solved for local regions of larger waveforms.

To simplify the analysis, but without further loss of generality, it will be assumed that both $h(x)$ and $i(x)$ start with an index of $-(N-1)/2$, and are centered at the origin.

Then by inspection, the restricted convolution equation (B.35), can be written as the following series of linear equations

$$\begin{aligned}
i(-\tfrac{N-1}{2}) &= s(0)h(-\tfrac{N-1}{2}) + s(-1)h(-\tfrac{N-3}{2}) + \cdots + s(1-N)h(\tfrac{N-1}{2}) \ , \\
i(-\tfrac{N-3}{2}) &= s(1)h(-\tfrac{N-1}{2}) + s(0)h(-\tfrac{N-3}{2}) + \cdots + s(2-N)h(\tfrac{N-1}{2}) \ , \\
\vdots &= \vdots \\
i(\tfrac{N-1}{2}) &= s(N-1)h(-\tfrac{N-1}{2}) + s(N-2)h(-\tfrac{N-3}{2}) + \cdots \\
&\quad + s(0)h(\tfrac{N-1}{2}) \ .
\end{aligned} \tag{B.38}$$

The linear equations above can be represented by the matrix equation

$$[\mathbf{s}_\mathbf{T}] \cdot \mathbf{h} = \mathbf{i} \ , \tag{B.39}$$

[1]Since the formulation is slightly different if N is even or odd, N will be considered to be odd for this analysis.

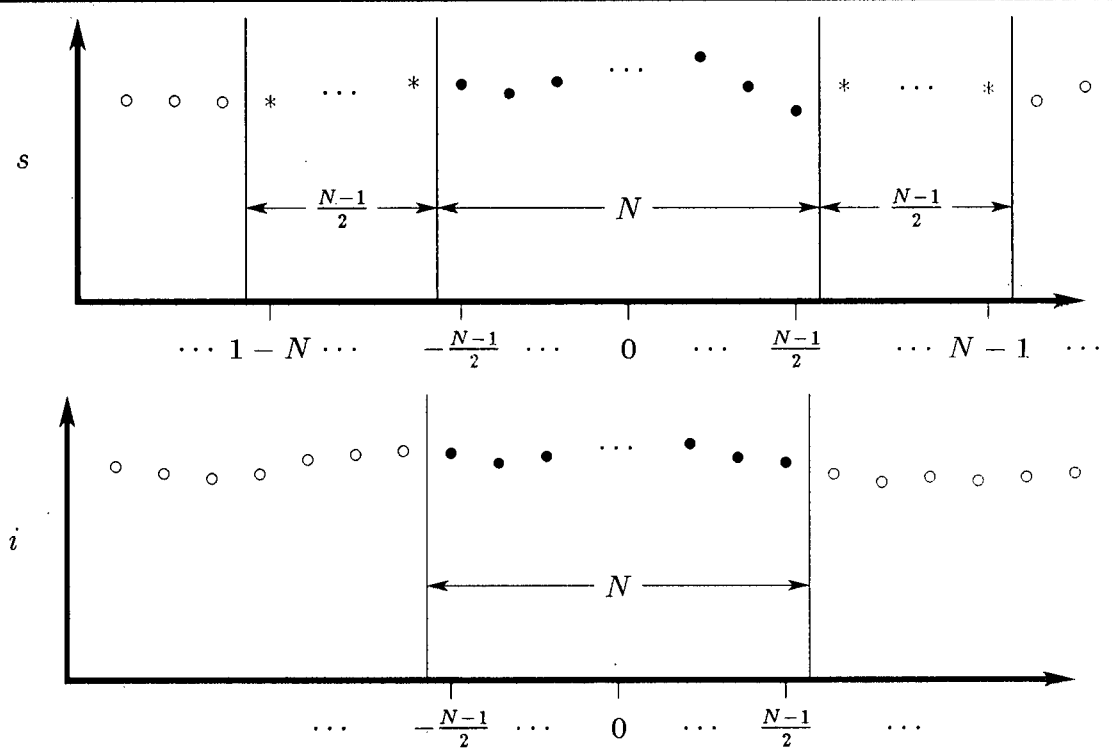


Figure B.1: The local segments $s(x)$ and $i(x)$ of the larger signals s and i are denoted by ($\bullet$) symbols, and the ($*$) symbol denotes the extra length of $s(x)$ required by the convolution kernel $h(x)$.

where

$$[\mathbf{s_T}] = \begin{bmatrix} s(0) & s(-1) & \cdots & s(1-N) \\ s(1) & s(0) & \ddots & \vdots \\ \vdots & \ddots & \ddots & s(-1) \\ s(N-1) & \cdots & s(1) & s(0) \end{bmatrix}, \tag{B.40}$$

$$\mathbf{h} = \begin{bmatrix} h(-\frac{N-1}{2}) \\ h(-\frac{N-3}{2}) \\ \vdots \\ h(\frac{N-1}{2}) \end{bmatrix} \text{ and} \tag{B.41}$$

$$\mathbf{i} = \begin{bmatrix} i(-\frac{N-1}{2}) \\ i(-\frac{N-3}{2}) \\ \vdots \\ i(\frac{N-1}{2}) \end{bmatrix}. \tag{B.42}$$

Note that sequence of $s(x)$ has been inverted and shifted row by row into the matrix $[\mathbf{s_T}]$ in (B.40). Then $[\mathbf{s_T}]$ can be recognized as a Toeplitz matrix, since $[\mathbf{s_T}]$ is $N \times N$ and there exist scalars $s(1-N), s(2-N), \ldots, s(0), \ldots, s(N-1)$, for $k, i = 0, 1, 2, \ldots, N-1$, such that [49, page 183],

$$\mathbf{s_T}(k,i) = s(k-i)\,. \tag{B.43}$$

Therefore, convolution of a signal by a kernel, where the ends of the kernel do not convolve past the ends of the signal, can be represented by a multiplication of a Toeplitz matrix with a vector, which completes the proof for Lemma 2.

B.5 SOLUTIONS

Lemmas 1 and 2 can now be used to draw some conclusions about solving restricted deconvolution problems using both inverse filtering and matrix based methods.

Lemma 3 *In a noiseless environment, given signals $s(x) \in \Re(x)$ and $i(x) \in \Re(x)$, and the convolution kernel $h(x) \in \Re(x)$, where $\Re(x)$ is the set of all real signals, and the relation*

$$s(x)\ [\otimes]\ h(x) = i(x)\ , \tag{B.44}$$

where the symbol $[\otimes]$ represents restricted convolution such that the ends of $h(x)$ do not convolve beyond the ends of $s(x)$. If $I\{s(x), i(x)\}$ is the set of all signals $s(x)$ and $i(x)$ where the exact recovery of $h(x)$ is possible using inverse filtering in the spatial frequency domain, and $M\{s(x), i(x)\}$ is the set of all signals $s(x)$ and $i(x)$ where the exact recovery of $h(x)$ is possible using a matrix approach in the spatial domain, then

$$I\{s(x), i(x)\} \subseteq M\{s(x), i(x)\}\ . \tag{B.45}$$

That is, all restricted deconvolution problems that can be solved using inverse filtering in the spatial frequency domain can also be solved using a matrix approach in the spatial domain.

The proof of Lemma 3 follows closely after Lemma 1. Given the complex signals $S(f_x)$, $H(f_x)$ and $I(f_x)$, which are the discrete Fourier transforms of $s(x)$, $h(x)$, and $i(x)$ respectively, the restricted convolution equation has been represented by the matrix equation (B.29),

$$[\mathbf{S_D}] \cdot \mathbf{H} = \mathbf{I} \text{ and} \tag{B.46}$$

$$\mathbf{H} = [\mathbf{S_D}]^{-1} \cdot \mathbf{I}\ . \tag{B.47}$$

The diagonal matrix $[\mathbf{S_D}]$, is invertible as long as all of its elements are non-zero. Therefore, a solution to the restricted deconvolution problem using inverse filtering is possible if $[\mathbf{S_D}]$ has no non-zero elements.

From the proof of Lemma 1, it was concluded that (B.46) in the discrete Fourier domain can be equivalently expressed as (B.34) in the spatial domain,

$$[\mathbf{s_C}] \cdot \mathbf{h} = \mathbf{i} \text{ and} \tag{B.48}$$

$$\mathbf{h} = [\mathbf{s_C}]^{-1} \cdot \mathbf{i} \,. \tag{B.49}$$

The circulant matrix $[\mathbf{s_C}]$, is invertible if it is not singular. But a matrix is singular if and only if it has a zero eigenvalue [21, p. 60]. It was shown by (B.21) and (B.26) that $[\mathbf{S_D}]$ contains the eigenvalues of $[\mathbf{s_C}]$. Therefore, if (B.47) can be solved for $\mathbf{H}$, (B.49) can be solved for $\mathbf{h}$. Therefore, all restricted deconvolution problems that can be solved using inverse filtering in the spatial frequency domain can also be solved with a matrix based approach in the spatial domain.

That completes the proof of Lemma 3. As corroborating evidence, it can be pointed out that representation of the restricted deconvolution problem using inverse filtering yielded a circulant matrix in the spatial domain, while the representation of the restricted deconvolution problem in the matrix approach could be more generally expressed using a Toeplitz matrix. Since a circulant matrix is a specialized form of a Toeplitz matrix, therefore it appears that solutions calculated using inverse filtering are a *subset* of the solutions possible with a matrix based approach. This suspicion is confirmed in the next lemma.

Lemma 4 *In a noiseless environment, given signals $s(x) \in \Re(x)$ and $i(x) \in \Re(x)$, and the convolution kernel $h(x) \in \Re(x)$, where $\Re(x)$ is the set of all real signals, and the relation*

$$s(x) \; [\otimes] \; h(x) = i(x) \,, \tag{B.50}$$

where the symbol $[\otimes]$ represents restricted convolution such that the ends of $h(x)$ do not convolve beyond the ends of $s(x)$. If $I\{s(x), i(x)\}$ is the set of all signals $s(x)$ and

$i(x)$ where the exact recovery of $h(x)$ is possible using inverse filtering in the spatial frequency domain, and $M\{s(x), i(x)\}$ is the set of all signals $s(x)$ and $i(x)$ where the exact recovery of $h(x)$ is possible with a matrix based approach in the spatial domain, then there exists a subset $M_s\{s(x), i(x)\}$,

$$M_s\{s(x), i(x)\} \subset M\{s(x), i(x)\} \,, \tag{B.51}$$

such that

$$M_s\{s(x), i(x)\} \notin I\{s(x), i(x)\} \,, \tag{B.52}$$

That is, some restricted deconvolution problems that can be solved using a matrix approach in the spatial domain, cannot be exactly solved using inverse filtering in the spatial frequency domain.

Based on Lemmas 1 and 2, Lemma 4 can be proved, if it is found that a general Toeplitz matrix (representing the restricted deconvolution problem using a matrix based approach in the spatial domain) cannot always be converted into a circulant matrix (representing the restricted deconvolution problem using inverse filtering in the spatial frequency domain).

The steps of the proof will follow Gray [53][54], who has shown that a Toeplitz matrix is only *asymptotically* equivalent to a circulant matrix. The proof begins with (B.39), the matrix equation for restricted convolution, and (B.40) the definition of $[\mathbf{s_T}]$,

$$[\mathbf{s_T}] \cdot \mathbf{h} = \mathbf{i} \,, \tag{B.53}$$

where

$$[\mathbf{s_T}] = \begin{bmatrix} s(0) & s(-1) & \cdots & s(1-N) \\ s(1) & s(0) & \ddots & \vdots \\ \vdots & \ddots & \ddots & s(-1) \\ s(N-1) & \cdots & s(1) & s(0) \end{bmatrix} . \tag{B.54}$$

The objective will be to replace $[\mathbf{s_T}]$ with a circulant matrix $[\mathbf{s'_C}]$. The only possible way to turn $[\mathbf{s_T}]$ into $[\mathbf{s'_C}]$ is to expand its size to $2N-1 \times 2N-1$, since modifying any of the elements in the current-sized $[\mathbf{s_T}]$ would violate the exact relations given in (B.39). If $[\mathbf{s_T}]$ is expanded, then the vectors $\mathbf{h}$ and $\mathbf{i}$ must be expanded to $\mathbf{h}'$ and $\mathbf{i}'$ respectively, each of length $2N-1$. Then (B.53) becomes

$$[\mathbf{s'_C}] \cdot \mathbf{h}' = \mathbf{i}' \text{ and} \tag{B.55}$$

$$\begin{bmatrix} ? & \cdots & ? \\ \vdots & \boxed{\mathbf{s_T}} & \vdots \\ ? & \cdots & ? \end{bmatrix} \cdot \begin{bmatrix} ? \\ \underline{\overline{\mathbf{h}}} \\ ? \end{bmatrix} = \begin{bmatrix} ? \\ \underline{\overline{\mathbf{i}}} \\ ? \end{bmatrix}. \tag{B.56}$$

The question marks represent values which are yet to be assigned. The matrix of interest is

$$[\mathbf{s'_C}] = \begin{bmatrix} ? & ? & \cdots & \cdots & ? & ? \\ ? & s(0) & s(-1) & \cdots & s(1-N) & ? \\ \vdots & s(1) & s(0) & \ddots & \vdots & ? \\ ? & \vdots & \ddots & \ddots & s(-1) & \vdots \\ ? & s(N-1) & \cdots & s(1) & s(0) & ? \\ ? & ? & ? & \cdots & ? & ? \end{bmatrix}. \tag{B.57}$$

Some of the question marks can be filled in immediately, according to the definition of a Toeplitz matrix in (B.43),

$$[\mathbf{s'_C}] = \begin{bmatrix} s(0) & s(-1) & \cdots & \cdots & ? & ? \\ s(1) & s(0) & s(-1) & \cdots & s(1-N) & ? \\ \vdots & s(1) & s(0) & \ddots & \vdots & ? \\ ? & \vdots & \ddots & \ddots & s(-1) & \vdots \\ ? & s(N-1) & \cdots & s(1) & s(0) & s(-1) \\ ? & ? & ? & \cdots & s(1) & s(0) \end{bmatrix}, \tag{B.58}$$

however the upper right hand corner and lower left hand corner need further consideration. The definition for a circulant matrix was given in (B.18) for $s(x)$ ranging over $s(0), s(1), s(2), \ldots, s(N-1)$, however in this case the function $s(x)$ is defined over a larger range $s(1-N), \ldots, 0, \ldots, s(N-1)$. Gray gives the general form for $[\mathbf{s}'_{\mathbf{C}}]$ as

$$[\mathbf{s}'_{\mathbf{C}}] = \begin{bmatrix} s(0) & \cdots & s(1-N) & & s(N-1) & \cdots & s(1) \\ \vdots & s(0) & \ddots & & & \ddots & \vdots \\ s(N-1) & \ddots & s(0) & \ddots & 0 & & s(N-1) \\ & & \ddots & \ddots & & 0 & \\ & 0 & & \ddots & \ddots & & \\ s(1-N) & & 0 & \ddots & s(0) & \ddots & s(1-N) \\ \vdots & \ddots & & & \ddots & s(0) & \vdots \\ s(-1) & \cdots & s(1-N) & & s(N-1) & \cdots & s(0) \end{bmatrix}. \quad \text{(B.59)}$$

It can be seen from (B.59), that to comply with the circulant requirements, the upper right corner and lower left corner *must* repeat or wrap-around. This directly contradicts the requirements in building up the Toeplitz matrix (B.40) from Figure B.1, since there is no guarantee that the general waveform $s(x)$ is periodic.

Therefore, in general, the Toeplitz matrix created from the local waveform $s(x)$ cannot be exactly coerced into a circulant matrix, since $s(x)$ is not generally periodic. Therefore, there exist some restricted deconvolution problems that can be solved with a matrix based approach in the spatial domain, yet cannot be exactly solved using inverse filtering in the spatial frequency domain. That completes the proof of Lemma 4.[2]

Finally, from the proof of Lemma 3, all restricted deconvolution problems that can be solved using inverse filtering in the spatial frequency domain can also be solved using

[2]Gray [53][54] goes on to show, however, that if the question marks in (B.57) are filled with zeros, then only as N approaches infinity, are the eigenvalues of the Toeplitz and circulant matrices asymptotically equivalent.

a matrix based approach in the spatial domain, and from the proof of Lemma 4, some restricted deconvolution problems that can be solved using a matrix based approach in the spatial domain cannot be solved using inverse filtering in the spatial frequency domain. Therefore it can be concluded that more restricted deconvolution problems can be solved exactly with a matrix approach in the spatial domain than by using inverse filtering in the spatial frequency domain, which is the proof of Theorem 1.

B.6 CONCLUSION

This appendix has proved that in one dimension, in a noiseless environment, given signals $s(x) \in \Re(x)$ and $i(x) \in \Re(x)$, and the convolution kernel $h(x) \in \Re(x)$, where $\Re(x)$ is the set of all real signals, and the relation

$$s(x) \; [\otimes] \; h(x) = i(x) \; , \tag{B.60}$$

where the symbol $[\otimes]$ represents restricted convolution such that the ends of $h(x)$ do not convolve beyond the ends of $s(x)$. If $I\{s(x), i(x)\}$ is the set of all signals $s(x)$ and $i(x)$ where the exact recovery of $h(x)$ is possible using inverse filtering in the spatial frequency domain, and $M\{s(x), i(x)\}$ is the set of all signals $s(x)$ and $i(x)$ where the exact recovery of $h(x)$ is possible using a matrix based approach in the spatial domain, then

$$I\{s(x), i(x)\} \subset M\{s(x), i(x)\} \; . \tag{B.61}$$

That is, the exact recovery of $h(x)$ can be performed for a larger family of local signals $s(x)$ and $i(x)$ by using a matrix based approach in the spatial domain, than by using inverse filtering in the spatial frequency domain. Extension to two dimensions is straightforward, since another dimension does not change any of the basic premises on which this proof is based.

Bibliography

[1] *Fairchild Charge Coupled Device (CCD) Catalog.* Fairchild CCD Imaging, Palo Alto, California, 1982–1983.

[2] A. Lynn Abbott and Narendra Ahuja. Surface reconstruction by dynamic integration of focus, camera vergence, and stereo. In *Second International Conference on Computer Vision*, pages 532–543, IEEE Computer Society, December 1988.

[3] Hirotugu Akaike. Block toeplitz matrix inversion. *SIAM Journal on Applied Mathematics*, 24(2):234–241, March 1973.

[4] H.C. Andrews and B.R. Hunt. *Digital Image Restoration.* Prentice-Hall Inc., Englewood Cliffs, 1977.

[5] G. Backus and G. Gilbert. Uniqueness in the inversion of inaccurate gross earth data. *Philosophical Transactions of the Royal Society of London*, 266:123–192, 1970.

[6] H. Bässmann and Ph.W. Besslich. Monocular computer vision. In *Third International Conference on Image Processing and its applications*, pages 107–111, IEE, July 18–20 1989.

[7] R.H.T. Bates and M.J. McDonnell. *Image Restoration and Reconstruction. The Oxford Engineering Science Series*, Oxford University Press, 1986.

[8] Mario Bertero and Tomaso A. Poggio. Ill-Posed problems in early vision. *Proceedings of the IEEE*, 76(8):869–889, August 1988.

[9] Paul J. Besl. Active, optical range imaging sensors. *Machine Vision and Applications*, 1(2):127–152, 1988.

[10] Paul J. Besl. Active optical range imaging sensors. In Jorge L.C. Sanz, editor, *Advances in Machine Vision*, chapter 1, pages 1–63, Springer-Verlag, New York, 1989.

[11] Paul J. Besl. *Range Imaging Sensors.* Research Publication GMR-6090, Computer Science Department, General Motors Research Laboratories, Warren, Michigan, March 1988.

[12] Paul J. Besl. *Surfaces in range image understanding. Springer series in perception engineering*, Springer-Verlag, New York, 1988.

[13] J.D.E. Beynon and D.R. Lamb, editors. *Charge-coupled Devices and Their Applications.* McGraw-Hill, London, 1980.

[14] François Blais. Biris: A simple 3-D sensor. In *Optics, Illumination, and Image Sensing for Machine Vision*, pages 235–242, SPIE, 1986. Vol. 728.

[15] Bruce P. Bogert, M.J.R. Healy, and John W. Tukey. The quefrency alanysis of time series for echoes: cepstrum, pseudo-autocovariance, cross-cepstrum and saphe cracking. In Murray Rosenblatt, editor, *Proceedings of the Symposium on Time Series Analysis*, chapter 15, pages 209–243, John Wiley & Sons, New York, June 11-14, 1962.

[16] Max Born and Emil Wolf. *Principles of Optics.* Pergammon Press, New York, sixth edition, 1980.

[17] V. Michael Bove, Jr. Discrete fourier transform based depth-from-focus. In *Image Understanding and Machine Vision, 1989 Technical Digest Series V.14*, pages 118–121, Optical Society of America and Air Force Office of Scientific Research, June 1989.

[18] Ronald N. Bracewell. *The Fourier Transform and Its Applications.* McGraw-Hill Book Company, New York, revised second edition edition, 1986.

[19] Rafael Bracho, John F. Schlag, and Arthur C. Sanderson. *POPEYE: A Gray-Level Vision System for Robotics Applications.* Technical Report CMU-RI-TR-83-6, Carnegie-Mellon University, May 1983.

[20] Richard P. Brent. *Algorithms for Minimization without Derivatives.* Prentice-Hall, Englewood Cliffs, N.J., 1973.

[21] Richard Bronson. *Theory and Problems of Matrix Operations. Schaum's Outline Series*, McGraw-Hill, New York, 1989.

[22] F.W. Campbell. Correlation of accommodation between the two eyes. *Journal of the Optical Society of America*, 50(7):738, July 1960.

[23] F.W. Campbell and J.G. Robson. Application of fourier analysis to the visibility of gratings. *Journal of Physiology (London)*, 197:551–566, 1968.

[24] F.W. Campbell and G. Westheimer. Dynamics of accommodation responses of the human eye. *Journal of Physiology*, 151(2):285–295, May 1960.

[25] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, November 1986.

[26] P. Chavel and T.C. Strand. Range measurement using Talbot diffraction imaging of gratings. *Applied Optics*, 23(6):862–871, March 1984.

[27] Donald G. Childers, David P. Skinner, and Robert C. Kemerait. The cepstrum: A guide to processing. *Proceedings of the IEEE*, 65(10):1428–1443, October 1977.

[28] Donald G. Childers, Robert S. Varga, and Nathan W. Perry, Jr. Composite signal decomposition. *IEEE Transactions on Audio and Electroacoustics*, AU-18(4):471–477, December 1970.

[29] P.D. Chuang. The cepstrum for passive monocular visual ranging in robotics. In *Proceedings of the International Workshop on Industrial Applications of Machine Vision and Machine Intelligence*, pages 222–228, IEEE, Feb 2–5 1987.

[30] T.A.C.M. Claasen and W.F.G. Mecklenbräuker. The wigner distribution–A tool for time-frequency analysis Part I: continuous-time signals. *Philips Journal of Research*, 35(3):217–250, 1980.

[31] T.A.C.M. Claasen and W.F.G. Mecklenbräuker. The wigner distribution–A tool for time-frequency analysis Part II: discrete-time signals. *Philips Journal of Research*, 35(4/5):276–300, 1980.

[32] T.A.C.M. Claasen and W.F.G. Mecklenbräuker. The wigner distribution–A tool for time-frequency analysis Part III: relations with other time-frequency signal transformations. *Philips Journal of Research*, 35(6):372–389, 1980.

[33] James J. Clark and Nicola J. Ferrier. Modal control of an attentive vision system. In *Second International Conference on Computer Vision*, pages 532–543, IEEE Computer Society, December 1988.

[34] Timothy R. Corle, Jeffrey T. Fanton, and Gordon S. Kino. Distance measurements by differential confocal optical ranging. *Applied Optics*, 26(12):2416–2420, June 1987.

[35] Tom N. Cornsweet. *Visual Perception.* Academic Press, New York, 1970.

[36] Hewitt D. Crane. *A Theoretical Analysis of the Visual Accommodation System in Humans.* Technical Report NAS 2-2760, NASA Ames Research Center, 1966.

[37] Trevor Darrell and Kwangyoen Wohn. Calculating depth from focus using a pyramid architecture. In *Automated Inspection and High Speed Vision Architectures*, pages 57–62, SPIE, 1987. Vol. 849.

[38] Trevor Darrell and Kwangyoen Wohn. Pyramid based depth from focus. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 504–509, 1988.

[39] Hal Denstman. State-of-the-art optics: automated image focusing. *Industrial Photography*, 33–37, July 1980.

[40] K. Engelhardt and Gerd Häusler. Acquisition of 3-D data by focus sensing. *Applied Optics*, 27(22):4684–4689, November 1988.

[41] Ahmed Erteza. Depth of convergence of a sharpness index autofocus system. *Applied Optics*, 16(8):2273–2278, August 1977.

[42] Ahmed Erteza. Sharpness index and its application to focus control. *Applied Optics*, 15(4):877–881, April 1976.

[43] B. Roy Frieden. Image restoration by discrete convolution of minimal length. *Journal of the Optical Society of America*, 64(5):682–686, May 1974.

[44] D. Gabor. Theory of communication. *Journal of the Institute of Electrical Engineers*, 93-III:429–457, 1946.

[45] G. Garibotto and P. Storace. 3-D range estimation from the focus sharpness of edges. In V. Cantoni, V. Di Gesù, and S. Levialdi, editors, *Image Analysis and Processing II*, pages 321–328, Plenum Press, New York, 1988.

[46] Harinath Garudadri. *Identification of Invariant Acoustic Cues in Stop Consonants Using the Wigner Distribution.* PhD thesis, University of British Columbia, Vancouver, December 1987.

[47] Bernd Girod and Stephen Scherock. Depth from defocus of structured light. In *Optics, Illumination, and Image Sensing for Machine Vision IV*, SPIE, 1989. Vol. 1194.

[48] Norman Goldberg. Inside autofocus: How the magic works. *Popular Photography*, 77–83, February 1982.

[49] Gene H. Golub and Charles F. Van Loan. *Matrix Computations.* The Johns Hopkins University Press, second edition, 1989.

[50] Rafael C. Gonzalez and Paul Wintz. *Digital Image Processing.* Addison-Wesley, Reading Massachusetts, second edition, 1987.

[51] Joseph W. Goodman. *Introduction to Fourier Optics. Physical and Quantum Electronics Series*, McGraw-Hill, Toronto, 1968.

[52] K. Gopalan and C.S. Chen. Computation of the two-dimensional fourier transform of circularly symmetric functions. In *Proceedings, International Conference on Acoustics, Speech, and Signal Processing*, pages 1546–1548, IEEE, March 26–29, 1985.

[53] Robert M. Gray. *Toeplitz and Circulant Matrices: A Review.* Technical Report 6502-1, Information Systems Laboratory, Stanford Electronics Laboratories, Stanford University, Stanford, California, June 1971.

[54] Robert Molten Gray. On the asymptotic eigenvalue distribution of toeplitz matrices. *IEEE Transactions on Information Theory*, IT-18(6):725–730, November 1972.

[55] W. Eric L. Grimson. Computational experiments with a feature based stereo algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-7(1):17–34, January 1985.

[56] P. Grossman. Depth from focus. *Pattern Recognition Letters*, 5:63–69, January 1987.

[57] Jacques Hadamard. *Lectures on Cauchy's Problem in Linear Partial Differential Equations.* Dover Publications, New York, 1952.

[58] Kentaro Hanma, Michio Masuda, Hiroaki Nabeyama, and Yohei Saito. Novel technologies for automatic focusing and white balancing of solid state color video camera. *IEEE Transactions on Consumer Electronics*, CE-29(3):376–382, August 1983.

[59] Fredric J. Harris. On the use of windows for harmonic analysis with the discrete fourier transform. *Proceedings of the IEEE*, 66(1):51–83, January 1978.

[60] Eugene Hecht. *Optics.* Addison-Wesley, Reading, Massachusetts, second edition, 1987.

[61] H.H. Hopkins. The frequency response of a defocused optical system. *Proceedings of the Royal Society of London, A*, 231:91–103, 1955.

[62] H.H. Hopkins. The numerical evaluation of the frequency response of optical systems. *The Proceedings of the Physical Society (London), B*, 70:1002, 1957.

[63] H.H. Hopkins. On the diffraction theory of optical images. *Proceedings of the Royal Society of London, A*, 217:408–432, 1953.

[64] H.H. Hopkins. *Wave Theory of Aberrations.* Oxford University Press, London, 1950.

[65] Berthold Klaus Paul Horn. *Focusing.* A.I. Memo 160, Massachusetts Institute of Technology, May 1968.

[66] Berthold Klaus Paul Horn. *Robot Vision. The MIT Electrical Engineering And Computer Science Series*, The MIT Press, Cambridge Massachusetts, 1986.

[67] To R. Hsing. Correction of pixel nonuniformities for solid-state imagers. In *Processing of Images and Data from Optical Sensors*, pages 218–224, SPIE, 1981. Vol. 292.

[68] B.R. Hunt. A matrix theory proof of the discrete convolution theorem. *IEEE Transactions on Audio and Electroacoustics*, AU-19(4):285–288, December 1971.

[69] Ten-lee Hwang, James J. Clark, and Alan L. Yuille. A depth recovery algorithm using defocus information. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 476–481, 1989.

[70] Ten-lee Hwang, James J. Clark, and Alan L. Yuille. *A Depth Recovery Algorithm Using Defocus Information.* Technical Report 89-2, Harvard Robotics Laboratory, Cambridge, MA 02138, 1988.

[71] Akira Ishizaki et al. Distance detection apparatus. United States Patent # 4614418, September 30 1986.

[72] R.A. Jarvis. Focus optimisation criteria for computer image processing. *Microscope*, 24(2nd quarter):163–180, 1976.

[73] R.A. Jarvis. A perspective on range-finding techniques for computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-5:122–139, March 1983.

[74] R.A. Jarvis. Range from out of focus blur. In *Australian Joint Artificial Intelligence Conference*, November 1988.

[75] Francis A. Jenkins and Harvey E. White. *Fundamentals of Optics.* McGraw-Hill, New York, third edition, 1957.

[76] Elbert T. Johnson and Lawrence J. Goforth. Metaphase spread detection and focus using closed circuit television. *The Journal of Histochemistry and Cytochemistry*, 22(7):536–545, 1974.

[77] Avinash C. Kak. Depth perception for robots. In Shimon Y. Nof, editor, *Handbook of Industrial Robotics*, chapter 16, pages 272–319, John Wiley & Sons, New York, 1985.

[78] N. Kalouptsidis, G. Carayannis, and D. Manolakis. On block matrices with elements of special structure. In *International Conference on Acoustics Speech and Signal Processing*, pages 1744–1747, IEEE, May 1982.

[79] Takeo Kanade and Haruhiko Asada. Noncontact visual three-dimensional ranging devices. In Bruce R. Altschuler, editor, *3-D Machine Perception*, pages 48–53, SPIE, April 23–24 1981. Vol. 283.

[80] H.A.E. Keitz. *Light Calculations and Measurements.* St. Martin's Press Inc, New York, second revised edition, 1971.

[81] Genichiro Kinoshita, Masanori Idesawa, and Shigeo Naomi. Robotic range sensor with projection of bright ring pattern. *Journal of Robotic Systems*, 3(3):249–257, 1986.

[82] Miles V. Klein and Thomas E. Furtak. *Optics.* John Wiley & Sons, New York, second edition, 1986.

[83] Eric Krotkov. Focusing. *International Journal of Computer Vision*, 1(3):223–237, October 1987.

[84] Eric Krotkov. *Focusing.* Technical Report MS-CIS-86-22, GRASP LAB 63, Department of Computer and Information Science, Moore School, University of Pennsylvania, April 1986.

[85] Eric Krotkov, Filip Fuma, and John Summers. An agile stereo camera system for flexible image acquisition. *IEEE Journal of Robotics and Automation*, 4(1):108–113, February 1988.

[86] Eric Krotkov and Ralf Kories. Adaptive control of cooperating sensors: focus and stereo ranging with an agile camera system. In *Proceedings 1988 IEEE International Conference on Robotics and Automation*, pages 548–553, April 1988.

[87] Eric Krotkov and Ralf Kories. Cooperative focus and stereo ranging. In *Proceedings The Fourth Conference on Artificial Intelligence Applications*, pages 76–81, March 1988.

[88] Eric Krotkov and Jean-Paul Martin. Range from focus. In *International Conference on Robotics and Automation*, pages 1093–1098, IEEE, 1986.

[89] Eric Krotkov, John Summers, and Filip Fuma. Computing range with an active camera system. In *Proceedings of the International Joint Conference on Pattern Recognition*, pages 1156–1158, IEEE, 1986.

[90] Eric P. Krotkov and Jean-Paul Martin. *Range from Focus.* Technical Report MS-CIS-86-09, GRASP LAB 59, Department of Computer and Information Science, Moore School/D2, University of Pennsylvania, January 1986.

[91] Eric Paul Krotkov. *Exploratory Visual Sensing For Determining Spatial Layout With An Agile Stereo Camera System.* Technical Report MS-CIS-87-29, GRASP LAB 101, Department of Computer and Information Science, School of Engineering and Applied Science, University of Pennsylvania, April 1987.

[92] Eric Paul Krotkov. *Exploratory Visual Sensing For Determining Spatial Layout With An Agile Stereo Camera System.* PhD thesis, Computer and Information Science, University of Pennsylvania, 1987.

[93] M. Ali Kujoory, Brian H. Mayall, and Mortimer L. Mendelsohn. Focus-assist device for a flying-spot microscope. *IEEE Transactions on Biomedical Engineering*, BME-20(2):126–132, March 1973.

[94] Richard A. Langlais, Francis T. Ogawa, and Dennis J. Wilwerding. Method and apparatus for determining focus direction and amount. Canadian # 1164570, March 27 1984.

[95] B.P. Lathi. *Signals, Systems and Communication.* John Wiley & Sons, New York, 1965.

[96] James R. Leger and Michael A. Snyder. Real-time depth measurement and display using Fresnel diffraction and white-light processing. *Applied Optics*, 23(10):1655–1670, May 1984.

[97] Fang Lei and H.J. Tiziani. A comparison of methods to measure the modulation transfer function of aerial survey lens systems from image structures. *Photogrammetric Engineering and Remote Sensing*, 54(1):41–46, January 1988.

[98] Reimar K. Lenz and Roger Y. Tsai. Techniques for calibration of the scale factor and image center for high accuracy 3-D machine vision metrology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(5):713–720, September 1988.

[99] Norman Levinson. The wiener RMS (root mean square) error criterion in filter design and prediction. In Norbert Wiener, editor, *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*, pages 129–148, John Wiley & Sons, New York, 1949.

[100] Norman Levinson. The wiener RMS (root mean square) error criterion in filter design and prediction. *Journal of Mathematics and Physics*, 25(4):261–278, January 1947.

[101] Guido Ligthart and Frans C.A. Groen. A comparison of different autofocus algorithms. In *Proceedings of the International Joint Conference on Pattern Recognition*, pages 597–600, IEEE, 1982.

[102] David G. Lowe. The viewpoint consistency constraint. *International Journal of Computer Vision*, 1(1):57–72, 1987.

[103] Ren C. Luo and Min-Hsiung Lin. Issues and approaches of automatic focusing algorithms for intelligent robot eye-in-hand system. *Journal of Robotic Systems*, 4(4):459–476, 1987.

[104] Ren C. Luo, Min-Hsiung Lin, and Ralph S. Scherp. Design and implementation of new auto-focusing algorithms for intelligent robot eye-in-hand system. In *Proceedings of the International Workshop on Industrial Automation Systems: Seiken Symposium*, pages 11–16, IEEE, February 4–6 1987.

[105] S. Lawrence Marple Jr. A tutorial overview of modern spectral estimation. In *Proceedings, International Conference on Acoustics Speech and Signal Processing*, pages 2152–2157, IEEE, May 1989.

[106] D. Marr and E. Hildreth. Theory of edge detection. *Proceedings of the Royal Society of London, B*, 207:187–217, 1980.

[107] David Marr. *Vision.* W.H. Freeman and Company, New York, 1982.

[108] D.C. Mason and D.K. Green. Automatic focusing of a computer-controlled microscope. *IEEE Transactions on Biomedical Engineering*, BME-22(4):312–317, July 1975.

[109] Mortimer L. Mendelsohn and Brian H. Mayall. Computer-oriented analysis of human chromosomes–III focus. *Computers in Biology and Medicine*, 2:137–150, 1972.

[110] Richard A. Muller and Andrew Buffington. Real-time correction of atmospherically degraded telescope images through image sharpening. *Journal of the Optical Society of America*, 64(9):1200–1210, September 1974.

[111] Yasuo Nakada. Apparatus for detecting distance to an object. United States Patent # 4690549, September 1 1987.

[112] Takashi Nishibe et al. Range finder. United States Patent # 4611910, September 16 1986.

[113] David Nitzan. Three-dimensional vision structure for robot applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(3):291–309, May 1988.

[114] Allen Nussbaum and Richard A. Phillips, editors. *Contemporary Optics for Scientists and Engineers. Solid State Physical Electronics Series*, Prentice-Hall, New Jersey, 1976.

[115] S. Ohteru, H. Kobayashi, and T. Kato. Eyes of the wabot. In K.S. Fu and Julius T. Tou, editors, *Learning Systems and Intelligent Robots*, pages 343–364, Plenum Press, New York, 1973.

[116] Alan V. Oppenheim, George V. Frisk, and David R. Martinez. An algorithm for the numerical evaluation of the hankel transform. *Proceedings of the IEEE*, 66(2):264–265, February 1978.

[117] A. Pentland, T. Darrell, M. Turk, and W. Huang. A simple, real-time range camera. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 256–261, 1989.

[118] Alex P. Pentland. Depth of scene from depth of field. In *Proceedings, Image Understanding Workshop*, pages 253–259, April 1982.

[119] Alex P. Pentland. The focal gradient: optics ecologically salient. *Supplement to Investigative Ophthalmology and Visual Science*, 243, April 1985.

[120] A.P. Pentland. A new sense of depth of field. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9:523–531, July 1987.

[121] A.P. Pentland. A new sense of depth of field. In *Ninth International Joint Conference on A.I.*, pages 988–994, August 1985.

[122] Michael Potmesil and Indranil Chakravarty. Synthetic image generation with a lens and aperture camera model. *ACM Transactions on Graphics*, 1(2):85–108, April 1982.

[123] William H. Press, Brian P. Flannery, Saul A. Teukolsky, and William T. Vetterling. *Numerical Recipes.* Cambridge University Press, 1987.

[124] K. Preston, Jr. Automated microscopy for cytological analysis. In R. Barer and V.E. Cosslett, editors, *Advances in Optical and Electron Microscopy*, pages 43–93, Academic Press, New York, 1973.

[125] R.W. Reading. *Binocular Vision.* Butterworth Publishers, Boston, 1983.

[126] M. Rioux, P. Boulanger, and T. Kasvand. Segmentation of range images using sine wave coding and Fourier transformation. *Applied Optics*, 26(2):287–293, January 1987.

[127] Marc Rioux and François Blais. Compact three-dimensional camera for robotic applications. *Journal of the Optical Society of America*, 3(9):1518–1521, September 1986.

[128] Azriel Rosenfeld, editor. *Techniques for 3-D Machine Perception.* Volume 3 of *Machine Intelligence and Pattern Recognition Series*, Elsevier Science Publishers, Amsterdam, 1986.

[129] Azriel Rosenfeld and Avinash C. Kak. *Digital Picture Processing.* Volume 1, Academic Press, New York, second edition, 1982.

[130] M.R. Sayeh, M. Daneshdoost, and F. Pourboghrat. A computational model for sensing depth from a single 2-D image. In *Optics, Illumination, and Image Sensing for Machine Vision II*, pages 28–32, SPIE, 1987. Vol. 850.

[131] J.F. Schlag, A.C. Sanderson, C.P. Neumann, and F.C. Wimberly. *Implementation of Automatic Focusing Algorithms for a Computer Vision System with Camera Control.* Technical Report CMU-RI-TR-83-14, Carnegie-Mellon University, August 1983.

[132] William F. Schreiber. *Fundamentals of Electronic Imaging Systems.* Volume 15 of *Springer Series in Information Sciences*, Springer-Verlag, Berlin, 1986.

[133] Henry Stark, editor. *Image Recovery: Theory and Application.* Academic Press Inc., 1987.

[134] Per A. Stokseth. Properties of a defocused optical system. *Journal of the Optical Society of America*, 59(10):1314–1321, October 1969.

[135] T.C. Strand. Optical three-dimensional sensing for machine vision. *Optical Engineering*, 24(1):33–40, January/February 1985.

[136] Murali Subbarao. Parallel depth recovery by changing camera parameters. In *Second International Conference on Computer Vision*, pages 149–155, IEEE Computer Society, December 1988.

[137] Muralidhara Subbarao. *Direct Recovery of Depth-map.* Technical Report 87-02, Computer Vision and Graphics Laboratory, Department of Electrical Engineering, State University of New York at Stony Brook, February 1987.

[138] Muralidhara Subbarao. Direct recovery of depth-map I: Differential methods. In *Proceedings of the IEEE Computer Society Workshop on Computer Vision*, pages 58–65, December 1987.

[139] Muralidhara Subbarao. *Direct Recovery of Depth-map II: A New Robust Approach.* Technical Report 87-03, Computer Vision and Graphics Laboratory, Department of Electrical Engineering, State University of New York at Stony Brook, April 1987.

[140] Muralidhara Subbarao. Efficient depth recovery through inverse optics. In Hebert Freeman, editor, *Machine Vision for Inspection and Measurement*, pages 101–126, Academic Press Inc., 1989.

[141] Muralidhara Subbarao. Method and apparatus for determining the distances between surface-patches of a three-dimensional spatial scene and a camera system. European Patent Application # 88310672.6, November 11 1988.

[142] Muralidhara Subbarao. *Progress in Research on Direct Recovery of Depth and Motion.* Technical Report 87-04, Computer Vision and Graphics Laboratory, Department of Electrical Engineering, State University of New York at Stony Brook, May 1987.

[143] Muralidhara Subbarao and Natarajan Gurumoorthy. Depth recovery from blurred edges. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 498–503, 1988.

[144] Bruce W. Suter and Stanley R. Deans. A hankel transform algorithm for uniformly sampled data. In *Proceedings, International Conference on Acoustics, Speech, and Signal Processing*, pages 1542–1545, IEEE, March 26–29, 1985.

[145] J.M. Tenenbaum. *Accommodation in Computer Vision.* PhD thesis, Stanford University, November 1970.

[146] A.N. Tihonov. Solution of incorrectly formulated problems and the regularization method. *Soviet Mathematics, Translation of Doklady Akademii Nuak SSSR*, 4:1035–1038, 1963.

[147] Vincent Torre and Tomaso A. Poggio. On edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(2):147–163, March 1986.

[148] William F. Trench. An algorithm for the inversion of finite toeplitz matrices. *Journal of the Society for Industrial and Applied Mathematics*, 12(3):515–522, September 1964.

[149] Roger Y. Tsai. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE Journal of Robotics and Automation*, RA-3(4):323–344, August 1987.

[150] J.K. Tsotsos. Image understanding. In S. Shapiro, editor, *Encyclopedia of AI*, pages 389–409, J. Wiley, New York, 1987.

[151] Mati Wax and Thomas Kailath. Efficient inversion of toeplitz-block toeplitz matrix. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-31(5):1218–1221, October 1983.

[152] E. Wigner. On the quantum correction for thermodynamic equilibrium. *The Physical Review*, 40(Second Series):749–759, June 1 1932.

[153] R.J. Woodham. Oral communication, January 12, 1989.

[154] Hiroharu Yamamoto, Toshio Hara, and Kunihiko Edamatsu. Autofocus camera system for FA. In *Optics, Illumination, and Image Sensing for Machine Vision II*, pages 28–32, SPIE, 1987. Vol. 850.

[155] R. Yarlagadda and B.N. Suresh Babu. A note on the application of FFT to the solution of a system of toeplitz normal equations. *IEEE Transactions on Circuits and Systems*, CAS-27(2):151–154, February 1980.