
Dissertations, Theses, and Masters Projects

Theses, Dissertations, & Master Projects

1989

Small-kernel image restoration

Stephen Edward Reichenbach
College of William & Mary - Arts & Sciences

Follow this and additional works at: <https://scholarworks.wm.edu/etd>



Part of the [Computer Sciences Commons](#)

Recommended Citation

Reichenbach, Stephen Edward, "Small-kernel image restoration" (1989). *Dissertations, Theses, and Masters Projects*. Paper 1539623783.

<https://dx.doi.org/doi:10.21220/s2-bytc-m947>

This Dissertation is brought to you for free and open access by the Theses, Dissertations, & Master Projects at W&M ScholarWorks. It has been accepted for inclusion in Dissertations, Theses, and Masters Projects by an authorized administrator of W&M ScholarWorks. For more information, please contact scholarworks@wm.edu.

INFORMATION TO USERS

The most advanced technology has been used to photograph and reproduce this manuscript from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

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

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

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book. These are also available as one exposure on a standard 35mm slide or as a 17" x 23" black and white photographic print for an additional charge.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

U·M·I

University Microfilms International
A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
313/761-4700 800/521-0600

Order Number 9008530

Small-kernel image restoration

Reichenbach, Stephen Edward, Ph.D.

The College of William and Mary, 1989

U·M·I
300 N. Zeeb Rd.
Ann Arbor, MI 48106

SMALL-KERNEL IMAGE RESTORATION

A DISSERTATION

PRESENTED TO

THE FACULTY OF THE DEPARTMENT OF COMPUTER SCIENCE
THE COLLEGE OF WILLIAM AND MARY IN VIRGINIA

IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

BY

STEPHEN E. REICHENBACH


1989


APPROVAL SHEET

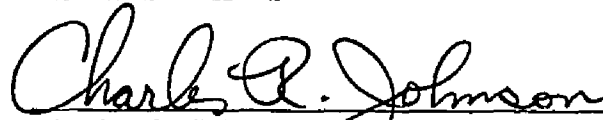
This dissertation is submitted in partial fulfillment of
the requirements for the degree of
Doctor of Philosophy



Stephen E. Reichenbach, Author

Approved, July 1989


Stephen K. Park, Dissertation Director


Friedrich O. Huck


Charles R. Johnson


John P. Kearns


Keith W. Miller

DEDICATION

This dissertation is dedicated to my parents George and Patricia Reichenbach, my wife Ellen, and my children Patrick and Katie. I thank God for all he has given me.

Contents

1	Introduction	2
2	Mathematical Preliminaries	12
2.1	Digital Images	12
2.2	Imaging Systems	14
2.3	Pointwise Operations	15
2.4	Linear, Shift-Invariant Systems	16
2.5	Matrix Representation of Linear Systems	22
2.6	Complex Arithmetic	23
2.7	Spatial Frequency Analysis	25
2.8	Discrete Fourier Transform	32
2.9	Continuous Fourier Transform	34
2.10	Inverse Problems	36
2.11	Image Statistics	37
2.12	Image Ensembles	40
3	The Imaging Process	44
3.1	Images of Physical Scenes	44
3.2	Image Acquisition	44
3.2.1	Image Formation	45
3.2.2	Spatial Sampling	49
3.2.3	Brightness Quantization and Noise	53
3.2.4	Simplified Acquisition Model	54
3.3	Display	56

4	Traditional Restoration Techniques	60
4.1	Problems, Solutions, and Algorithms	60
4.2	Inverse Filter	61
4.3	Wiener Filter	64
4.4	Constrained-Least-Squares Filters	65
4.5	Minimizing the Composite Point Spread Function	66
4.6	End-to-End Wiener Filter	68
5	Small-Kernel Restoration	71
5.1	Small-Kernel Convolution	71
5.2	Small Kernels from Large Kernels	74
5.3	Minimizing the Composite Point Spread Function	76
5.4	Small, Least-Squares Kernels	79
6	Optimal Small-Kernel Restoration	82
6.1	One-Dimensional Discrete Restoration	83
6.1.1	Formulation	83
6.1.2	Discrete Wiener Filter	86
6.1.3	Discrete Spatially-Constrained Kernel	87
6.1.4	Theorems	93
6.2	End-to-End Restoration	96
6.2.1	Formulation	96
6.2.2	End-to-End Wiener Filter	99
6.2.3	End-to-End, Spatially-Constrained Kernel	99
6.3	Two-Dimensional Restoration	101
6.3.1	Two-Dimensional Wiener Filter	101
6.3.2	Two-Dimensional, Spatially-Constrained Kernel	102
6.4	Asides	104
6.4.1	Symmetry	104
6.4.2	Generalized Inverse	105

7	Artificial Scenes and Simulated Imaging	107
7.1	Experimental Methodology	107
7.2	Artificial Scenes	108
7.2.1	Fourier Scenes	109
7.2.2	Digital Scenes	115
7.2.3	Relationship Between Fourier Scenes and Digital Scenes	116
7.3	Simulated Imaging	117
7.3.1	Image Formation	118
7.3.2	Sampling	120
7.3.3	Noise	121
7.3.4	Display	122
8	Experimental Results	126
8.1	One-Dimensional Simulation Results	126
8.2	Two-Dimensional Simulation Results	142
8.3	Restoration of Digital Images	149
9	Conclusions	157
A	Useful Functions	162
B	Characterizing Digital Image Acquisition Devices	167
B.1	Introduction	167
B.2	The Traditional Knife-Edge Technique	168
B.2.1	Theoretical Basis	168
B.2.2	Dealing with Noise	170
B.2.3	Problems with Undersampling	171
B.3	Sampled Systems	172
B.4	Two-Dimensional Estimates	178
B.5	Simulation Results	181
B.6	Experimental Results	185
B.7	Conclusion	190

ACKNOWLEDGEMENTS

I am grateful to Professor Stephen K. Park for teaching me the importance of first principles and guiding me in the process of this research. I am indebted to Dr. Keith W. Miller and my wife Ellen M. Reichenbach for their comments and suggestions about the manuscript. I would like to acknowledge the help of Mr. Friedrich O. Huck at the NASA Langley Research Center and the support of NASA Graduate Researcher Fellowship Grant Number NGT-50117.

List of Figures

1.1	Restoration Example	3
2.1	Digital Image	13
2.2	Parametric Imaging System	14
2.3	Complex Numbers	25
2.4	Periodic Sawtooth and Its Periodic Spectrum	30
2.5	Sawtooth Spatial-Frequency Decomposition	31
3.1	Imaging Process	44
3.2	Image Acquisition	45
3.3	Image Formation	45
3.4	Perspective Projection	46
3.5	Example Point Spread Function and Optical Transfer Function	47
3.6	Spatial Blurring	48
3.7	Frequency Attenuation	48
3.8	Spatial Sampling	50
3.9	Idealized Model of Sensor Point Spread Function	51
3.10	Spatial Aliasing	52
3.11	Frequency Folding	53
3.12	Simplified Imaging Acquisition Model	55
3.13	Display Blur and Ripple	56
3.14	Simplified Imaging Display Model	57
5.1	Spatial Domain and Frequency Domain Processing	72
5.2	Execution Times	73

6.1	Discrete Image Formation and Spatial Restoration	83
6.2	Discrete Image Formation and Frequency Restoration	88
6.3	End-to-End Imaging Model and Spatial Restoration	97
7.1	Simulation Processing	119
8.1	Artificial Scene Statistics	128
8.2	Representative Artificial Scenes	129
8.3	Simulated Acquisition Functions	130
8.4	Representative Sensor Noise	132
8.5	Simulated Display Functions	133
8.6	Simulated End-to-End Processing of a Representative Scene .	135
8.7	Restoration Functions	136
8.8	Representative Restoration Results	138
8.9	Relative Restoration Error (Low Frequency Scenes)	139
8.10	Relative Restoration Error (Medium Frequency Scenes) . . .	140
8.11	Relative Restoration Error (High Frequency Scenes)	141
8.12	Two-Dimensional Kernel Elements	143
8.13	Restoration for Low Blur and High Noise	144
8.14	Restoration for Moderate Blur and Moderate Noise	145
8.15	Restoration for High Blur and Low Noise	146
8.16	Relative Restoration Error	148
8.17	Relative Restoration Error (Normalized Images)	150
8.18	Acquisition Functions	152
8.19	Restoration of In-Focus Image	153
8.20	Restoration of Defocused Image	154
8.21	Restoration of Extremely Defocused Image	155
A.1	Some Useful Functions	163
B.1	Spatial Derivatives in the Fourier Frequency Domain	170
B.2	Example System and Estimates	173

B.3	Registering Scans With Shifted Knife Edge	174
B.4	Intervals for Registering Scan Lines	176
B.5	CCD Detector Array	179
B.6	Idealized Horizontal and Vertical Spatial Responses	180
B.7	Diagonal Knife-Edge Scans	181
B.8	Simulated System Function Along- x	183
B.9	Scans	184
B.10	Relative MSE of Estimates of System Function Along- x	186
B.11	Analysis of System Separability	187
B.12	Estimates of CCD System OTF	188
B.13	Square-Wave Estimated and Observed Contrast	189

List of Tables

2.1	Fourier Transform Theorems	33
2.2	Fourier Transform Relationships	34
2.3	Fourier Transform Pairs	35
8.1	Two-Dimensional Kernel Sizes	147

ABSTRACT

The goal of image restoration is to remove degradations that are introduced during image acquisition and display. Although image restoration is a difficult task that requires considerable computation, in many applications the processing must be performed significantly faster than is possible with traditional algorithms implemented on conventional serial architectures. As demonstrated in this dissertation, digital image restoration can be efficiently implemented by convolving an image with a small kernel. Small-kernel convolution is a local operation that requires relatively little processing and can be easily implemented in parallel. A small-kernel technique must compromise effectiveness for efficiency, but if the kernel values are well-chosen, small-kernel restoration can be very effective.

This dissertation develops a small-kernel image restoration algorithm that minimizes expected mean-square restoration error. The derivation of the mean-square-optimal small kernel parallels that of the Wiener filter, but accounts for explicit spatial constraints on the kernel. This development is thorough and rigorous, but conceptually straightforward: the mean-square-optimal kernel is conditioned only on a comprehensive end-to-end model of the imaging process and spatial constraints on the kernel. The end-to-end digital imaging system model accounts for the scene, acquisition blur, sampling, noise, and display reconstruction. The determination of kernel values is directly conditioned on the specific size and shape of the kernel. Experiments presented in this dissertation demonstrate that small-kernel image restoration requires significantly less computation than a state-of-the-art implementation of the Wiener filter yet the optimal small-kernel yields comparable restored images.

The mean-square-optimal small-kernel algorithm and most other image restoration algorithms require a characterization of the image acquisition device (i.e., an estimate of the device's point spread function or optical transfer function). This dissertation describes an original method for accurately determining this characterization. The method extends the traditional knife-edge technique to explicitly deal with fundamental sampled system considerations of aliasing and sample/scene phase. Results for both simulated and real imaging systems demonstrate the accuracy of the method.

Small-Kernel Image Restoration

Chapter 1

Introduction

One of the first applications of digital image processing was the restoration of images from NASA space flights in the 1960's. Scientists at the Jet Propulsion Laboratory (JPL) realized that computer processing could significantly improve the clarity and apparent resolution of television images from the Ranger lunar missions. The success of this work[1] and the high-visibility of the space program provided a dramatic impetus to digital image processing research. Since those early results, digital image processing equipment and techniques have improved tremendously[2,3,4,5,6,7]. Today, applications include biomedicine, remote sensing, astronomy, computer-aided manufacturing, facsimile (fax) reproduction, desk-top publishing, entertainment, advertising, surveillance, and night vision. Emerging technologies in digital photography, digital television, and telerobotics will further expand the field of digital image processing.

Image restoration is a central problem in many imaging applications. Devices for acquiring and displaying images inevitably introduce degradations and produce imperfect pictures. Figure 1.1A shows a high-resolution picture of several one-dollar bills. Figure 1.1B illustrates the picture that an imaging system with lower spatial resolution might produce. The goal of image restoration is to process degraded images (such as Figure 1.1B) in order to produce pictures with greater fidelity (like Figure 1.1A). In a digital imaging system, the image is *digitized* (converted to digital data) during acquisition. Digital image restoration and any other digital processing is performed on this data. The display device then reconstructs a picture from the processed data.

Digital image restoration is a difficult task, one that challenges the most advanced computer systems. The problem is complex, typical images contain a great deal of data, and many applications require real-time processing. A number of well-known digital image restoration techniques can effectively

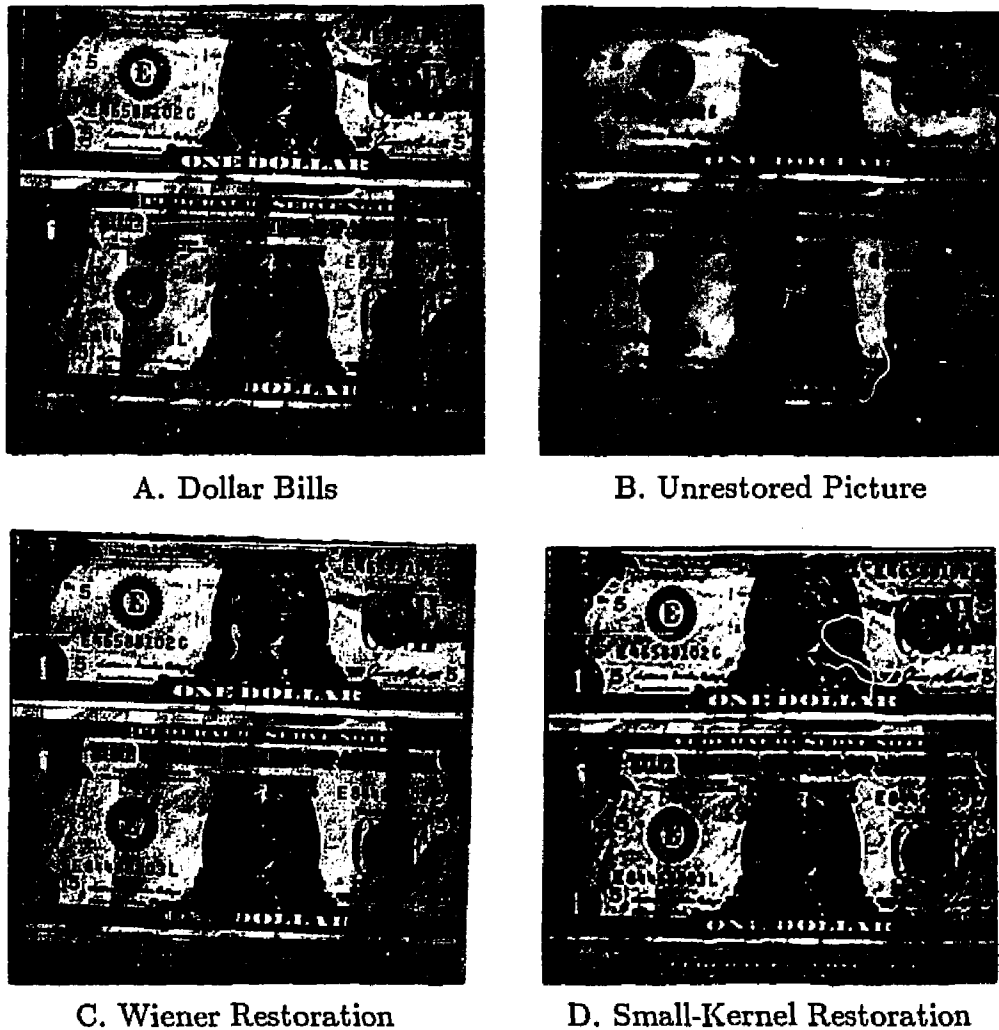


Figure 1.1: Restoration Example

improve the fidelity of an image. However, most of these techniques (e.g., the inverse filter[1,8], the Wiener filter[9,10,11,12], the constrained-least-squares filter[13,14,15], and two filters designed to control the composite function of the acquisition device and restoration process[16,17]) are *global* operations—each value of the restored image is a function of *all* of the values of the input image. Global operations require a great deal of processing. Though the development of the fast Fourier transform algorithm has made these techniques practical in some applications, many other applications require a more efficient approach.

The image restoration algorithm described in this dissertation is a *local* operation—each value of the restored image is a function of only a few neigh-

boring pixel values in the input image. Restoration is efficiently implemented by convolving the image with a small kernel. (The convolution operation is a weighted sum of the input pixels. The small kernel delimits the neighborhood and defines the weights associated with each neighboring pixel's value.) Small-kernel convolution requires significantly fewer calculations than global operations and can be more easily implemented in parallel. This approach also facilitates adaptive processing.

Small-kernel restoration must compromise effectiveness for efficiency, but this dissertation demonstrates that if the kernel values are well-chosen then small kernels can be nearly as effective as global operations. Three significant design issues bear directly on the effectiveness of a small-kernel restoration algorithm: the underlying model of the imaging process, the imposition of spatial constraints on the kernel, and the operative measure of restoration quality.

Modeling the Imaging Process. An image restoration algorithm is designed to correct imperfections that are introduced during the imaging process. The underlying model of the imaging process should capture the significant components of the actual process. If the model is incomplete, the algorithm may be ineffective. For example, traditional formulations of the Wiener filter for image restoration are based on a model that includes the scene, acquisition blur, and noise[2,3,4,5,6,7]. This model fails to account for the significance of sampling and display reconstruction in digital systems. A recent re-formulation of the Wiener filter that is based on an end-to-end model that accounts for sampling and display is more effective[11,12]. The small-kernel algorithm developed in this dissertation is conditioned on this comprehensive, end-to-end model of a digital imaging system. The model accounts for the character of the scene, acquisition blur, sampling, noise, and display reconstruction.

Imposing Spatial Constraints. There are several approaches to designing small restoration kernels. A popular approach is to generate a large kernel using a global technique and then window a small kernel from it[18]. Unfortunately, the windowing operation can introduce undesirable side-effects. If spatial constraints are imposed *ad hoc*, after the kernel values are determined, the constraints are an uncontrolled variable. The kernel values should be conditioned on the constraints. Another approach is to modify the global technique in order to reduce the impact of subsequent windowing[19,20,21], but this two-step approach does not directly coordinate the determination of the kernel values and the windowing operation. A third approach is to design the kernel based on the shape of the composite function of the acquisition device and restoration process[22,23,24,25]. This approach integrates explicit constraints on the kernel into the design, but does not

account for the character of the scene (i.e., the model is incomplete). Therefore, these algorithms are somewhat difficult to tune for specific applications and do not adapt to changing scenes. Also, because these algorithms employ several constraints that relate only indirectly to the character of the restored image, they are conceptually complicated (i.e., restoration quality is difficult to control). Because the nature of visual quality is illusive[26], restoration is somewhat of an art and conceptual simplicity is important. In the algorithm developed in this dissertation, the small kernel is designed subject to explicit spatial constraints. The specific size and shape of the kernel is an *a priori* constraint.

Measuring Restoration Quality. Because some aspects of the imaging process cannot be specified precisely (e.g., noise), it is impossible to perfectly restore images. Therefore a restoration technique must estimate the ideal image. In most applications, judging the relative quality of alternative estimates is a subjective process. Without human intervention, digital image restoration techniques must rely on mathematical measures of restoration quality that correlate with (but are not identical to) subjective judgements. For example, subject to the image model, the Wiener filter minimizes the expected mean-square restoration error (MSRE). Using MSRE allows the straightforward derivation of an optimal filter. The small-kernel algorithm developed in this dissertation is mean-square optimal subject to the image model *and* the spatial constraints on the kernel.

The derivation of the mean-square-optimal small kernel parallels that of the Wiener filter. This development is thorough and rigorous, but conceptually straightforward—the optimal kernel is conditioned only on the model, the spatial constraints, and the MSRE criterion. The values of the constrained convolution kernel that minimize the expected MSRE are determined using a frequency analysis of the end-to-end imaging system. Minimizing mean-square error with respect to the kernel elements yields a system of linear equations in the kernel values. The optimal kernel is the solution of this system of equations.

The experiments presented in this dissertation demonstrate that small-kernel image restoration requires significantly less computation than a state-of-the-art implementation of the Wiener filter yet the optimal small-kernel yields comparable results. For example, the image displayed in Figure 1.1C was restored with the Wiener filter. The image displayed in Figure 1.1D was restored with the optimal 3×3 kernel. Both of the restored images are clearly superior to the unrestored image in Figure 1.1B. The small kernel convolution required only 22% as much serial processing as the Wiener filter, yet the resulting image is nearly as good. Moreover, small kernel convolution can be more easily implemented in parallel than the global Wiener filter.

The organization of the dissertation follows a standard approach to software development: define the problem, specify the solution, develop and implement an algorithm, and test the implementation. There are also two appendices.

Chapter 2: Mathematical Preliminaries

This chapter introduces terminology and mathematical notation. It also provides a brief review of some fundamental mathematics including image operations, Fourier analysis, image statistics, and stochastic processes. In contrast to the presentation found in many texts on digital image processing, the discrete data structures of digital images are emphasized. This material can be skimmed by those familiar with these topics.

Chapter 3: The Imaging Process

Chapter 3 examines the nature of the image restoration problem—the degradations introduced by the imaging process. To some degree, these degradations vary from application to application and from system to system. Fortunately, these degradations can be modeled fairly accurately and simply. The end-to-end model of the imaging system described in this chapter is more complete than that used by many researchers.

Chapter 4: Traditional Restoration Techniques

Chapter 4 reviews several traditional techniques for image restoration. Different techniques use different models of the imaging process and criteria for image restoration. All of the techniques described in this chapter are global techniques. The most important of these techniques is the Wiener filter.

Chapter 5: Small-Kernel Restoration

This chapter is a critical review of techniques for reducing the computation associated with traditional restoration methods. These methods constrain the size of the region in the input image used to determine each output value. The key issue is how the kernel constraints are imposed in the design process.

Chapter 6: Optimal Small-Kernel Restoration

Chapter 6 details an original derivation of mean-square-optimal, small restoration kernels. Although the small kernel is applied directly to the image (in the spatial domain), the optimal values are derived using a Fourier frequency-domain analysis. The development accounts for the effects of sampling and display reconstruction—important components of digital imaging systems that are often ignored. The derivation

parallels that of the Wiener filter, but explicitly accounts for spatial constraints on the kernel.

Chapter 7: Artificial Scenes and Simulated Imaging

This chapter describes a software environment for generating artificial scenes and simulating the imaging process. Artificial scenes model the important characteristics of real scenes, but are exactly known, highly controlled, and easily communicated. Simulated imaging eliminates the confounding effects of inexact estimates of the characteristics of real acquisition and display devices, provides a flexible imaging environment, and accommodates replication of the imaging process. Artificial scenes and simulated imaging allow greater exactness, control, and portability for image processing research. These rigorous procedures provide quantitative benchmarks for image processing research.

Chapter 8: Results

Results for one-dimensional simulations, two-dimensional simulations, and real images are presented. These results indicate that optimal small kernels can achieve much of the success of the Wiener filter, but with significantly less computation. The results of the experiments also suggest that optimal small-kernel restoration may be more robust than the Wiener filter. This possibility merits further research.

Chapter 9: Conclusions

The concluding chapter focuses on the practical aspects of image restoration. A rigorous development and effective technique are important, but so are efficiency and simplicity. Optimal small-kernel restoration meets all of these demands. Unresolved questions and topics for future research are discussed.

Appendix A: Useful Functions

This appendix describes several useful functions that are used extensively in image processing. This material is intended for those not already familiar with image processing.

Appendix B: Characterizing Digital Image Acquisition Devices

Nearly every restoration technique requires an accurate characterization of the acquisition device. Appendix B describes an original technique for accurately characterizing the system functions of digital image acquisition devices. Previous methods were developed for analog systems and presume oversampling. Because digital acquisition devices are *designed* to undersample[27], these techniques do not give accurate estimates for digital systems. The technique described in this appendix

is specifically designed for digital systems and accurately estimates the system transfer function beyond the sampling passband.

References

- [1] R. Nathan. *Digital Video Handling*. Technical Report 32-877, NASA, 1966.
- [2] William. K. Pratt. *Digital Image Processing*. John Wiley and Sons, New York, NY, 1978.
- [3] Kenneth R. Castleman. *Digital Image Processing*. Prentice-Hall, Englewood Cliffs, NJ, 1979.
- [4] Azriel Rosenfeld and Avinash C. Kak. *Digital Picture Processing*. Academic Press, Orlando, FL, second edition, 1982.
- [5] William B. Green. *Digital Image Processing: A Systems Approach*. Van Nostrand Reinhold, New York, NY, 1983.
- [6] Rafael C. Gonzalez and Paul Wintz. *Digital Image Processing*. Addison-Wesley, Reading, MA, second edition, 1987.
- [7] H. C. Andrews and B. R. Hunt. *Digital Image Restoration*. Prentice-Hall, Englewood Cliffs, NJ, 1977.
- [8] James L. Harris, Sr. Image evaluation and restoration. *Journal of the Optical Society of America*, 56(5):569–574, 1966.
- [9] C. W. Helstrom. Image restoration by the method of least squares. *Journal of the Optical Society of America*, 57(3):297–303, 1967.
- [10] D. Slepian. Linear least-squares filtering of distorted images. *Journal of the Optical Society of America*, 57(7):918–922, 1967.
- [11] Friedrich O. Huck, Carl L. Fales, Nesim Haylo, Richard W. Samms, and Kathryn Stacy. Image gathering and processing: Information and fidelity. *Journal of the Optical Society of America A*, 2(10):1644–1666, 1985.
- [12] Carl L. Fales, Friedrich O. Huck, Judith A. McCormick, and Stephen K. Park. Wiener restoration of sampled image data: end-to-end analysis. *Journal of the Optical Society of America A*, 5(3):300–314, 1988.
- [13] David L. Phillips. A technique for the numerical solution of certain integral equations of the first kind. *Journal of the Association for Computing Machinery*, 9(1):84–97, 1962.

- [14] S. Twomey. On the numerical solution of fredholm integral equations of the first kind by inversion of the linear system produced by quadrature. *Journal of the Association for Computing Machinery*, 10(1):97–101, 1963.
- [15] B. R. Hunt. The application of constrained least squares estimation to image restoration by digital computer. *IEEE Transactions on Computers*, 22(9):805–812, 1973.
- [16] Harvey A. Smith. Improvement of the resolution of a linear scanning device. *SIAM Journal of Applied Mathematics*, 14(1):23–41, 1966.
- [17] G. Backus and F. Gilbert. Uniqueness in the inversion of inaccurate gross earth data. *Philosophical Transactions of the Royal Society of London A*, 266:123–192, 1970.
- [18] Roger J. Arguello, Harvey R. Sellner, and John A. Stuller. Transfer function compensation of sampled imagery. *IEEE Transactions on Computers*, 21(7):812–818, 1972.
- [19] Terry E. Riemer and Clare D. McGillem. Constrained optimization of image restoration filters. *Applied Optics*, 12(9):2027–2029, 1973.
- [20] Terry E. Riemer and Clare D. McGillem. Optimum constrained image restoration filters. *IEEE Transactions on Aerospace and Electronic Systems*, 13(2):136–146, 1977.
- [21] R. W. Schutten and G. F. Vermeij. The approximation of image blur restoration filters by finite impulse responses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(2):176–180, 1980.
- [22] John A. Stuller. An algebraic approach to image restoration filter design. *Computer Graphics and Image Processing*, 1:107–122, 1972.
- [23] B. Roy Frieden. Image restoration by discrete convolution of minimal length. *Journal of the Optical Society of America*, 64(5):682–686, 1974.
- [24] Bahaa E. A. Saleh. Trade off between resolution and noise in restoration by superposition of images. *Applied Optics*, 13(8):1833–1838, 1974.
- [25] Nim-Yau Chu and Clare D. McGillem. Image restoration filters based on a 1-0 weighting over the domain of support of the PSF. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 27(5):457–464, 1979.
- [26] Judith A. McCormick, Rachel Alter-Gartenberg, and Friedrich O. Huck. Image gathering and restoration: Information and visual quality. *Journal of the Optical Society of America A*, 6(7):987–1005, 1989.

- [27] William F. Schreiber. *Fundamentals of Electronic Imaging Systems: Some Aspects of Image Processing*. Springer-Verlag, New York, NY, 1986.

Chapter 2

Mathematical Preliminaries

This chapter briefly introduces the mathematics that are used in later chapters. The presentation focuses on discrete data structures. A number of image processing textbooks provide more comprehensive reviews[1,2,3,4], however virtually all of the popular texts, even those about *digital* image processing, introduce most important concepts in a traditional calculus-style as mathematical operations on continuous functions. If processing is performed on a digital computer, the more natural approach is to introduce concepts using discrete mathematics and treat continuous functions as the continuum limit.

Readers who are already familiar with fundamental digital image processing concepts such as pointwise operations, linear shift-invariant (LSI) systems, spatial frequency analysis, the discrete Fourier transform, image statistics, and two-dimensional stochastic processes can skim this chapter with attention mostly to notational conventions. The field of image processing does not have universally accepted notation for its basic operations; this chapter introduces the notation used in the dissertation.

2.1 Digital Images

Digital images are discrete data structures. The individual elements of a digital image are called *pixels* (picture elements). (They are also called *pels*, *samples*, or just *points*.) The domain of a digital image is a set of locations (usually in two-dimensions). In most systems, the pixel locations are defined by the uniform rectangular pattern illustrated in Figure 2.1. Following the row-major *raster-scan* of a video monitor (left to right, and top to bottom), the x -axis is vertical and increases from top to bottom. The y -axis is horizontal and increases from left to right. The vertical unit distance (Δx) is

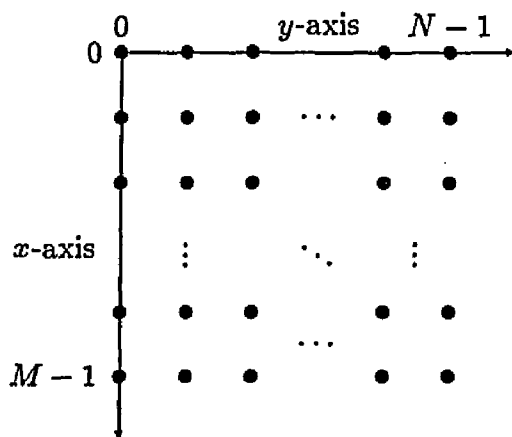


Figure 2.1: Digital Image

the distance between pixel rows and the horizontal unit distance (Δy) is the distance between pixel columns. In the notation of this dissertation, the integer M is the number of pixel rows and the integer N is the number of columns. Values of M and N between 64 and 4096 are typical. Often, M and N are powers-of-two; however, this is not required. The rows are located along the x -axis at integers from 0 to $M - 1$. The columns are located along the y -axis at integers from 0 to $N - 1$. A pixel is uniquely identified by a row and column ordered pair— $[m, n]$ names the pixel in row m and column n . The square brackets distinguish the discrete ordered pairs used to specify pixels from the continuum of (x, y) pairs used to specify arbitrary locations in the image plane.

Associated with each pixel is one or more values (usually measures of light intensity or brightness). Pixel values are stored in a two-dimensional array. The value of a pixel is identified by the array name and the row and column indices— $p[m, n]$ specifies the pixel value of image p at row m and column n . The data type used for pixel values varies. In many applications, each pixel has a single value, but in other applications, each pixel may have several values (e.g., a color image with values for red, green, and blue components) and the data type must be structured accordingly. Typically, the value associated with a pixel is a measure of irradiant light (light falling on a surface), called a *brightness* value. Brightness values are necessarily real, finite, and non-negative. Single-valued, *achromatic* images, such as are displayed on a gray-scale display monitor, are considered in this dissertation. The scale of achromatic brightness values is called a *gray scale*. The values on a gray scale are called *gray levels*. Unsigned, eight-bit integers are commonly used to implement a digital gray scale with integer values 0–255. If

a scale with more values is needed, integers with more bits are used. High-precision, floating-point numbers are required to approximate a continuous scale of brightness values or when extensive arithmetic is performed on the image. The convention used here bounds the gray scale to the L integers in the interval 0 to $L - 1$. A small L can be accommodated by a short integer; a large L requires a data type that can represent more values.

2.2 Imaging Systems

An *imaging system* processes an input image to generate an output image. A hand-held video camera with a small display is an example of a self-contained, end-to-end, imaging system. The input is the light entering the lens. The video camera processes this input and generates an image on the display. This camera system is composed of many imaging components—the lens, photo-sensor, electronic circuits, and display. *Digital systems* process digital input and produce digital output. Most imaging systems are characterized by parameters that can be adjusted (e.g., the focus of a camera or the brightness and contrast controls of a display device). The imaging system model is illustrated in Figure 2.2.

Imaging systems can be described (modeled) by mathematical operations. The imaging system of Figure 2.2 is expressed mathematically as

$$\mathcal{O}_f\{p\} = r \quad (2.1)$$

where the imaging system characterized by f operates on the input image p to produce the output image r . It is sometimes convenient to use an equivalent binary operator notation:

$$p \odot f = r. \quad (2.2)$$

This notation is sensible when, as is often the case, the characterization of the system f has the same mathematical structure as the images.

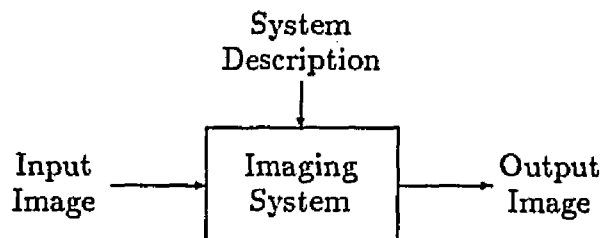


Figure 2.2: Parametric Imaging System

2.3 Pointwise Operations

Pointwise operators act on each element of an image independently. For example, the pointwise sum of an input image p and a random noise process e is

$$r = p + e. \quad (2.3)$$

The value of each pixel in the resulting image is the sum of the values of the corresponding pixels in the input image and noise:

$$r[m, n] = p[m, n] + e[m, n] \quad (2.4)$$

for all $[m, n]$. The other pointwise arithmetic operations (pointwise subtraction, multiplication, and division) are similarly defined.

Scalar arithmetic is a special case of pointwise arithmetic where one or both of the image operands has constant value. For example, a system might create its output image r by boosting the value of all pixels of the input image p by a constant a :

$$r = p + a \quad (2.5)$$

where

$$r[m, n] = p[m, n] + a \quad (2.6)$$

for all $[m, n]$. Similarly a system that caused multiplicative gain would be

$$r = ap \quad (2.7)$$

where

$$r[m, n] = ap[m, n]. \quad (2.8)$$

Common functions such as exponentiation (p^a), logarithm ($\log p$), and absolute value ($|p|$) are also applied in a pointwise manner. For example, multiplicative noise in an image might be more easily removed as additive noise (of a different statistical character) by processing the logarithm of the image:

$$\begin{aligned} \log(r) &= \log(pe) \\ &= \log(p) + \log(e) \end{aligned} \quad (2.9)$$

where

$$\log(r[m, n]) = \log(p[m, n]) + \log(e[m, n]). \quad (2.10)$$

In the digital signal processing literature, where time is the independent variable, systems that operate pointwise on the input are called *memoryless*. The input is processed sequentially—the current value of the output depends only on the current value of the input and none of the previously encountered values needs to be retained. Many imaging systems are memoryless, processing images pixel-by-pixel in raster order.

2.4 Linear, Shift-Invariant Systems

Linearity consists of two aspects: additivity and scaling. *Additivity* means the operation \mathcal{L}_f is distributive with respect to pointwise addition:

$$\mathcal{L}_f \{p_1 + p_2\} = \mathcal{L}_f \{p_1\} + \mathcal{L}_f \{p_2\}. \quad (2.11)$$

Scaling (or *homogeneity*) means the operation is commutative with respect to scalar multiplication (multiplication by a constant):

$$\mathcal{L}_f \{ap\} = a\mathcal{L}_f \{p\} \quad (2.12)$$

where a is a scalar constant. These two aspects of linearity can be expressed in a single equation:

$$\mathcal{L}_f \{a_1p_1 + a_2p_2\} = a_1\mathcal{L}_f \{p_1\} + a_2\mathcal{L}_f \{p_2\} \quad (2.13)$$

and extended by induction to any finite combination of images:

$$\mathcal{L}_f \left\{ \sum_i a_i p_i \right\} = \sum_i a_i \mathcal{L}_f \{p_i\}. \quad (2.14)$$

Spatial shift is the operation of geometric translation. Mathematically, shift is defined as

$$r = \mathcal{S}_{a,b} \{p\} \quad (2.15)$$

where the vertical shift is a pixels and the horizontal shift is b pixels:

$$r[m, n] = p[m - a, n - b] \quad (2.16)$$

for all $[m, n]$. The pixel values of the shifted output image ($0 \leq m < M$ and $0 \leq n < N$) are determined by pixels outside the boundary of the input image. Therefore, some convention is required for assigning values to these pixels. One convention is to assume all pixels outside the borders of the input image have zero value. Another convention is to periodically

extend the input image—that is, tessellate or tile the infinite image plane with replicate images. An infinite image is doubly periodic with periods M and N if and only if for all pixels $[m, n]$,

$$p[m, n] = p[m \bmod M, n \bmod N] \quad (2.17)$$

where \bmod is the usual *modulus* operator.¹ The *fundamental periods* are the smallest M and N for which Equation 2.17 is true. Infinite images that are not periodic are *aperiodic*. In this dissertation, all infinite images are assumed to be periodic unless otherwise stated.

If a system is *shift-invariant*, then it is commutative with respect to spatial shift:

$$\mathcal{O}\{\mathcal{S}_{a,b}\{p\}\} = \mathcal{S}_{a,b}\{\mathcal{O}\{p\}\}. \quad (2.18)$$

A system that does not have this property is *shift-variant*.

The operation of a linear system is *superposition*. Theorem 2.1 gives the mathematical form of the superposition operation.

Theorem 2.1 (Superposition Principle) *Each output value of a digital, linear system can be expressed as the sum of weighted values of the input.*

That is, for every digital, linear imaging system $r = \mathcal{L}_f\{p\}$, there exists a unique four-dimensional array f that weights the elements of the input p to produce the values of the output r in the following way:

$$r[m, n] = \frac{1}{MN} \sum_{m'} \sum_{n'} p[m', n'] f[m, n; m', n'] \quad (2.19)$$

where the summations are taken over the image region (m' from 0 to $M - 1$ and n' from 0 to $N - 1$ or, if the image is periodic, any full period).

Proof: The proof entails a procedure called *impulse decomposition*. The digital impulse is a periodic function that has a value equal to the image size at the origin:

$$\delta[m, n] = \begin{cases} MN & \text{if } [m \bmod M, n \bmod N] = [0, 0] \\ 0 & \text{otherwise.} \end{cases} \quad (2.20)$$

¹The modulus operator is implemented in some computer languages as the division remainder. This is not the same operation for negative numbers. For example, $-1 \bmod N$ is $N - 1$, but the division remainder is -1 .

The input image is decomposed into a linear combination of scaled, shifted impulses:

$$p = \frac{1}{MN} \sum_m \sum_n a_{m,n} \delta_{m,n} \quad (2.21)$$

where the scalar constants are the input image values:

$$a_{m,n} = p[m,n] \quad (2.22)$$

and the component images are shifted digital impulses:

$$\delta_{m,n} = \mathcal{S}_{m,n} \{\delta\}. \quad (2.23)$$

Because the system is linear, the output of the weighted sum is equal to the weighted sum of the system responses to the component impulses (as in Equation 2.14):

$$\begin{aligned} r &= \mathcal{L}_f \{p\} \\ &= \mathcal{L}_f \left\{ \frac{1}{MN} \sum_{m'} \sum_{n'} a_{m',n'} \delta_{m',n'} \right\} \\ &= \frac{1}{MN} \sum_{m'} \sum_{n'} a_{m',n'} \mathcal{L}_f \{\delta_{m',n'}\}. \end{aligned} \quad (2.24)$$

Defining the four-dimensional array f as the system response at pixel $[m,n]$ to an input impulse at $[m',n']$ gives the desired form:

$$r[m,n] = \frac{1}{MN} \sum_{m'} \sum_{n'} p[m',n'] f[m,n;m',n']. \quad (2.25)$$

□

The response f of the system to an input impulse is its *point spread function* (PSF). The four-dimensional structure of the array f in the superposition operation for linear systems accommodates a system whose PSF varies according to a shift of the input impulse (i.e., a shift-variant PSF).

If the system is linear *and* shift-invariant, the structure of the array f that characterizes the system PSF is inherently two-dimensional (because the output image is the same except for shift regardless of the position of the input impulse). The operation of a linear, shift-invariant (LSI) system is *convolution*. Theorem 2.2 gives the mathematical form of the convolution operation.

Theorem 2.2 (Convolution Sum) *Each output value of a digital linear, shift-invariant system can be expressed as the sum of weighted values of the input where the weights are shift-invariant.*

That is, for every digital LSI imaging system $r = \mathcal{L}_f\{p\}$, there exists a unique two-dimensional array f that weights the elements of the input p to produce the values of the output r in the following way:

$$r[m, n] = \frac{1}{MN} \sum_{m'} \sum_{n'} p[m', n'] f[m - m', n - n']. \quad (2.26)$$

Proof: The proof of the convolution sum for LSI systems follows the proof of the superposition principle of linear systems through Equation 2.24. At that point, the shift-invariance of the system is used to simplify further:

$$\begin{aligned} r &= \frac{1}{MN} \sum_{m'} \sum_{n'} a_{m', n'} \mathcal{L}_f\{\delta_{m', n'}\} \\ &= \frac{1}{MN} \sum_{m'} \sum_{n'} a_{m', n'} \mathcal{S}_{m', n'}\{\mathcal{L}_f\{\delta\}\}. \end{aligned} \quad (2.27)$$

Defining the two-dimensional array f as the shift-invariant system response to an input impulse gives the desired form:

$$r[m, n] = \frac{1}{MN} \sum_{m'} \sum_{n'} p[m', n'] f[m - m', n - n']. \quad (2.28)$$

□

Because of the important role of LSI systems in image processing, this theorem and the method of its proof merit elaboration. As mentioned in the proof of Theorem 2.1, the decomposition of an image into a collection of scaled, shifted impulses is called impulse decomposition. The response of an LSI system to a single impulse, or point input, is its point spread function (PSF) which is shift-invariant. The output of a LSI system is the sum of its responses to the component impulses of the input image. The PSF of a LSI system is a complete characterization of the system. This method of defining the operation of an LSI system by characterizing its PSF is called *spatial analysis*. An alternative approach, frequency analysis, is presented in Section 2.7.

Convolution (as defined by Equation 2.26) can be written as a binary operation:

$$r = p * f. \quad (2.29)$$

(The '*' denotes convolution, not multiplication.) Even though the PSF f is used to characterize the system, it has the same mathematical structure as the image p . (The PSF is the image formed by the system for an impulse

input.) Convolution is a linear operation and is therefore distributive with respect to addition:

$$(p_1 + p_2) * f = p_1 * f + p_2 * f. \quad (2.30)$$

It is also commutative:

$$p * f = f * p \quad (2.31)$$

and associative:

$$(p * f_1) * f_2 = p * (f_1 * f_2). \quad (2.32)$$

The impulse is the identity with respect to convolution:

$$p = p * \delta. \quad (2.33)$$

The convolution of an image with itself is its *autoconvolution*.

Convolution is sometimes called *serial multiplication*. To illustrate convolution and to revisit the problem of assigning values outside the image borders during the shift operation, consider the one-dimensional, 8-element array

$$p = [1 \ 2 \ 4 \ 0 \ 3 \ 3 \ 3 \ 1] \quad (2.34)$$

and the LSI system impulse response or PSF f , where $f[-1] = 2$, $f[0] = 4$, and $f[1] = 2$.

First, assume that the pixel values outside the input image borders are zero:

$$\begin{array}{cccccccccccc}
 0 & 0 & 1 & 2 & 4 & 0 & 3 & 3 & 3 & 1 & 0 & 0 \\
 & & & & & & & & * & 2 & 4 & 2 \\
 \hline
 & & & 2 & 4 & 8 & 0 & 6 & 6 & 6 & 2 & \\
 & & 4 & 8 & 16 & 0 & 12 & 12 & 12 & 4 & & \\
 & 2 & 4 & 8 & 0 & 6 & 6 & 6 & 2 & & & \\
 \hline
 0 & 2 & 8 & 18 & 20 & 14 & 18 & 24 & 20 & 10 & 2 & 0.
 \end{array}$$

Normalizing by the dimension $N = 8$ yields the output

$$r = [0.25 \ 1.00 \ 2.25 \ 2.50 \ 1.75 \ 2.25 \ 3.00 \ 2.50 \ 1.25 \ 0.25] \quad (2.35)$$

from the zero-padded input

$$p = [0 \ 1 \ 2 \ 4 \ 0 \ 3 \ 3 \ 3 \ 1 \ 0]. \quad (2.36)$$

Note that the output has more non-zero elements than the input. It is problematic to have an image change size as it passes through a system (e.g., for the design of hardware to be used in a reconfigurable cascade or pipeline system). Of course, one can drop the elements on the ends to maintain a constant size, but this destroys some of the properties of the operation (e.g., associativity).

In contrast, convolution with the assumption of periodicity, called *circular convolution*, yields a slightly different result:

$$\begin{array}{cccccccccc}
 \dots & 1 & 2 & 4 & 0 & 3 & 3 & 3 & 1 & \dots \\
 & & & & & & * & 2 & 4 & 2 \\
 \hline
 & 2 & 2 & 4 & 8 & 0 & 6 & 6 & 6 & \\
 & 4 & 8 & 16 & 0 & 12 & 12 & 12 & 4 & \\
 & 4 & 8 & 0 & 6 & 6 & 6 & 2 & 2 & \\
 \hline
 \dots & 10 & 18 & 20 & 14 & 18 & 24 & 20 & 12 & \dots
 \end{array}$$

Normalizing by the dimension $N = 8$ yields the output

$$r = \left[\dots \ 1.25 \ 2.25 \ 2.50 \ 1.75 \ 2.25 \ 3.00 \ 2.50 \ 1.50 \ \dots \right] \quad (2.37)$$

from the periodic input

$$p = \left[\dots \ 1 \ 2 \ 4 \ 0 \ 3 \ 3 \ 3 \ 1 \ \dots \right]. \quad (2.38)$$

This result is periodic with the same period as the input image. Note that the middle six elements of the results in Equations 2.35 and 2.37 are the same; the differences at the ends are called *border effects*. Images are usually quite large and, provided the PSF is small, border effects are usually limited to a small band around the border. In that sense, border effects are usually insignificant. Unless stated otherwise, periodicity and circular convolution are used in subsequent chapters. The discrete Fourier transform (described in Section 2.8) requires periodicity. Its utility provides a decisive incentive for assuming periodically extended images.

Deterministic cross-correlation is related to convolution. The deterministic cross-correlation of two images is their convolution but with one of the images index-reversed and conjugated. (Conjugation is indicated by the asterisk exponent and described in Section 2.6. Real-valued functions are unchanged by conjugation.) Deterministic cross-correlation is written as

$$r = p_1 \star p_2 \quad (2.39)$$

where

$$r[m, n] = \frac{1}{MN} \sum_{m'} \sum_{n'} p_1[m', n'] p_2^*[m' - m, n' - n]. \quad (2.40)$$

In general, deterministic cross-correlation is not commutative. *Deterministic autocorrelation* is the deterministic cross-correlation of an image with itself.

2.5 Matrix Representation of Linear Systems

The one-dimensional, linear system $r = \mathcal{L}_f\{p\}$ where

$$r[n] = \sum_{n'} f[n, n'] p[n'] \quad (2.41)$$

can be equivalently expressed as a matrix equation:

$$\mathbf{r} = \mathbf{F}\mathbf{p} \quad (2.42)$$

where \mathbf{r} is the $N \times 1$ output column vector, \mathbf{p} is the $N \times 1$ input column vector, and \mathbf{F} is the $N \times N$ square coefficient matrix.

If the system is both linear and shift-invariant, the $N \times N$ coefficient matrix \mathbf{F} is a *Toeplitz* matrix—that is, the entries are constant along the diagonals:

$$\mathbf{F}[n_1, n_2] = f[n_1 - n_2]. \quad (2.43)$$

The Toeplitz matrix formed from the impulse is the identity matrix:

$$\mathbf{I}[n_1, n_2] = \delta[n_1 - n_2]. \quad (2.44)$$

If periodicity is assumed, then the coefficient matrix is a *circulant* matrix—that is, each successive row is the same as the previous row circularly-shifted one element:

$$\mathbf{F}[n_1, n_2] = f[(n_1 - n_2) \bmod N]. \quad (2.45)$$

A circulant matrix is a Toeplitz matrix, but not necessarily conversely.

The matrix representation of two-dimensional systems requires arrays of higher dimensionality— $M \times N$ for the input and output images and $M \times N \times M \times N$ for the coefficient matrices. It is possible, however, to express the two-dimensional images as one-dimensional vectors and the four-dimensional arrays as two-dimensional matrices using *stacking* (or *lexicographic ordering*)[5]. For a two-dimensional, linear operator where

$$r[m, n] = \sum_{m'} \sum_{n'} p[m', n'] f[m, n; m', n'], \quad (2.46)$$

the input and output matrices can be stacked in raster-scan order (row-major, that is across the columns of the 0th row, then the columns of the 1st row, etc.) to form block matrices. The $M \times N$ matrices are reduced to $MN \times 1$ block matrices of the form

$$\mathbf{p} = \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_{M-1} \end{bmatrix} \quad (2.47)$$

where each block \mathbf{p}_i is a column vector containing row i of the image:

$$\mathbf{p}_i = \begin{bmatrix} p[i, 0] \\ p[i, 1] \\ \vdots \\ p[i, N-1] \end{bmatrix}. \quad (2.48)$$

The $M \times N \times M \times N$ coefficient matrix is reduced to a $MN \times MN$ block matrix of the form

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_{0,0} & \mathbf{F}_{0,1} & \cdots & \mathbf{F}_{0,M-1} \\ \mathbf{F}_{1,0} & \mathbf{F}_{1,1} & \cdots & \mathbf{F}_{1,M-1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{F}_{M-1,0} & \mathbf{F}_{M-1,1} & \cdots & \mathbf{F}_{M-1,M-1} \end{bmatrix} \quad (2.49)$$

where each block is

$$\mathbf{F}_{i,j} = \begin{bmatrix} f[i, 0, j, 0] & f[i, 0, j, 1] & \cdots & f[i, 0, j, N-1] \\ f[i, 1, j, 0] & f[i, 1, j, 1] & \cdots & f[i, 1, j, N-1] \\ \vdots & \vdots & \ddots & \vdots \\ f[i, N-1, j, 0] & f[i, N-1, j, 1] & \cdots & f[i, N-1, j, N-1] \end{bmatrix}. \quad (2.50)$$

Stacking is merely a representational convention and is of no practical significance.

2.6 Complex Arithmetic

Image transforms that employ complex numbers often simplify the mathematical analysis of real systems. A *complex number* has two components that can be expressed as an ordered pair of real numbers $z = (x, y)$. The x value is the real part of the complex number; the y value is called the imaginary part. The ordered pair can be interpreted as the Cartesian coordinates of a point in a complex plane with real abscissa and imaginary ordinate.

The sum of two complex numbers is defined as

$$\begin{aligned} z_1 + z_2 &= (x_1, y_1) + (x_2, y_2) \\ &= (x_1 + x_2, y_1 + y_2). \end{aligned} \quad (2.51)$$

The complex product is

$$\begin{aligned} z_1 z_2 &= (x_1, y_1)(x_2, y_2) \\ &= (x_1 x_2 - y_1 y_2, y_1 x_2 + x_1 y_2). \end{aligned} \quad (2.52)$$

Complex addition and multiplication are commutative and associative. Complex multiplication is distributive with respect to complex addition. The identity element for complex addition is zero, written 0 or $(0, 0)$. The identity element for complex multiplication is the real number one, written 1 or $(1, 0)$. The square of the unit imaginary number, $i = (0, 1)$, is the real number -1 or $(-1, 0)$:

$$\begin{aligned} i^2 &= (0, 1)(0, 1) \\ &= (-1, 0). \end{aligned} \quad (2.53)$$

As an alternative to the ordered pair notation, complex numbers are conventionally written as a sum:

$$\begin{aligned} z &= (z_r, z_i) \\ &= (z_r, 0) + (0, 1)(z_i, 0) \\ &= z_r + iz_i. \end{aligned} \quad (2.54)$$

The *conjugate* of a complex number has the same real part and the negative of the imaginary part:

$$z^* = z_r - iz_i. \quad (2.55)$$

The *absolute value* of a complex number is the square-root of the product of the number and its conjugate:

$$\begin{aligned} |z| &= \sqrt{zz^*} \\ &= \sqrt{z_r^2 + z_i^2}. \end{aligned} \quad (2.56)$$

The additive inverse of a complex number $z = z_r + iz_i$ is

$$-z = -z_r - iz_i \quad (2.57)$$

and the multiplicative inverse (or reciprocal) is

$$\begin{aligned} z^{-1} &= z^* |z|^{-2} \\ &= \frac{z_r - iz_i}{z_r^2 + z_i^2} \\ &= \frac{z_r}{z_r^2 + z_i^2} - i \frac{z_i}{z_r^2 + z_i^2}. \end{aligned} \quad (2.58)$$

The multiplicative inverse of zero is undefined.

Another representation for complex numbers is the *polar* or *magnitude-phase* form:

$$z = z_\rho \cos z_\phi + iz_\rho \sin z_\phi. \quad (2.59)$$

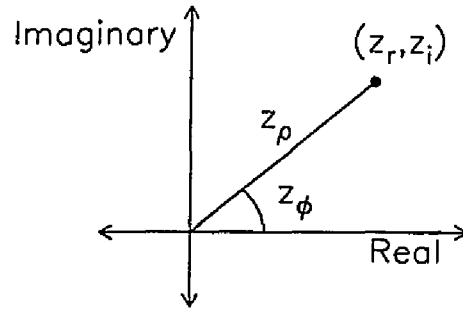


Figure 2.3: Complex Numbers

As Figure 2.3 illustrates, the magnitude-phase notation employs polar coordinates to locate points in the complex plane. The *magnitude* is the absolute value:

$$z_\rho = |z|. \quad (2.60)$$

The *phase* can be determined from the real and imaginary parts by the equation

$$z_\phi = \arctan\left(\frac{z_i}{z_r}\right) \quad (z \neq 0) \quad (2.61)$$

and the quadrant of the point.

Euler's formula:

$$\exp(i\theta) = \cos\theta + i\sin\theta \quad (2.62)$$

justifies writing the magnitude-phase representation as a complex exponential:

$$z = z_\rho \exp(iz_\phi). \quad (2.63)$$

Complex exponentials provide a valuable notation for simplifying the mathematics of spatial frequency analysis (Section 2.7).

2.7 Spatial Frequency Analysis

If the output of a discrete linear system is a scaled version of the input, the input is an *eigenvector* of the system and the scalar factor is the corresponding *eigenvalue* of the system. That is, if there exists a real or complex scalar a such that

$$\mathcal{L}\{p\} = ap, \quad (2.64)$$

then the image p is an eigenvector and a is the corresponding eigenvalue.

Spatial analysis, described in Section 2.4, is accomplished by decomposing an $M \times N$ image into MN impulse components and evaluating the system's response to each impulse (as in Theorems 2.1 and 2.2). *Spatial frequency analysis* proceeds in a similar fashion, by decomposing an image into MN components that are eigenvectors of the system and evaluating the system's response to each eigenvector. The eigenvectors of linear, shift-invariant systems are given in the following theorem.

Theorem 2.3 *The eigenvectors of linear, shift-invariant systems are the complex exponentials $\exp\left(i2\pi\left(\frac{\mu m}{M} + \frac{\nu n}{N}\right)\right)$.*

Each complex exponential consists of a real cosine wave and an imaginary sine wave (Equation 2.62):

$$\exp\left(i2\pi\left(\frac{\mu m}{M} + \frac{\nu n}{N}\right)\right) = \cos\left(2\pi\left(\frac{\mu m}{M} + \frac{\nu n}{N}\right)\right) + i \sin\left(2\pi\left(\frac{\mu m}{M} + \frac{\nu n}{N}\right)\right). \quad (2.65)$$

The values of μ and ν determine the *spatial frequency* of the cosine and sine waves—the waves have μ periods per M pixels vertically and ν periods per N pixels horizontally. For a $M \times N$ system, there are MN unique eigenvectors and MN corresponding complex eigenvalues. (The set for $0 \leq \mu < M$ and $0 \leq \nu < N$ is complete; the modulus maps the eigenvectors for any other integer frequencies to this set.)

Proof: The proof demonstrates that given an arbitrary complex exponential as input, the output of an LSI system is the complex exponential scaled by the eigenvalue. Let the input be a complex exponential $\omega_{\mu,\nu}$ where

$$\omega_{\mu,\nu}[m, n] = \exp\left(i2\pi\left(\frac{\mu m}{M} + \frac{\nu n}{N}\right)\right) \quad (2.66)$$

for all $[m, n]$. The LSI system response is

$$r = f * \omega_{\mu,\nu}. \quad (2.67)$$

Then for any pixel $[m, n]$,

$$\begin{aligned} r[m, n] &= \frac{1}{MN} \sum_{m'} \sum_{n'} f[m', n'] \exp\left(i2\pi\left(\frac{\mu(m-m')}{M} + \frac{\nu(n-n')}{N}\right)\right) \\ &= \exp\left(i2\pi\left(\frac{\mu m}{M} + \frac{\nu n}{N}\right)\right) \frac{1}{MN} \sum_{m'} \sum_{n'} f[m', n'] \exp\left(-i2\pi\left(\frac{\mu m'}{M} + \frac{\nu n'}{N}\right)\right) \\ &= \exp\left(i2\pi\left(\frac{\mu m}{M} + \frac{\nu n}{N}\right)\right) \hat{f}[\mu, \nu] \end{aligned} \quad (2.68)$$

where

$$\hat{f}[\mu, \nu] = \frac{1}{MN} \sum_{m'} \sum_{n'} f[m', n'] \exp\left(-i2\pi\left(\frac{\mu m'}{M} + \frac{\nu n'}{N}\right)\right). \quad (2.69)$$

So, the output is a scaled version of the input:

$$r = \hat{f}[\mu, \nu] \omega_{\mu, \nu} \quad (2.70)$$

where the eigenvalue $\hat{f}[\mu, \nu]$ defined in Equation 2.69 depends on the spatial frequency $[\mu, \nu]$ of the input eigenvector $\omega_{\mu, \nu}$.

□

A complex exponential $\omega_{\mu, \nu}$ passes through a LSI system unchanged except that the *modulation* of the waves (the height from the peak of the wave to its trough) is scaled by the magnitude of the eigenvalue $\hat{f}_\rho[\mu, \nu]$ and the waves are shifted by the phase of the eigenvalue $\hat{f}_\phi[\mu, \nu]$. The eigenvalue magnitude as a function of frequency is the *modulation transfer function* (MTF) of the system. The *transfer function* or *optical transfer function* (OTF) specifies both the magnitude and phase of the eigenvalues.

An abbreviated notation for the complex exponentials is useful:

$$W_N = \exp\left(\frac{i2\pi}{N}\right). \quad (2.71)$$

The spatial frequency ν and the spatial position n are then specified in the exponent:

$$W_N^{\nu n} = \exp\left(i2\pi \frac{\nu n}{N}\right). \quad (2.72)$$

The complex exponentials are *separable*—the two-dimensional function can be written as the product of separate functions in each of the dimensions:

$$\begin{aligned} \exp\left(i2\pi\left(\frac{\mu m}{M} + \frac{\nu n}{N}\right)\right) &= \exp\left(i2\pi \frac{\mu m}{M}\right) \exp\left(i2\pi \frac{\nu n}{N}\right) \\ &= W_M^{\mu m} W_N^{\nu n}. \end{aligned} \quad (2.73)$$

The decomposition of an image into complex exponentials is less obvious than impulse decomposition (described in Section 2.4), but just as the set of shifted impulses are the basis of spatial analysis so the complex exponentials are the basis of spatial frequency analysis. The first step in illustrating spatial frequency decomposition is to show that the MN complex exponentials are *orthogonal*—that the inner product (or scalar product) of any two different complex exponentials is zero. The use of complex exponentials as the basis for decomposition follows directly from this result.

Theorem 2.4 (Orthogonality Principle) *The set of MN unique complex exponentials is orthogonal.*

Proof: The inner product of two arbitrary complex exponentials is

$$\begin{aligned} \sum_m \sum_n (W_M^{\mu m} W_N^{\nu n}) (W_M^{-\mu' m} W_N^{-\nu' n}) &= \sum_m \sum_n W_M^{(\mu-\mu')m} W_N^{(\nu-\nu')n} \\ &= \left(\sum_m W_M^{(\mu-\mu')m} \right) \left(\sum_n W_N^{(\nu-\nu')n} \right). \end{aligned} \quad (2.74)$$

(Note that the pointwise product of two complex exponentials is a complex exponential whose frequency is the sum of the frequencies of the factors.) Each separable factor is a geometric series. The sum of a geometric series

$$S_N = \sum_{n=0}^{N-1} r^n \quad (2.75)$$

is

$$S_N = \begin{cases} N & \text{if } r = 1 \\ \frac{1-r^N}{1-r} & \text{if } r \neq 1. \end{cases} \quad (2.76)$$

For this series,

$$\begin{aligned} r &= W_N^{(\nu-\nu')} \\ &= \begin{cases} 1 & \text{if } (\nu - \nu') \bmod N = 0 \\ \exp\left(i2\pi \frac{(\nu-\nu')}{N}\right) & \text{otherwise} \end{cases} \end{aligned} \quad (2.77)$$

and

$$\begin{aligned} r^N &= W_N^{(\nu-\nu')N} \\ &= \exp\left(i2\pi \frac{(\nu-\nu')N}{N}\right) \\ &= 1. \end{aligned} \quad (2.78)$$

Therefore,

$$\sum_n W_N^{(\nu-\nu')n} = \begin{cases} N & \text{if } (\nu - \nu') \bmod N = 0 \\ 0 & \text{otherwise} \end{cases} \quad (2.79)$$

and in two-dimensions,

$$\begin{aligned} \sum_m \sum_n (W_M^{\mu m} W_N^{\nu n}) (W_M^{-\mu' m} W_N^{-\nu' n}) \\ = \begin{cases} MN & \text{if } [(\mu - \mu') \bmod M, (\nu - \nu') \bmod N] = [0, 0] \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \quad (2.80)$$

□

The spatial frequency decomposition of an image can now be demonstrated:

$$\begin{aligned} p[m, n] &= \sum_{m'} \sum_{n'} p[m', n'] \frac{1}{MN} \sum_{\mu} \sum_{\nu} W_M^{\mu(m-m')} W_N^{\nu(n-n')} \\ &= \sum_{\mu} \sum_{\nu} \left(\frac{1}{MN} \sum_{m'} \sum_{n'} p[m', n'] W_M^{-\mu m'} W_N^{-\nu n'} \right) W_M^{\mu m} W_N^{\nu n} \end{aligned} \quad (2.81)$$

and so

$$p[m, n] = \sum_{\mu} \sum_{\nu} \hat{p}[\mu, \nu] W_M^{\mu m} W_N^{\nu n} \quad (2.82)$$

where each frequency component is given by

$$\hat{p}[\mu, \nu] = \frac{1}{MN} \sum_m \sum_n p[m, n] W_M^{-\mu m} W_N^{-\nu n}. \quad (2.83)$$

Equation 2.82 is often referred to as the *synthesis equation*—the image is the synthesis of the frequency components. The image specified in Equation 2.82 is periodic. Equation 2.83 is the *analysis equation*—the frequency components are distilled from the image. The array $\hat{p}[\mu, \nu]$ defined by Equation 2.83 is called the *discrete Fourier spectrum* of the image. The Fourier spectrum of a periodic image is discrete and the Fourier spectrum of a discrete image is periodic. The analysis equation is also used to compute the OTF from the PSF. (Equation 2.83 is identical to Equation 2.69.) The synthesis equation is also used to compute the PSF from the OTF. As was noted in Section 2.4, images and PSFs have the same mathematical structure—the difference between them is semantic. Image spectra and OTFs also have the same mathematical structure.

It is instructive to look at an example of frequency decomposition. Consider the one-dimensional, periodic, discrete sawtooth pictured on the left in Figure 2.4 ($N = 64$). The imaginary part of the periodic Fourier spectrum is pictured on the right of Figure 2.4. (The real part is zero.) As is typical with most images, the magnitude of the Fourier spectrum is greater at lower frequencies. This reflects the fact that pixels are highly correlated with nearby pixels and less correlated with distant pixels. Figure 2.5 shows how the scaled sine waves sum to the sawtooth. Only the first few frequencies are shown. The first column illustrates the sine wave scaled by its spectral magnitude. The second column is the accumulated sum. Here also, note that the magnitude of the sine wave components is greater at low frequencies than high and that with just a few terms, the shape of the sawtooth is evident.

A LSI system is completely characterized by its responses to the complex exponentials (i.e., the transfer function as in Equation 2.69). The input

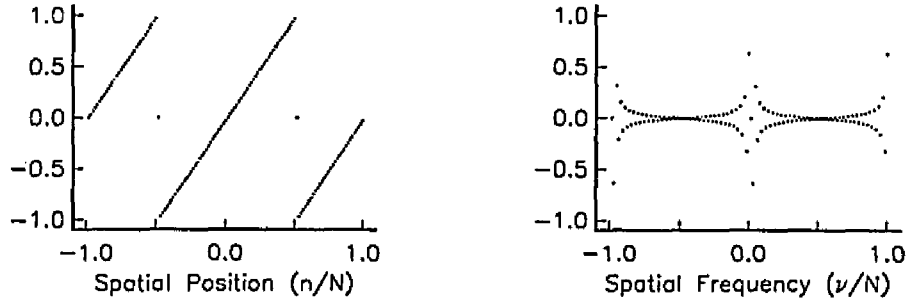


Figure 2.4: Periodic Sawtooth and Its Periodic Spectrum

image is decomposed into a linear combination of scaled complex exponentials (i.e., the Fourier spectrum as in Equation 2.83). The output of a linear system is the sum of its responses to the image components. In this case, the components are eigenvectors, so the output image components can be computed by pointwise multiplication.

Theorem 2.5 (Convolution Theorem) *The Fourier spectrum of the output of a linear, shift-invariant system is the pointwise product of the Fourier spectrum of the input and the system transfer function.*

Proof: Given a LSI system $r = p * f$, where p is the input image, f is the system PSF, and r is the output image, the Fourier spectrum of the output image \hat{r} is the product of the input image spectrum \hat{p} and the system transfer function \hat{f} :

$$\hat{r} = \hat{p}\hat{f}. \quad (2.84)$$

The output image can be rewritten as

$$\begin{aligned} r[m, n] &= \frac{1}{MN} \sum_{m'} \sum_{n'} p[m', n'] f[m - m', n - n'] \\ &= \frac{1}{MN} \sum_{m'} \sum_{n'} \left(\sum_{\mu} \sum_{\nu} \hat{p}[\mu, \nu] W_M^{\mu m'} W_N^{\nu n'} \right) \\ &\quad \times \left(\sum_{\mu'} \sum_{\nu'} \hat{f}[\mu', \nu'] W_M^{\mu'(m-m')} W_N^{\nu'(n-n')} \right) \\ &= \sum_{\mu} \sum_{\nu} \sum_{\mu'} \sum_{\nu'} \hat{p}[\mu, \nu] \hat{f}[\mu', \nu'] W_M^{\nu' m} W_N^{\nu' n} \\ &\quad \times \frac{1}{MN} \sum_{m'} \sum_{n'} W_M^{(\mu-\mu')m'} W_N^{(\nu-\nu')n'} \\ &= \sum_{\mu} \sum_{\nu} \hat{p}[\mu, \nu] \hat{f}[\mu, \nu] W_M^{\nu m} W_N^{\nu n}. \end{aligned} \quad (2.85)$$

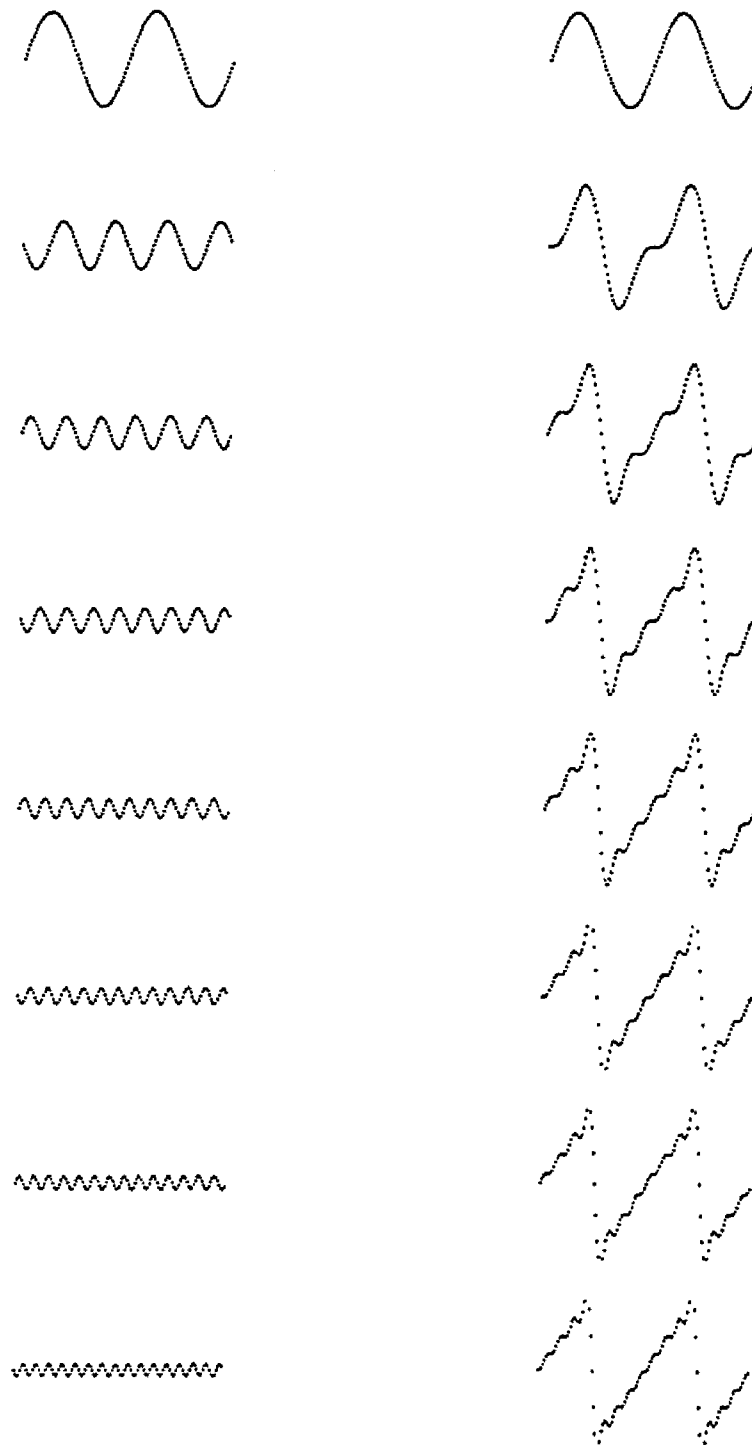


Figure 2.5: Sawtooth Spatial-Frequency Decomposition

The Fourier spectrum of the output image is

$$\begin{aligned}
 \hat{r}[\mu, \nu] &= \frac{1}{MN} \sum_m \sum_n r[m, n] W_M^{-\mu m} W_N^{-\nu n} \\
 &= \frac{1}{MN} \sum_m \sum_n \sum_{\mu'} \sum_{\nu'} \hat{p}[\mu', \nu'] \hat{f}[\mu', \nu'] W_M^{\mu' m} W_N^{\nu' n} W_M^{-\mu m} W_N^{-\nu n} \\
 &= \sum_{\mu'} \sum_{\nu'} \hat{p}[\mu', \nu'] \hat{f}[\mu', \nu'] \frac{1}{MN} \sum_m \sum_n W_M^{(\mu' - \mu)m} W_N^{(\nu' - \nu)n} \\
 &= \hat{p}[\mu, \nu] \hat{f}[\mu, \nu].
 \end{aligned} \tag{2.86}$$

□

The operation of a LSI system on the spatial frequency components of an image is pointwise multiplication. Spatial analysis leads to a definition of the system in terms of convolution (Equation 2.26). Pointwise multiplication is a more familiar operation than convolution and it may require less computation. Convolution of a $M \times N$ image with a $M \times N$ PSF is $O(M^2 N^2)$. Pointwise multiplication is $O(MN)$. Typically, images are acquired in pixel form (i.e., as spatial components rather than spectral frequency components) and output images typically are expected in the same form, so there is usually some overhead in computing the image spectrum and synthesizing the output image from its spectrum. The relative computational costs of spatial convolution and frequency multiplication are examined in detail in Chapter 5.

Spatial frequency analysis is also called *Fourier analysis*. The calculation of the discrete Fourier spectrum of an image (Equation 2.83) is called the discrete Fourier transform. The calculation of the image from the Fourier spectrum (Equation 2.82) is the inverse transform. The relationship between images and their Fourier spectra is examined more fully in the next section.

2.8 Discrete Fourier Transform

The *discrete Fourier transform* (DFT) is

$$\hat{p} = \mathcal{F}\{p\} \tag{2.87}$$

where

$$\hat{p}[\mu, \nu] = \frac{1}{MN} \sum_m \sum_n p[m, n] W_M^{-\mu m} W_N^{-\nu n}. \tag{2.88}$$

Theorem	p	\hat{p}
Linearity	$ap_1 + bp_2$	$a\hat{p}_1 + b\hat{p}_2$
Convolution	$p_1 * p_2$	$\hat{p}_1 \hat{p}_2$
Autocorrelation	$p_1 * p_1$	$ \hat{p}_1 ^2$
Shift	$p[m - a, n - b]$	$\hat{p}[\mu, \nu] W_M^{a\mu} W_N^{b\nu}$
Parseval's	$\frac{1}{MN} \sum_m \sum_n p[m, n] ^2$	$\sum_\mu \sum_\nu \hat{p}[\mu, \nu] ^2$
Separable	$p_1[m] p_2[n]$	$\hat{p}_1[\mu] \hat{p}_2[\nu]$

Table 2.1: Fourier Transform Theorems

The inverse discrete Fourier transform is

$$p = \mathcal{F}^{-1}\{\hat{p}\} \quad (2.89)$$

where

$$p[m, n] = \sum_\mu \sum_\nu \hat{p}[\mu, \nu] W_M^{\mu m} W_N^{\nu n}. \quad (2.90)$$

The functions p and \hat{p} are a transform pair— p in the spatial domain and \hat{p} in the spatial-frequency domain. If the spatial function is an image, the frequency function is the Fourier spectrum. If the spatial function is a PSF, the frequency function is the OTF. By convention, the horizontal and vertical unit spatial distances are the sampling intervals. (See Figure 2.1.) This simplifies spatial equations by making spatial indices correspond to spatial location (i.e., the location indexed by $[m, n]$ is (m, n)). The frequency indices μ and ν are relative to the image size— $[\mu, \nu]$ indicates μ periods per M pixels vertically and ν periods per N pixels horizontally—so the frequency indices $[\mu, \nu]$ correspond to the spatial frequency $(\mu/M, \nu/N)$.

Table 2.1 lists several theorems for the discrete Fourier transform. The Fourier transform is linear. The convolution theorem is proven in Section 2.7. The transform of the autocorrelation function is called the *power spectrum*. Parseval's theorem equates the mean-square in the spatial domain and the sum-of-the-squares in the frequency domain. The transform of a separable function is the separable product of the separate transforms.

The symmetry of a function and the symmetry of its transform are related. A function can be separated into an even part (symmetric about the

origin) and an odd part (antisymmetric):

$$p[m, n] = p_e[m, n] + p_o[m, n] \quad (2.91)$$

where

$$\begin{aligned} p_e[m, n] &= p_e[-m, -n] \\ &= \frac{1}{2}(p[m, n] + p[-m, -n]) \end{aligned} \quad (2.92)$$

$$\begin{aligned} p_o[m, n] &= -p_o[-m, -n] \\ &= \frac{1}{2}(p[m, n] - p[-m, -n]). \end{aligned} \quad (2.93)$$

If the function is complex, both its real and imaginary parts can be separated into even and odd parts. These distinct components of a function result in distinct components of the transform. Table 2.2 gives the relationships between the even and odd components of a function and its transform. The Fourier transform is linear, so the combination of any or all of the functional components yields the corresponding combination of transform components. For example, the transform of a real function (i.e., consisting of even real and odd real parts) is *Hermitian*—having an even real part and an odd imaginary part.

p	\hat{p}
Real & Even	Real & Even
Real & Odd	Imaginary & Odd
Imaginary & Even	Imaginary & Even
Imaginary & Odd	Real & Odd

Table 2.2: Fourier Transform Relationships

2.9 Continuous Fourier Transform

Depending on one's point of view, the DFT is a special case of the Fourier transform or the Fourier transform is the limiting case of the DFT. The discrete Fourier transforms, defined in Equations 2.88 and 2.90, can be written

$$\hat{p}[\mu, \nu] = \frac{1}{MN} \sum_{m=-M/2}^{(M/2)-1} \sum_{n=-N/2}^{(N/2)-1} p[m, n] W_M^{-\mu m} W_N^{-\nu n} \quad (2.94)$$

$$p[m, n] = \sum_{\mu=-M/2}^{(M/2)-1} \sum_{\nu=-N/2}^{(N/2)-1} \hat{p}[\mu, \nu] W_M^{\mu m} W_N^{\nu n} \quad (2.95)$$

p	\hat{p}
Impulse	1
1	Impulse
Pulse	Sinc
Sinc	Pulse
Comb	Comb
Bell	Bell

Table 2.3: Fourier Transform Pairs

where the frequency component indexed as $\hat{p}[\mu, \nu]$ is actually the frequency component $\hat{p}(\mu/M, \nu/N)$. In the DFT equations, both the image and the transform are discrete and periodic.

A discrete function has a periodic transform and a periodic function has a discrete transform. A continuous, periodic image has unlimited resolution and therefore there is no limit on the component frequencies:

$$\hat{p}[\mu, \nu] = \frac{1}{MN} \int_{-M/2}^{M/2} \int_{-N/2}^{N/2} p(x, y) W_M^{-\mu x} W_N^{-\nu y} dx dy \quad (2.96)$$

$$p(x, y) = \sum_{\mu=-\infty}^{\infty} \sum_{\nu=-\infty}^{\infty} \hat{p}[\mu, \nu] W_M^{\mu x} W_N^{\nu y} \quad (2.97)$$

Here, the discrete transform \hat{p} is called the *complex Fourier series* of the periodic image p . Similarly, a discrete, aperiodic image (called a *time series* in the signal processing literature) has a continuous, periodic transform:

$$\hat{p}(u, v) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} p[m, n] W^{-um} W^{-vn} \quad (2.98)$$

$$p[m, n] = \int_{-1/2}^{1/2} \int_{-1/2}^{1/2} \hat{p}(u, v) W^{um} W^{vn} du dv \quad (2.99)$$

If the image is continuous *and* aperiodic, the transform is also continuous and aperiodic. This yields the forward and inverse Fourier transforms:

$$\hat{p}(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} p(x, y) W^{-ux} W^{-vy} dx dy \quad (2.100)$$

$$p(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \hat{p}(u, v) W^{ux} W^{vy} du dv \quad (2.101)$$

Several useful functions are illustrated in Appendix A. Table 2.3 lists the transform pairs of these functions. The bell and comb functions are their own

transforms. The sinc and pulse functions are a transform pair. The impulse (identity under convolution) and 1 (identity under pointwise multiplication) are a transform pair.

2.10 Inverse Problems

The problem considered in Sections 2.4 and 2.7 is to determine the output image r of the system

$$r = \mathcal{O}_f\{p\} \quad (2.102)$$

from the input image p and knowledge of the system. The *inverse problem* is to determine the input image p from the output image r and knowledge of the system.

Stable or well-conditioned systems can be accurately inverted on a digital computer. If small inaccuracies due to digital calculations or other sources of noise can result in large errors in the inversion, then the system is *ill-conditioned* or *unstable*. *Singular systems* have no (unique) inverse. Restoration is an inverse operation, so ill-conditioned and singular systems must be considered in restoration filter design. In fact, in real imaging systems, noise is virtually never insignificant.

The system of Equation 2.102 is invertible if and only if there exists an operator \mathcal{O}' such that for any image p ,

$$\begin{aligned} \mathcal{O}'\{r\} &= \mathcal{O}'\{\mathcal{O}\{p\}\} \\ &= p. \end{aligned} \quad (2.103)$$

A linear system $\mathbf{r} = \mathbf{F}\mathbf{p}$ is invertible if and only if its coefficient matrix \mathbf{F} can be inverted—i.e., there is a matrix \mathbf{F}^{-1} such that $\mathbf{F}\mathbf{F}^{-1} = \mathbf{F}^{-1}\mathbf{F} = \mathbf{I}$. A LSI system is invertible if and only if the PSF has a convolutional inverse—that is, for a system with PSF f , there exists f' such that

$$f * f' = \delta. \quad (2.104)$$

Fourier analysis simplifies consideration of the invertibility of LSI systems.

Theorem 2.6 (Invertibility) *A linear, shift-invariant system has an inverse if and only if the system transfer function has no zero values.*

Proof: By the Convolution Theorem (Theorem 2.5), the output spectrum \hat{r} is the product of the image spectrum \hat{p} and the system OTF \hat{f} :

$$\hat{r} = \hat{p}\hat{f}. \quad (2.105)$$

If the inverse \hat{f}' exists, then at each frequency $[\mu, \nu]$,

$$\hat{p}[\mu, \nu] = \hat{p}[\mu, \nu] \hat{f}[\mu, \nu] \hat{f}'[\mu, \nu]. \quad (2.106)$$

This equation must be true for all images, even those with no zeros in the spectrum. Therefore,

$$1 = \hat{f}[\mu, \nu] \hat{f}'[\mu, \nu]. \quad (2.107)$$

So the OTF has no zeros— $\hat{f}[\mu, \nu] \neq 0$. The converse is also true; if the OTF has no zeros the LSI system has a LSI inverse whose transfer function \hat{f}' is defined at each frequency as

$$\hat{f}'[\mu, \nu] = \frac{1}{\hat{f}[\mu, \nu]}. \quad (2.108)$$

□

A *condition number* is a measure of the conditioning of a system. The common definition of the condition number of a matrix is the ratio of the magnitude of the eigenvalue with the largest magnitude (the spectral radius) to the magnitude of the eigenvalue with the smallest magnitude.² The condition number of a LSI system is calculated from its MTF:

$$\kappa_f = \frac{\max \langle |\hat{f}[\mu, \nu]| \rangle}{\min \langle |\hat{f}[\mu, \nu]| \rangle} \quad (2.109)$$

If the ratio is 1, the system is perfectly-conditioned. This condition exists if and only if the values of the MTF are all equal. Ill-conditioned systems have a large ratio. A singular system will have at least one zero eigenvalue, in which case the ratio is undefined.

2.11 Image Statistics

A typical digital image contains a large number of elements; for example, a 1024×1024 image has more than one million pixels. Our visual system can receive this volume of information readily (e.g., as it is displayed on a

²This is the definition of the condition number of a nonsingular, normal matrix with respect to the spectral norm. The spectral norm is:

$$\|F\|_2 \triangleq \max \langle \sqrt{\lambda}: \lambda \text{ is an eigenvalue of } F^*F \rangle.$$

Other matrix norms can also be used to define the condition number[6].

high-resolution monitor) and make sense of it. However, for many image processing operations, there is so much data in an image that it must be reduced to a more compact form appropriate for the operation. Consider the simple image processing operation of manually adjusting the brightness control on a television. When you ask another viewer if the brightness should be changed, you don't want to hear, "Pixel [0, 0] has brightness level 32, pixel [0, 1] has brightness level 35, . . ." You expect the image data to be reduced to one of three characterizations: the image is too bright, it is too dark, or it looks fine. *Descriptive statistics* characterize a relevant aspect of an image in a useful form. This section defines the most common descriptive image statistics for gray-scale images (i.e., each pixel has one of finitely many real, non-negative values).

The *arithmetic mean* or *average* brightness value of a digital image is the sum of the pixel values divided by the number of pixels:

$$\bar{p} = \frac{1}{MN} \sum_m \sum_n p[m, n]. \quad (2.110)$$

The image mean is a global measure of image brightness. The brightness control on a television set effectively raises or lowers \bar{p} .

Image *variance* is defined as

$$\begin{aligned} \sigma_p^2 &= \frac{1}{MN} \sum_m \sum_n |p[m, n] - \bar{p}|^2 \\ &= \frac{1}{MN} \sum_m \sum_n |p[m, n]|^2 - |\bar{p}|^2. \end{aligned} \quad (2.111)$$

The square root of the variance is the *standard deviation* σ . The standard deviation is a measure of image *contrast*; the contrast control on a television set increases or decreases the standard deviation. Following Park[7], image contrast is defined as the standard deviation.

The *energy* of a pixel is the square of its brightness, $|p[m, n]|^2$. The *total energy* of a digital image is the sum of the pixel energies. The *mean-square energy* is the total energy divided by the number of pixels:

$$\gamma_p^2 = \frac{1}{MN} \sum_m \sum_n |p[m, n]|^2. \quad (2.112)$$

The mean-square energy reflects both the average and contrast of an image:

$$\gamma_p^2 = \sigma_p^2 + |\bar{p}|^2. \quad (2.113)$$

The *root-mean-square* (RMS) energy γ is the square root of the mean-square energy.

The image *gray-level distribution* or *histogram* is the relative frequency function of pixel values:

$$g_p[l] = \frac{1}{MN} \text{count} \langle [m, n] : p[m, n] = l \rangle. \quad (2.114)$$

The gray-level distribution is conventionally normalized by the number of pixels so that the sum of the histogram values over all gray levels is 1:

$$1 = \sum_{l=0}^{L-1} g_p[l]. \quad (2.115)$$

The *cumulative gray-level distribution* or *cumulative histogram* is

$$\begin{aligned} G_p[l] &= \frac{1}{MN} \text{count} \langle [m, n] : p[m, n] \leq l \rangle \\ &= \sum_{l'=0}^l g_p[l']. \end{aligned} \quad (2.116)$$

If the number of gray levels L is small (e.g., 256), the gray-level distribution and the cumulative gray-level distribution of an image are useful characterizations of an image. The image mean, contrast, and RMS energy can all be calculated from the histogram. For example, the mean is

$$\bar{p} = \sum_{l=0}^{L-1} l g_p[l]. \quad (2.117)$$

It is sometimes convenient to work with functions that are *normalized* or *standardized* to have zero mean ($\bar{p} = 0$) and unit RMS energy ($\gamma_p = 1$). An image can be normalized by subtracting out the image mean and dividing by contrast:

$$p[m, n] \leftarrow \frac{p[m, n] - \bar{p}}{\sigma_p} \quad (2.118)$$

for all $[m, n]$.

Statistics can also be used to relate two images—for example, characterizing their difference or similarity. The inner product of two images p and q normalized by the image size is

$$\gamma_{p,q}^2 = \frac{1}{MN} \sum_m \sum_n p[m, n] q^*[m, n]. \quad (2.119)$$

The *covariance* of two images is

$$\sigma_{p,q}^2 = \gamma_{p,q}^2 - \bar{p}\bar{q}^*. \quad (2.120)$$

The covariance of an image with itself is its variance. The *correlation coefficient* is

$$\rho_{p,q} = \frac{\sigma_{p,q}^2}{\sigma_p \sigma_q}. \quad (2.121)$$

For normalized images, the correlation coefficient is the inner product divided by the image area. In cases where the relative brightness and the relative contrast are not pertinent, the inner product divided by the image area may be used as the (unnormalized) correlation.

A *second-order gray-level distribution* gives the relative frequency of the co-occurrence of brightness values in two images. For example, $g_{p,q}[l_1, l_2]$ is the relative frequency of pixels with l_1 value in image p and l_2 value in image q . Second-order gray-level distribution functions are usually too large to be practical—a gray-scale with only 256 gray levels may require a 65,536-element second-order distribution table.

2.12 Image Ensembles

In many situations, it is desirable to consider an *ensemble* (set, group, or family) of images rather than a single image. A *stochastic image* has a discrete random variable rather than a value associated with each pixel. An image formed with a value for each of the random variables is a sample of the stochastic process or member of the ensemble.

For a given pixel $[m, n]$, the cumulative distribution function (CDF) of the associated discrete random variable is

$$\text{cdf}_p[l; m, n] = \Pr\{p[m, n] \leq l\}. \quad (2.122)$$

The associated probability density function (PDF) of the discrete random variable is

$$\text{pdf}_p[l; m, n] = \Pr\{p[m, n] = l\}. \quad (2.123)$$

For a given pixel $[m, n]$, the *ensemble mean* or *expected value* is

$$E\{p[m, n]\} = \sum_{l=0}^{L-1} l \text{pdf}_p[l; m, n]. \quad (2.124)$$

and the *expected energy* is

$$E\{|p[m, n]|^2\} = \sum_{l=0}^{L-1} l^2 \text{pdf}_p[l; m, n]. \quad (2.125)$$

Second-order ensemble statistics relate random variables associated with two pixels in either the same image or in two different images. The *auto-correlation* function is the expected product (with conjugation of complex functions) of the two random variables for two pixels in the same image:

$$R_p[m_1, n_1; m_2, n_2] = E\{p[m_1, n_1]p^*[m_2, n_2]\}. \quad (2.126)$$

(The autocorrelation function is typically defined without reference to normalization[8].) The autocorrelation function of real images is symmetric (even). *Cross-correlation* is the expected product of pixels in different images:

$$R_{p,q}[m_1, n_1; m_2, n_2] = E\{p[m_1, n_1]q^*[m_2, n_2]\}. \quad (2.127)$$

Autocovariance is the covariance function of the random variables for two pixels in the same image:

$$C_p[m_1, n_1; m_2, n_2] = E\{p[m_1, n_1]p^*[m_2, n_2]\} \\ - E\{p[m_1, n_1]\}E\{p^*[m_2, n_2]\}. \quad (2.128)$$

Cross-covariance is the covariance function of the random variables for two pixels in different images:

$$C_{p,q}[m_1, n_1; m_2, n_2] = E\{p[m_1, n_1]q^*[m_2, n_2]\} \\ - E\{p[m_1, n_1]\}E\{q^*[m_2, n_2]\}. \quad (2.129)$$

A stochastic image is *stationary* if its statistics are invariant with respect to spatial shift. A stochastic image is stationary with respect to the mean if the expected value is constant across the image:

$$E\{p[m, n]\} = \eta_p. \quad (2.130)$$

A stochastic image is stationary with respect to autocorrelation if the expected product is a function of distance between points regardless of spatial position:

$$R_p[m_1, n_1; m_2, n_2] = R_p[m_1 - m_2, n_1 - n_2]. \quad (2.131)$$

A process that is stationary with respect to mean and autocorrelation is *wide-sense stationary* (or *weakly stationary*). A stochastic image that is stationary with respect to all statistics (including those of higher order) is *strictly stationary* (or *strongly stationary*). A stochastic image is strictly stationary if and only if the cumulative distribution functions of all pixels are identical:

$$\text{cdf}_p[l; m_1, n_1] = \text{cdf}_p[l; m_2, n_2] \quad (2.132)$$

for all gray-levels l and all pixels $[m_1, n_1]$ and $[m_2, n_2]$.

Frequently, the statistics of a stochastic image are unknown and must be estimated. One approach is to obtain several images from the ensemble and use the statistics of the sample set of images to infer the statistics of the ensemble. Another approach is to hypothesize ergodicity. A stochastic image is *ergodic* if the ensemble averages are equal to the appropriate spatial averages of the images. This definition is usually made with reference to infinite functions. If a stochastic process is mean-ergodic then the spatial average of an image from the ensemble is equal to the expected value of a pixel. Define a local average of an image:

$$\bar{p}_W [m, n] = \frac{1}{(W+1)^2} \sum_{m'=-W/2}^{W/2} \sum_{n'=-W/2}^{W/2} p [m - m', n - n']. \quad (2.133)$$

If the stochastic image is mean-ergodic, then

$$E \{p [m, n]\} = \lim_{W \rightarrow \infty} \bar{p}_W [m, n]. \quad (2.134)$$

A wide-sense stationary stochastic image is mean-ergodic if points at large distances from one another are uncorrelated. If a process is stationary, and correlation ergodic, the autocorrelation (Equations 2.126 and 2.131) is equal to the deterministic autocorrelation (Equation 2.40):

$$R_p = p \star p \quad (2.135)$$

where:

$$R_p [m, n] = \frac{1}{MN} \sum_{m'} \sum_{n'} p [m', n'] p^* [m' - m, n' - n]. \quad (2.136)$$

In Chapter 7, an ensemble is defined such that the autocorrelation function (or equivalently the power-spectrum) is the only restriction. This process is strictly stationary and correlation-ergodic.

References

- [1] Azriel Rosenfeld and Avinash C. Kak. *Digital Picture Processing*. Academic Press, Orlando, FL, second edition, 1982.
- [2] William. K. Pratt. *Digital Image Processing*. John Wiley and Sons, New York, NY, 1978.
- [3] Rafael C. Gonzalez and Paul Wintz. *Digital Image Processing*. Addison-Wesley, Reading, MA, second edition, 1987.
- [4] Kenneth R. Castleman. *Digital Image Processing*. Prentice-Hall, Englewood Cliffs, NJ, 1979.
- [5] B. R. Hunt. The application of constrained least squares estimation to image restoration by digital computer. *IEEE Transactions on Computers*, 22(9):805–812, 1973.
- [6] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, New York, NY, 1985.
- [7] Stephen K. Park. *Introduction to Digital Image Processing*. 1987. Unpublished Draft (Computer Science Department, College of William and Mary).
- [8] Athansios Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, New York, NY, 1984.

Chapter 3

The Imaging Process

3.1 Images of Physical Scenes

This chapter examines the kinds of imperfections that restoration techniques are designed to correct. At the most general level, the imaging process consists of acquiring a digital image of a physical scene and displaying the digital image on a display device (as is illustrated in Figure 3.1). In this dissertation, the CCD-array camera and the video display monitor are the archetypes for these processes, but the model generalizes a wide range of processes. In an ideal imaging system, the displayed image would be a perfect two-dimensional projection of the three-dimensional scene, but real imaging systems are imperfect. The mathematical model developed in this chapter captures the significant characteristics of conventional imaging systems.

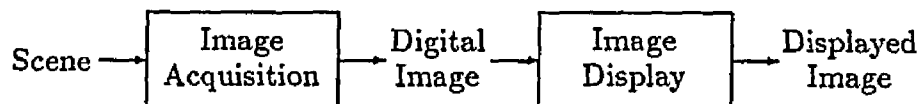


Figure 3.1: Imaging Process

3.2 Image Acquisition

A digital image acquisition system creates a digital image from a scene. The *scene* is infinite and continuous in three dimensions and each point has an associated *radiance* (emitted light) with a continuum of possible non-negative radiance values. A digital image is two-dimensional and has a finite number of pixels, each of which has an associated brightness value from a discrete

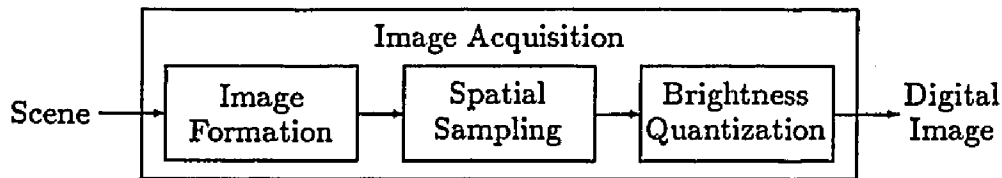


Figure 3.2: Image Acquisition

scale. Digital image acquisition can be divided into three steps: *image formation*, *spatial sampling*, and *brightness quantization*. The camera forms a finite, continuous, two-dimensional image on its *focal plane* (or *image plane*). Spatial sampling is the sensing of the *irradiance* (incident light) on the focal plane at discrete points. Brightness quantization is the measurement of the irradiance on a discrete scale. This division of the image acquisition process is illustrated in Figure 3.2.

3.2.1 Image Formation

The lens of a camera focuses light from the scene through an aperture to form an image. Gaskill[1] presents a detailed analysis of this process. If the lens is relatively free of aberrations and the object and image fields are not too large, the image formation process can be regarded as a system with three components: geometric projection, an image formation point spread function (PSF), and brightness scaling. Figure 3.3 illustrates this decomposition of the image formation process. In the context of image restoration, the significant image formation component is the PSF.

Geometric Projection The *perspective projection* of a three-dimensional scene into two dimensions is illustrated in Figure 3.4. Perspective projection is the geometric mapping produced by an idealized pinhole camera and is

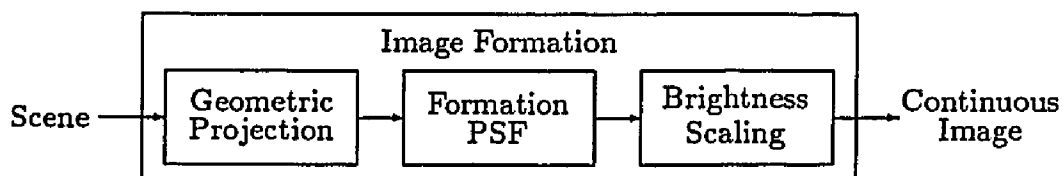


Figure 3.3: Image Formation

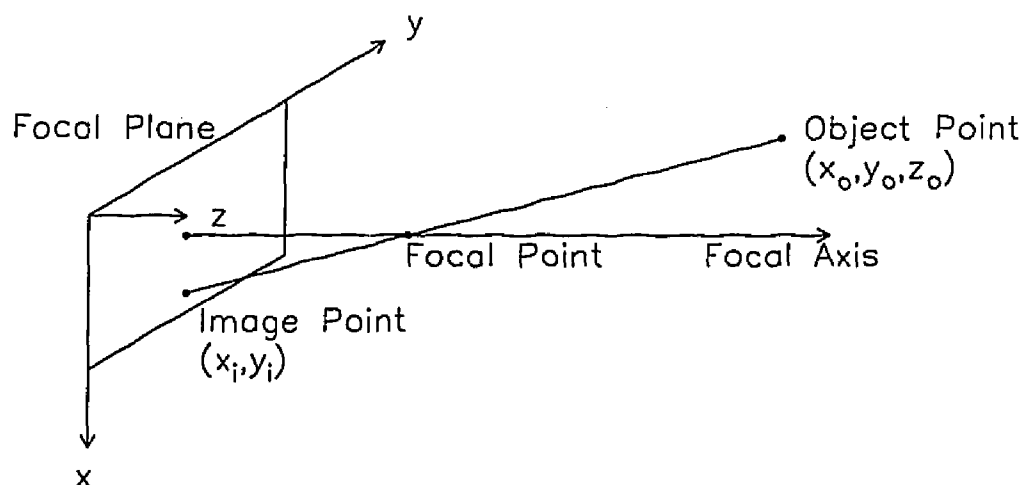


Figure 3.4: Perspective Projection

an approximate model of the geometric mapping that occurs in our eyes, in cameras, and in most other image acquisition devices. In practice, real image formation devices introduce some geometric distortion, particularly in the periphery of an image. Generally, wide-angle lenses introduce more geometric distortion and telephoto lenses cause less geometric distortion.

Typically, if the object field is small relative to the distance from the camera, geometric distortion is negligible. (Horn[2] and Ballard and Brown[3] discuss imaging geometry in more detail.) When geometric distortion is significant, it is usually considered separately from other restoration problems[4,5,6,7]. Therefore, geometric distortion is not incorporated into the imaging process model developed in this chapter. Instead, scenes are *defined* in two-dimensions as the ideal perspective projection onto the image plane.

Formation Point Spread Function The diffraction of a camera can be characterized by its PSF or equivalently by its optical transfer function (OTF). The PSF describes how light received from a point source in the scene is spread over a small region about the corresponding point in the image. (Degradations other than camera diffraction can be characterized in the same way. Two examples are uniform motion and atmospheric blur.) The scene is a continuum of point sources and the image is the linear superposition of the point-source responses. The PSF is typically a small, radially-symmetric spot that is generally-decreasing from the center out. The OTF is the Fourier transform of the PSF. Many of the degradations introduced by imaging systems are more easily pictured and understood in one dimension. Figure 3.5 illustrates a one-dimensional example PSF and OTF pair.

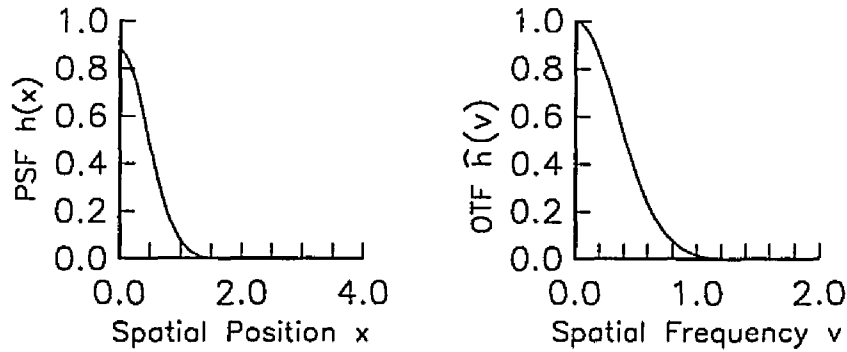


Figure 3.5: Example Point Spread Function and Optical Transfer Function

In a real system, the PSF will vary slightly with respect to position (i.e., it is shift-variant), particularly in the periphery, far from the focal axis. The PSF will also vary according to the object distance in the scene. If the object field is small, these variations are typically negligible and the diffraction of the camera is accurately modeled as a LSI operator. The modulation transfer function (MTF) of a wide range of image acquisition devices is accurately modeled as[8]

$$\hat{h}(u, v) = \exp\left(-\left(\sqrt{u^2 + v^2}/\alpha\right)^\beta\right) \quad (3.1)$$

where α is the spatial frequency at which the MTF is $\exp(-1)$ (approximately 0.37) and β determines the shape of the MTF. If $\beta = 2$, this is the two-dimensional *bell* (described in Appendix A). The one-dimensional system pictured in Figure 3.5 has $\alpha = 0.5$ and $\beta = 2$.

The PSF of the image formation process causes *blurring* of the scene, particularly of fine details. Both the scene and the PSF are continuous functions. The convolution of the scene s and formation PSF h_1 is

$$s' = s * h_1 \quad (3.2)$$

where

$$s'(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} s(x', y') h_1(x - x', y - y') dx' dy'. \quad (3.3)$$

Figure 3.6 illustrates the blurring of a one-dimensional example scene caused by convolution with the PSF illustrated in Figure 3.5.

Equivalently, multiplication by the image formation OTF \hat{h}_1 *attenuates* (diminishes) the high frequencies in the scene spectrum \hat{s} :

$$\hat{s}' = \hat{s} \hat{h}_1 \quad (3.4)$$

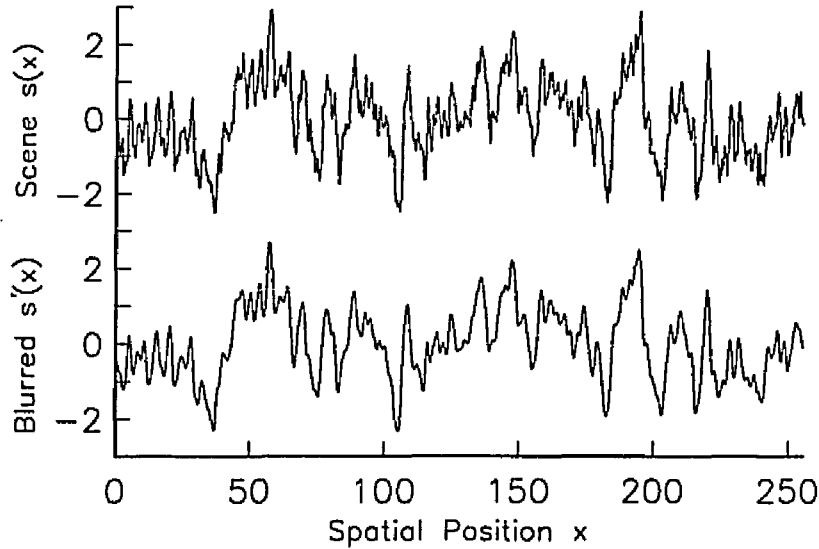


Figure 3.6: Spatial Blurring

where

$$\hat{s}'(u, v) = \hat{s}(u, v) \hat{h}_1(u, v). \quad (3.5)$$

Figure 3.7 illustrates the attenuation of the spectrum of the scene in Figure 3.6 caused by multiplication by the OTF in Figure 3.5. (The graph is of the magnitude of the complex-valued spectrum. The spectrum of a real-valued scene is Hermitian, so its magnitude is symmetric. Therefore, only half the spectral magnitude function is shown.) Traditionally, removing image blur (or high-frequency attenuation) has been the major objective of image restoration.

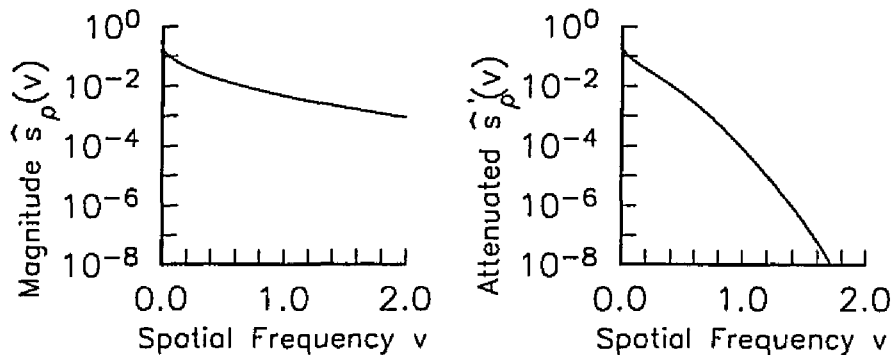


Figure 3.7: Frequency Attenuation

Brightness Scaling The brightness of an image is the measure of irradiance on the focal plane. The brightness of the ideal image is proportional to the radiance of the corresponding point in the scene. In a real system, there is some loss of brightness due to absorption and reflectance in the lens elements. Brightness scaling may also occur during brightness quantization. If the brightness scaling is linear (i.e., multiplication by a scalar constant), only the constant of proportionality (with the scene radiance) is changed. Non-linear scaling is more troublesome, but can be addressed independently of techniques that deal with other degradations. In many applications, the user of the restoration technique may wish to retain separate control over the brightness scale of the displayed image. In order to concentrate on more central problems and simplify the mathematics, brightness scaling is not explicitly included in the imaging system model that is used in this dissertation. If desired, brightness scaling can be incorporated into the definition of the acquisition PSF.

3.2.2 Spatial Sampling

Many types of devices are used to sense the irradiance of the image formed by the camera. Vidicon-tube cameras record incident light using an electrically-charged grid. Other systems employ photosensitive cells (e.g., charged-coupled devices or CCDs) that store charges in charge-potential wells for subsequent readout. Photosensitive cells are used in flying-spot scanners (one cell and a moving aperture), line scanners (a one-dimensional array of cells), and array sensors (a two-dimensional array of cells).

An ideal sample of the continuous image s' at (x, y) is determined by integrating the irradiance over an infinitesimal area about the point. Mathematically, this is accomplished by multiplying the image by a shifted *impulse* (or *dirac delta function*) δ . (The impulse is a generalized function that is nonzero over an infinitesimal area and has unit volume. It is described in Appendix A.)

Ideally, the sampled or discrete image p' is the product of the continuous image formed on the focal plane s' and a lattice of impulses. A rectangular grid is used almost exclusively as the sampling lattice. This corresponds to a sampling function called a two-dimensional *comb* III (or *shah* or *bed-of-nails*, described in Appendix A). Sampling on this lattice yields

$$p' = s' \text{III} \quad (3.6)$$

where

$$p'(x, y) = s'(x, y) \text{III}(x, y)$$

$$= \begin{cases} s'(x, y) & \text{if } x \text{ and } y \text{ are integers} \\ 0 & \text{otherwise} \end{cases} \quad (3.7)$$

or equivalently

$$p'[m, n] = s'(m, n) \quad (3.8)$$

for all pixels $[m, n]$. In a real system, there is undoubtedly some local geometric deviation from a true two-dimensional comb function. However, minor pixel spacing variations can usually be ignored. When they exist, more serious geometric distortions must be addressed as a separate problem.

The sampled values in real systems are not taken by integrating over an infinitesimal area, but are instead a function of the incident light in a small region or *neighborhood* around the sample point. The integration function is the *sensor response function* or sensor PSF h_2 :

$$p' = (s' * h_2)\text{III} \quad (3.9)$$

where

$$p'(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} s'(x', y') h_2(x - x', y - y') dx' dy' \text{III}(x, y) \quad (3.10)$$

or equivalently

$$p'[m, n] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} s'(x, y) h_2(m - x, n - y) dx dy \quad (3.11)$$

Figure 3.8 illustrates the two components of this process—integrating over a region and sampling the result.

Talmi and Simpson[9] and Hopwood[10] suggested an idealized model for how photon flux is collected about the sample points of line-scan devices. Figure 3.9 illustrates the sensor PSF for this idealized model. It is presented to illustrate how a system might integrate light over a small region about each sample point. This one-dimensional model can be extended to two-dimensions for sensor arrays by assuming separability.

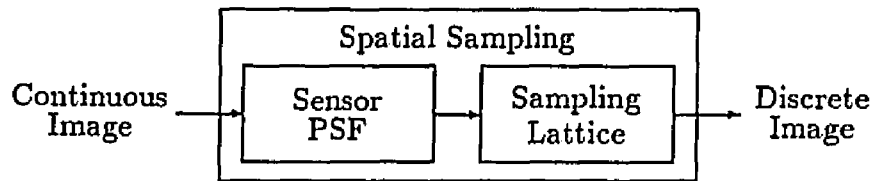


Figure 3.8: Spatial Sampling

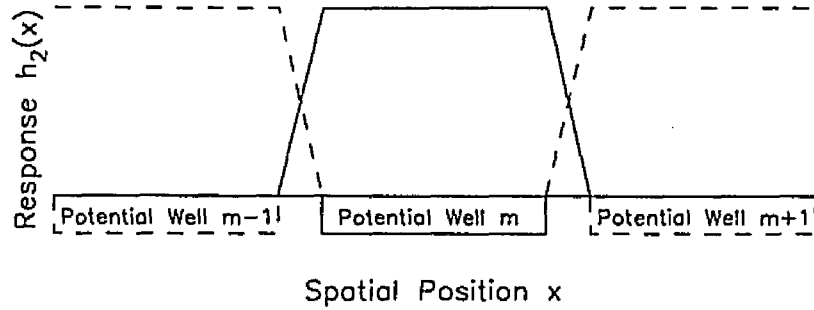


Figure 3.9: Idealized Model of Sensor Point Spread Function

Good detectors are linear over a wide range of irradiance. CCD detectors are especially linear. Nonetheless, even CCD detectors exhibit some non-linearity. For example, when the well of a CCD is saturated, charge flows into neighboring wells. Beal, et al.[11], found some charge leakage from wells filled to less than one-third capacity and significant leakage at 60% capacity. The sensor PSF of a good detector array is nearly the same for all array elements (i.e., the sensor PSF is shift-invariant). Nonetheless, there will be minor variations and perhaps even some elements with very different characteristics (e.g., a dead cell).

Sampling an image introduces an artifact called *aliasing*. Aliasing occurs when the sampling density is insufficient to capture the details in the image. Figure 3.10A illustrates a waveform that is sufficiently sampled. Figure 3.10B illustrates a waveform that is insufficiently sampled. The samples of the waveform in Figure 3.10B are indistinguishable from the samples of the lower-frequency waveform in Figure 3.10A.

Aliasing is more easily analyzed in the frequency domain. Sampling—spatial multiplication of the image by an array of impulses—is equivalent to frequency convolution of the scene spectrum \hat{s}' and the transform of the sampling lattice $\hat{\mathbb{I}}$:

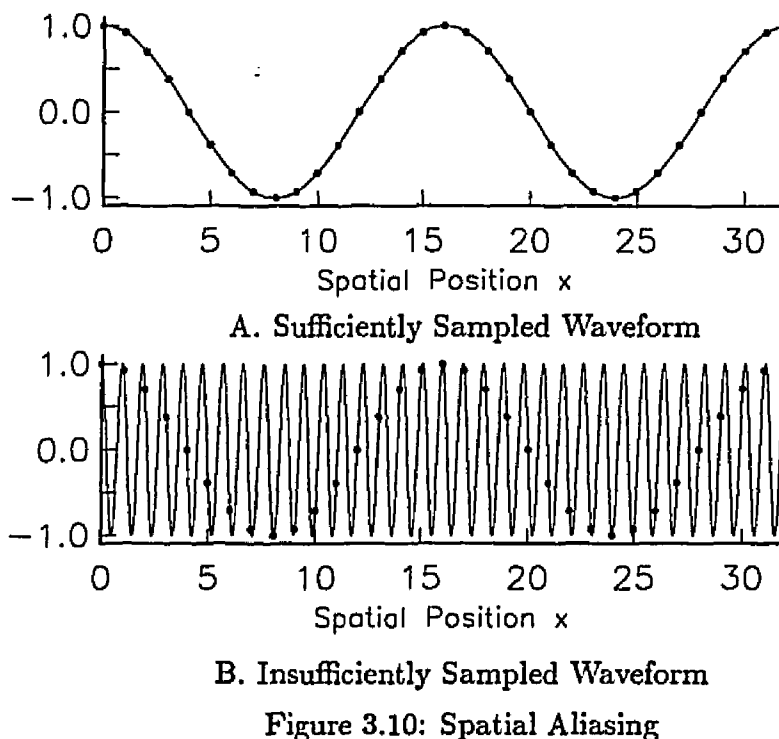
$$\hat{p}' = \hat{s}' * \hat{\mathbb{I}} \quad (3.12)$$

where

$$\hat{p}'(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \hat{s}'(u', v') \hat{\mathbb{I}}(u - u', v - v') du' dv'. \quad (3.13)$$

The comb function is its own transform. Therefore the convolution can be rewritten as a sum:

$$\hat{p}'(u, v) = \sum_{u'=-\infty}^{\infty} \sum_{v'=-\infty}^{\infty} \hat{s}'(u - u', v - v'). \quad (3.14)$$



The spectrum \hat{p}' is periodic with unit periods. (The periods are one because the unit spatial intervals were defined as the sampling intervals.)

A one-dimensional example is illustrated in Figure 3.11. (This example is not the frequency domain counterpart of Figure 3.10.) In Figure 3.11A, the image spectrum is zero beyond the *Nyquist limit* (i.e., at frequencies greater than half the sampling frequency— $|\nu| > 0.5$). In this case, when the frequency convolution replicates the spectrum (as in Equation 3.14), there is no overlap and no aliasing. In Figure 3.11B, the scene spectrum extends beyond the Nyquist limit. When the spectrum is convolved with the comb, the replicated *sidebands* ($|\nu| > 0.5$) fold back (i.e., are aliased) into the *baseband* ($|\nu| < 0.5$).

A prefilter (before sampling) can be used to reduce spectral components above the Nyquist limit in order to minimize aliasing. The ideal low-pass transfer function is the unit pulse described in Appendix A (1 where $|\nu| < 0.5$ and 0 where $|\nu| > 0.5$). Unfortunately, the corresponding spatial function, the *sinc* (also defined in Appendix A), has infinite extent, negative values and is not physically realizable. Moreover, a sharp cutoff in the spectrum of the prefilter causes *ringing* about sharp edges in the displayed image (*Gibb's phenomenon*), an unacceptable artifact. Also, optical lenses cannot realize sharp cutoffs. Therefore, the prefilter transfer function must roll off gradually. The tradeoff in the design of the roll-off is between aliasing (where the

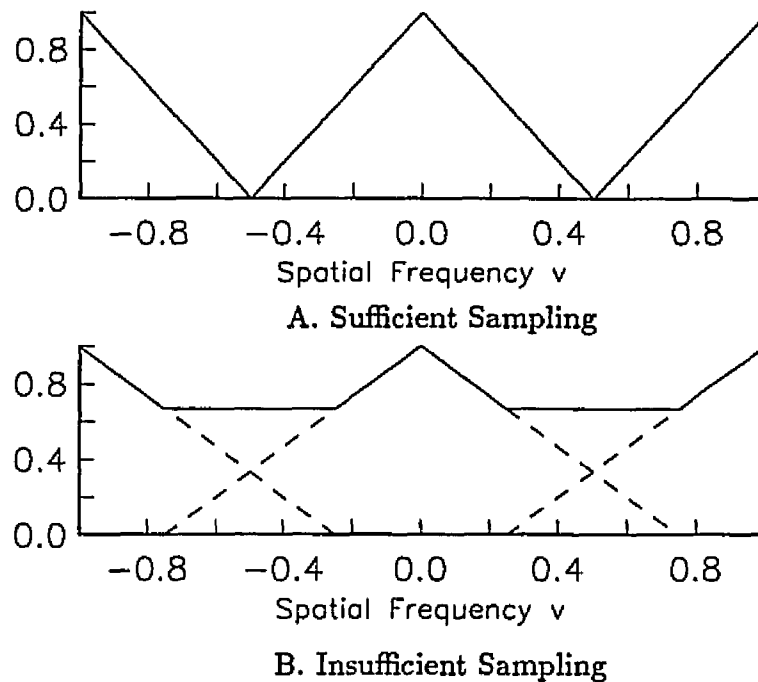


Figure 3.11: Frequency Folding

filter passes frequencies beyond the Nyquist limit) and blurring (where the filter attenuates frequencies within the Nyquist limit). Systems are usually designed so that there is some aliasing and some blurring[12]. In most systems, the OTF of the acquisition device serves as the prefilter and should be a good match to the sampling array. Restoration techniques must address both blurring and aliasing, though the importance of aliasing for restoration only recently has been considered fully[13].

A real sampled image p' is necessarily finite. The samples beyond the image border are unknown. As stated in Chapter 2, the assumption employed in this dissertation is that the finite image is one period of an infinite, periodic image. This assumption is not realistic (nor is any alternative), but it facilitates spectral analysis and the artifacts generally are limited to a small band near the image border.

3.2.3 Brightness Quantization and Noise

As the irradiance of the image is sampled, it must be *quantized* (converted to a finite, discrete scale with an analog-to-digital or A/D converter). A gray-scale employing eight-bit integers, allowing values 0 to 255, is a popular

choice. Quantization introduces errors. If the magnitude of the sample value exceeds some finite maximum, it must be *thresholded* to the top of the scale. This problem can be eliminated in subsequent images by adjusting the camera shutter or aperture or changing the scale. *Rounding* or *truncating* the measures to discrete values also introduces errors. These errors can be viewed as a random, additive *noise* process. The probability density function (pdf) for rounding noise is uniform in the range -0.5 to 0.5 . The pdf for brightness truncation noise is uniform over the range 0.0 to 1.0 .

Other sources of noise can plague real imaging systems. The randomness of the scene's photon flux means that successive images of a scene will differ (albeit slightly). More importantly, the circuits of the sensor (e.g., the A/D converters) add noise. This noise is often modeled as *white* noise. Noise is white if its values are spatially uncorrelated. That is, e is white noise if and only if for every two distinct points $[m_1, n_1] \neq [m_2, n_2]$ the expected product is the product of the expected values:

$$E \{e [m_1, n_1] e [m_2, n_2]\} - E \{e [m_1, n_1]\} E \{e [m_2, n_2]\} = 0. \quad (3.15)$$

Strictly white noise is not only spatially uncorrelated, but also independent. Unless otherwise stated, it will be assumed that white noise is wide-sense stationary and zero-mean, which means that the noise autocorrelation is a scaled impulse and the power spectrum is constant for all frequencies (except at the frequency origin).

Image restoration algorithms are usually based on the assumption of additive, white noise, but imaging systems can introduce other types of noise. For example, noise can have a fixed pattern or it may be signal dependent; it can be additive, multiplicative, or effect the image in more complicated ways. In this dissertation, sensor noise is modeled as an additive, signal-independent, stationary process.

3.2.4 Simplified Acquisition Model

The restoration algorithm developed in this dissertation is based on a simplified acquisition model that contains only the most important components of the image acquisition process. The simplified model is illustrated in Figure 3.12. It is assumed that there is no significant geometric distortion in the image formed by the camera. Therefore, the input to this simplified model is a two-dimensional brightness function s that is the ideal image—the perfect projection of a three-dimensional scene. Further, the scene is presumed to be doubly periodic with periods M and N .

The first source of degradation in the model is the acquisition PSF—the result of cascading the formation PSF (h_1 in Equation 3.2) and the sensor

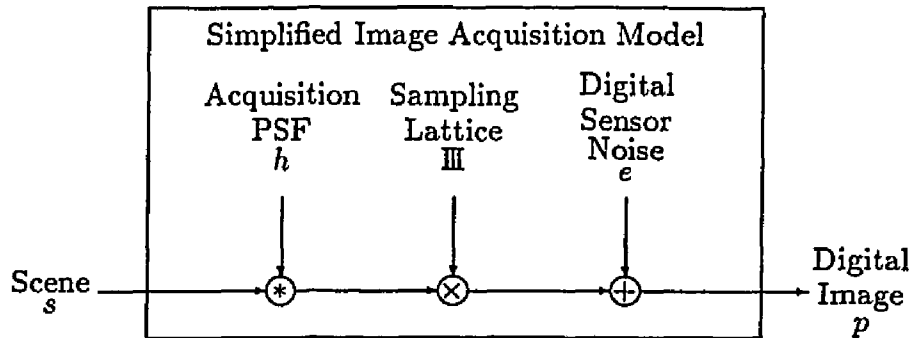


Figure 3.12: Simplified Imaging Acquisition Model

PSF (h_2 in Equation 3.9). Because convolution is associative, the output of the successive convolutions of the scene s , the image formation PSF h_1 , and the sensor PSF h_2 , $s * h_1 * h_2$, can be written as the convolution of the scene with a single PSF, $s * h$ where $h = h_1 * h_2$. The simplified model does not include brightness scaling explicitly, but the model of the acquisition PSF h can be specified to scale the image brightness linearly. The simplified model also assumes the sampling grid is a perfect two-dimensional comb function, avoiding the problems of geometric distortion or error and the issue of geometric transformations. Quantization error and circuit noise are combined as a single error term e . Following many others, noise is assumed to be additive, stationary, and signal-independent. Such noise is found in most systems and allows relatively straightforward analyses.

To summarize, the mathematical model of the acquisition process shown in 3.12 is

$$p = (s * h) \text{III} + e \quad (3.16)$$

where

$$p[m, n] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} s(x, y) h(m - x, n - y) dx dy + e[m, n]. \quad (3.17)$$

It may seem to those unfamiliar with image restoration research that a great deal of the imaging process has been ignored in this simplified model. Unfortunately, some simplification is unavoidable. Some of the degradations that have been ignored are significant in particular applications, but the most important characteristics of typical imaging acquisition systems are contained in this model. Historically, image restoration research has been conducted using an even simpler model that contains only a LSI acquisition function and additive noise. It is a common practice to assume that the scene is sufficiently sampled and to omit sampling from the acquisition model. However, most digital imaging systems are designed to undersample and therefore sampling is an important consideration that should not be ignored.

3.3 Display

The effect of the display device is often neglected in image processing research. (For example, display is not an index term in two popular image processing textbooks[6,7].) However, the display device directly affects the viewer's perception of the quality of the restoration. A video display monitor reconstructs a continuous image from a digital image by producing a spot on the screen for each pixel value. The display device is accurately modeled as a LSI system. The reconstructed image r is the convolution of the digital image p and the display PSF d :

$$r = p * d \quad (3.18)$$

where

$$r(x, y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} p[m, n] d(x - m, y - n) dx dy. \quad (3.19)$$

The sinc function is the ideal reconstruction function for a sufficiently sampled system. Any sufficiently sampled image can be exactly reconstructed from its samples by convolution with the sinc function. (This is the famous sampling theorem.) Unfortunately, the sinc function cannot be realized by physical displays. Moreover, because acquisition devices are designed to admit some aliasing (insufficient sampling) to limit blurring, it is unrealistic to assume sufficient sampling.

Realizable CRT display transfer functions must roll off smoothly. There is a tradeoff in display system design. On one hand, if the display transfer function rolls off within the Nyquist limit, then spectral components of the image are attenuated[14]. The result is a blurred image as illustrated in one dimension in Figure 3.13; the sharp step is blurred by the display. On the other hand if the display transfer function extends beyond the Nyquist

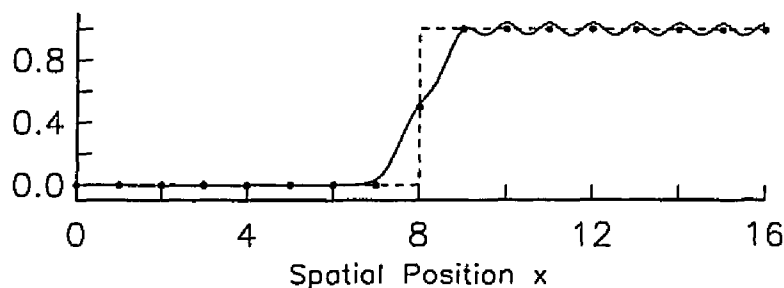


Figure 3.13: Display Blur and Ripple

limit, then the displayed image will contain components of the periodic sidebands. The resulting effects are called *post-aliasing*, *sample-frequency ripple*, or *anisotropy*[15]. The ripple pattern is evident on the top of the plateau in Figure 3.13. A primary consideration of display system design is to that a region of uniform brightness appear uniform[16,17].

Schade found that the display spot of a video monitor is accurately modeled as the sum of two Gaussians[18]—one for the nucleus of the spot and one for a flare about the nucleus due to “the finite thickness of the phosphor and optical reflections of the faceplate surfaces.”[18, p. 271] The common practice of using a single Gaussian (e.g., Castleman[4]) is less accurate.

The reconstruction PSF is a display’s most significant characteristic, but display devices also introduce noise. Display noise occurs in the form of variations in both intensity and spot position. Both types of noise, when random, will produce a salt and pepper pattern that is more evident in flat fields. Both types can also produce spurious fixed patterns that are also more evident in flat fields. Display noise is usually insignificant and is not included in the model. The display model is pictured in Figure 3.14.

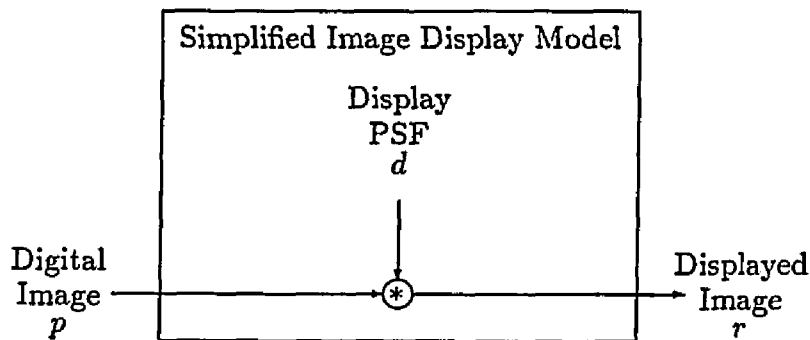


Figure 3.14: Simplified Imaging Display Model

References

- [1] Jack D. Gaskill. *Linear Systems, Fourier Transforms, and Optics*. John Wiley and Sons, New York, NY, 1978.
- [2] Berthold Klaus Paul Horn. *Robot Vision*. MIT Press, Cambridge, MA, 1986.
- [3] Dana H. Ballard and Christopher M. Brown. *Computer Vision*. Prentice-Hall, Englewood Cliffs, NJ, 1982.
- [4] Kenneth R. Castleman. *Digital Image Processing*. Prentice-Hall, Englewood Cliffs, NJ, 1979.
- [5] Rafael C. Gonzalez and Paul Wintz. *Digital Image Processing*. Addison-Wesley, Reading, MA, second edition, 1987.
- [6] William. K. Pratt. *Digital Image Processing*. John Wiley and Sons, New York, NY, 1978.
- [7] Azriel Rosenfeld and Avinash C. Kak. *Digital Picture Processing*. Academic Press, Orlando, FL, second edition, 1982.
- [8] C. B. Johnson. A method for characterizing electro-optical device modulation transfer functions. *Photographic Science and Engineering*, 14(6):413–415, 1970.
- [9] Yair Talmi and R. W. Simpson. Self-scanned photodiode array: A multi-channel spectrometric detector. *Applied Optics*, 19(9):1401–1414, 1980.
- [10] Ronald K. Hopwood. Design considerations for a solid-state image sensing system. In *Minicomputers and Microprocessors in Optical Systems*, pages 72–82, SPIE, 1980.
- [11] G. Béal, G. Boucharlat, J. Chabbal, J. P. Dupin, Fort B., and Y. Mellier. Thomson-CSF frame-transfer charge-coupled-device imagers: design and evaluation at low flux level. *Optical Engineering*, 26(9):902–910, 1987.
- [12] William F. Schreiber. *Fundamentals of Electronic Imaging Systems: Some Aspects of Image Processing*. Springer-Verlag, New York, NY, 1986.
- [13] Carl L. Fales, Friedrich O. Huck, Judith A. McCormick, and Stephen K. Park. Wiener restoration of sampled image data: end-to-end analysis. *Journal of the Optical Society of America A*, 5(3):300–314, 1988.

- [14] Stephen K. Park and Robert A. Schowengerdt. Image sampling, reconstruction, and the effect of sample-scene phasing. *Applied Optics*, 21(17):3142-3151, 1982.
- [15] Don P. Mitchell and Arun N. Netravali. Reconstruction filters in computer graphics. *Computer Graphics*, 22(4):221-228, 1988.
- [16] Otto H. Schade. Electro-optical characteristics of television systems: characteristics of vision and visual systems. *RCA Review*, 9:5-37, 1948.
- [17] Otto H. Schade. Electro-optical characteristics of television systems: electro-optical specifications for television systems. *RCA Review*, 9:245-286, 1948.
- [18] Otto H. Schade, Sr. Image reproduction by a line raster process. In Lucien M. Biberman, editor, *Perception of Displayed Information*, chapter 6, pages 233-278, Plenum Press, New York, NY, 1973.

Chapter 4

Traditional Restoration Techniques

4.1 Problems, Solutions, and Algorithms

This chapter describes several traditional image restoration techniques. Because the available knowledge of the scene and the degradations is invariably incomplete, image restoration is an estimation problem. From the digital image and what is known about the scene and the imaging system, a restoration technique must estimate the ideal image. The restored image should correspond closely to the ideal image, but it must be computed efficiently.

Practical methods are conditioned on a model of the scene and the imaging process. The model characterizes the knowledge (or assumptions) that can be used to solve the problem. Different techniques may be conditioned on different models; it is important to consider the accuracy of a model for each particular problem. The common models are reasonable for most systems, but they are motivated by practical considerations as well—the models must not make the problem intractable. To be successful, a method must capture the essential aspects of the problem in a simple model. If the technique is to be practical, the model must ignore some of the unpleasant complexities ever-present in real problems.

Practical techniques must yield an acceptable estimate. With incomplete knowledge, no method can always determine the correct solution. In many applications, a human observer is the best judge of the relative quality of possible solutions. Unfortunately, the criteria used by the human visual system are not well understood and even if human perception were fully modeled, it might be too complicated to be computed in a reasonable time with current technology. A practical criterion must be simple to calculate.

The most popular image restoration techniques use a single LSI operator. The LSI restoration operator is conditioned on assumptions about the imaging process and on judgements about which estimate is best. These assumptions and judgements are influenced by practical considerations. Non-linear methods and shift-variant (adaptive) methods are usually more complicated and computationally expensive. However, because even LSI operations on images require many calculations, computational complexity is often a critical concern. All of the techniques described in this chapter use a global, LSI operator. Chapters 5 and 6 describe techniques that reduce computation by using a local, LSI operator.

Traditional restoration techniques are successful because they define the model of the problem and the nature of the solution so that an effective and efficient algorithm exists. The models capture the important aspects of the restoration problem without being too complex, the measures of quality for the possible solutions correlate well with human perception, and the algorithm is tractable.

4.2 Inverse Filter

The first digital image restoration techniques focused on the problem of image blur (or high-frequency attenuation). One of the simplest techniques, the inverse filter, is based on the assumption that the imaging process is a non-singular, noise-free, LSI operation:

$$p = s * h \quad (4.1)$$

where s is the ideal image of the scene, h is the system PSF, and p is the degraded image. The ideal image of the scene s is unknown; the goal of restoration is to recover it from the digital image p using knowledge of the PSF h . Assuming the model of Equation 4.1, the ideal image can be recovered by applying the inverse of the system PSF:

$$\begin{aligned} r &= p * f \\ &= s * h * f \\ &= s * \delta \\ &= s \end{aligned} \quad (4.2)$$

where r is the restored image, and f is the PSF of the inverse filter ($h * f = \delta$).

The inverse filter assumes knowledge of the system PSF h . The system PSF can be estimated from the system response to test targets. Appendix B

describes an original technique for estimating the system functions of digital acquisition devices. Previous techniques for characterizing acquisition devices were designed for analog systems (e.g. film cameras) and rely on oversampling the analog output. Because digital systems are designed to undersample, these techniques do not accurately characterize digital systems. The technique described in Appendix B is specifically designed for digital systems and can accurately estimate the transfer function, even beyond the Nyquist frequency. The question of the existence of the inverse of the PSF is considered later in this section.

The inverse filter can be also applied in the frequency domain:

$$\begin{aligned}\hat{r} &= \hat{p}\hat{f} \\ &= \hat{s}\hat{h}\hat{f} \\ &= \hat{s}.\end{aligned}\tag{4.3}$$

Note that the system transfer function \hat{h} may be complex, so the inverse filter transfer function is defined as:

$$\begin{aligned}\hat{f} &= \hat{h}^{-1} \\ &= \frac{\hat{h}^*}{|\hat{h}|^2}\end{aligned}\tag{4.4}$$

Before the filter \hat{f} can be applied in the frequency domain, the image must be transformed. After the filter is applied, the resulting product must be inverse transformed. Except for rounding error differences, spatial domain processing and frequency-domain processing yield identical results.

The simple model of Equation 4.1 fails to account for several potentially important sources of degradation: sampling, system noise, and display reconstruction. Two early papers [1,2] that introduced inverse filtering to the digital image processing literature in 1966 addressed system noise, but failed to account for sampling and display. To this day, in the spirit of these early papers, traditional restoration techniques either ignore sampling and display or impose unrealistic assumptions to avoid the associated problems. Only recently have sampling and display been addressed directly[3,4]. In order to trace the historical development of traditional techniques, the inclusion of sampling and display in the system model is postponed until Section 4.6. Until then, restoration is posed as a problem without sampling or display reconstruction—the input scene, system functions, and system output are digital images with the same resolution as the degraded image.

Noise is a serious problem for the inverse filter. Restoration is an ill-conditioned problem—small errors in the image can result in large errors in

the restored image. In Equation 4.2, the restored image is identical to the ideal image of the scene. This equality is consistent with the model of the imaging system in Equation 4.1, but this model is unrealistic—some noise is invariably present. A more accurate model of the imaging process includes an additive noise term e :

$$p = s * h + e. \quad (4.5)$$

Using this model, the inverse filter restoration contains an error term:

$$\begin{aligned} r &= p * f \\ &= (s * h + e) * f \\ &= s + e * f. \end{aligned} \quad (4.6)$$

The frequency domain equations provide clearer insight into the nature of the ill-conditioning:

$$\begin{aligned} \hat{r} &= \hat{p}\hat{f} \\ &= (\hat{s}\hat{h} + \hat{e})f \\ &= \hat{s} + \hat{e}\hat{f}. \end{aligned} \quad (4.7)$$

That is, the restored image spectrum \hat{r} differs from the scene spectrum \hat{s} by the error term $\hat{e}\hat{f}$. Each component of the restored image spectrum is

$$\hat{r}[\mu, \nu] = \hat{s}[\mu, \nu] + \frac{\hat{e}[\mu, \nu]}{\hat{h}[\mu, \nu]} \quad (4.8)$$

If a system transfer function value $\hat{h}[\mu, \nu]$ is very small, then even a small amount of image noise at this frequency can result in a large error. In the worst case, $\hat{h}[\mu, \nu]$ is zero. Then the system is singular (i.e., cannot be inverted) and the image cannot be fully restored even if there is no noise.

In pioneering work at the NASA Jet Propulsion Laboratory, Nathan [1] recognized that the inverse filter was susceptible to noise. To address this problem, he placed an empirical upper-limit of 5 on the magnitude of the inverse filter transfer function to yield the restoration filter

$$\hat{f}[\mu, \nu] = \begin{cases} \hat{h}[\mu, \nu]^{-1} & \text{if } |\hat{h}[\mu, \nu]| > 0.2 \\ 5 & \text{otherwise.} \end{cases} \quad (4.9)$$

Nathan's modification of the inverse filter was based on the images to which the filter was applied (images transmitted by the Ranger and Mariner spacecraft). Subsequent techniques have attempted to incorporate more knowledge of the scene and noise into the system model to guide modification of the inverse filter.

4.3 Wiener Filter

The Wiener filter takes its name from an analogous one-dimensional result first established by Norbert Wiener [5] in the 1940's. The Wiener filter minimizes the expected mean-square error between the scene and the restored image:

$$S^2 = \frac{1}{MN} \sum_m \sum_n |s[m, n] - r[m, n]|^2. \quad (4.10)$$

The Wiener filter is also called the least-squares filter and the minimum mean-square-error filter. The method is premised on the model of Equation 4.5.

Most derivations of the filter treat the scene and noise as stochastic processes and assume that the noise and the the scene are independent, that both processes are stationary, and that their power spectra are known. The power spectra of the scene and noise are seldom known, but approximations are usually adequate. The Wiener filter is conditioned on the desirability of minimizing expected mean-square restoration error (MSRE). The MSRE measure does not correspond exactly to human perception of image quality, but it is generally satisfactory and it is easily computed. The popularity of the Wiener filter attests to its utility.

Helstrom [6] suggested Wiener's approach for restoring digital images. (See also Slepian [7].) The filter is typically defined in the Fourier frequency domain as

$$\hat{f} = \frac{\hat{h}^* \Phi_s}{|\hat{h}|^2 \Phi_s + \Phi_e} \quad (4.11)$$

where Φ_s is the scene power spectrum and Φ_e is the noise power spectrum. At frequencies where the scene power spectrum is zero ($\Phi_s[\mu, \nu] = 0$), the transfer function of the filter should be zero. At other frequencies, the filter is

$$\hat{f}[\mu, \nu] = \frac{\hat{h}^*[\mu, \nu]}{|\hat{h}[\mu, \nu]|^2 + \frac{\Phi_e[\mu, \nu]}{\Phi_s[\mu, \nu]}}. \quad (4.12)$$

At frequencies where the noise power spectrum is zero ($\Phi_e[\mu, \nu] = 0$), the Wiener filter is identical to the inverse filter. As the ratio of the energy of the noise to that of the scene increases, the value of the Wiener filter transfer function is attenuated. Nathan[1] attenuated the transfer function of the inverse filter using an arbitrary threshold that was not necessarily related

to the scene or noise. The Wiener filter uses the ratio of the noise energy to the scene energy to attenuate the restoration filter. The mathematical development of the Wiener filter is presented in Chapter 6.

The expected mean-square error of the Wiener filter restoration is

$$S_W^2 = \sum_{\mu} \sum_{\nu} \frac{\Phi_s[\mu, \nu] \Phi_e[\mu, \nu]}{|\hat{h}[\mu, \nu]|^2 \Phi_s[\mu, \nu] + \Phi_e[\mu, \nu]}. \quad (4.13)$$

If noise is negligible, S_W^2 approaches zero. If noise dominates the signal, S_W^2 is bounded by the energy of the signal. In comparison, the expected mean-square error of the inverse filter (using the assumptions about the signal and noise used to derive the Wiener filter) is

$$S_I^2 = \sum_{\mu} \sum_{\nu} \frac{\Phi_e[\mu, \nu]}{|\hat{h}[\mu, \nu]|^2}. \quad (4.14)$$

The expected mean-square error of the inverse filter restoration is an unbounded function of the noise. At each frequency, the expected square-error of the inverse filter restoration is at least as large as that of the Wiener filter restoration.

4.4 Constrained-Least-Squares Filters

Constrained-least-squares restoration minimizes a measure of the restored image subject to a mean-square-error constraint. The constraint expresses the expectation that the restored image is consistent with the knowledge of the system and noise—that given the expected level of noise the restored image could have accounted for the degraded image. In the frequency domain, this constraint is

$$\sum_{\mu} \sum_{\nu} |\hat{p}[\mu, \nu] - \hat{h}[\mu, \nu] \hat{r}[\mu, \nu]|^2 = \sum_{\mu} \sum_{\nu} \Phi_e[\mu, \nu]. \quad (4.15)$$

Many possible solutions may be consistent with this constraint. From among the possible solutions, the image selected is the one that minimizes

$$S^2 = \sum_{\mu} \sum_{\nu} |\hat{c}[\mu, \nu] \hat{r}[\mu, \nu]|^2 \quad (4.16)$$

where c is an empirical characteristic function.

The constrained restoration filter is

$$\hat{f} = \frac{\hat{h}^*}{|\hat{h}|^2 + \alpha |\hat{c}|^2} \quad (4.17)$$

where the non-negative scalar α is adjusted to satisfy the constraint in Equation 4.15. The value of α must be determined numerically.

Phillips [8] suggested that the smoothest of solutions consistent with Equation 4.15 be selected, so he used the discrete second derivative for c . His technique was refined by Twomey [9] and applied to image restoration by Hunt[10].

Using Φ_c/Φ_s for \hat{c} in Equation 4.17 yields the parametric Wiener filter:

$$\hat{f} = \frac{\hat{h}^*}{|\hat{h}|^2 + \alpha \frac{\Phi_c}{\Phi_s}} \quad (4.18)$$

which includes the traditional Wiener filter as the special case $\alpha = 1$. In practice, the adjustable parameter α has proven to be useful because real systems are difficult to model and characterize precisely.

4.5 Minimizing the Composite Point Spread Function

Given a noiseless model of the imaging system $p = s * h$, the restored image r is the convolution of the scene s , the system PSF h , and restoration filter f . The restored image can be written as

$$\begin{aligned} r &= p * f \\ &= s * h * f \\ &= s * g \end{aligned} \quad (4.19)$$

where $g = h * f$ is the composite PSF. For the techniques described in this section, the system PSF h is a continuous function and the restoration PSF f is discrete, so the composite PSF g is continuous. These techniques define the kernel f so that the composite PSF g will have specified characteristics. The inverse filter is one such method—it is specified so that the composite PSF is the convolution identity function, that is $g = \delta$.

Forcing the composite PSF to approximate the impulse function, as does the inverse filter, can excessively magnify noise. Smith [11] proposed minimizing the spread of the composite function, but subject to a constraint on the energy of the noise in the restored image. The minimization function is the radius of gyration of the power density of the composite PSF g :

$$S = \left| \frac{\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x^2 + y^2) |g(x, y)|^2 dx dy}{\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |g(x, y)|^2 dx dy} \right|^{\frac{1}{2}}. \quad (4.20)$$

(Because $S \geq 0$, minimizing S^2 gives the same result.) The radius of gyration of the power density is minimized if the composite PSF is the impulse; if $g = \delta$, then $S = 0$. The constraint on the energy of the restoration noise is

$$\sigma^2 = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left| \sum_m \sum_n f[m, n] e(x - m, y - n) \right|^2 dx dy. \quad (4.21)$$

There is a tradeoff between spatial resolution and noise. If the composite PSF is narrow, the resolution of the restoration is high but the energy of the noise energy in the restored image is also high. If the composite PSF is wide, the noise energy is low but the resolution is also low. The value of σ^2 determines how much noise is tolerated and limits how much resolution is possible. An additional constraint sets the energy of the composite PSF to 1:

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |g(x, y)|^2 dx dy = 1. \quad (4.22)$$

The restoration filter f that minimizes Equation 4.20 while meeting the constraints of Equations 4.21 and 4.22 must be determined numerically.

Backus and Gilbert [12] create a filter that depends on two components. The first component is based on minimizing the spread of the composite PSF:

$$S_1 = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x^2 + y^2) |g(x, y)|^2 dx dy. \quad (4.23)$$

They constrain the composite PSF and restoration PSF to have unit volume:

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x, y) dx dy = 1 \quad (4.24)$$

$$\sum_m \sum_n f(m, n) = 1. \quad (4.25)$$

The filter that corresponds to this single component is

$$f_1[m, n] = \sum_{m'} \sum_{n'} A_1^{-1}[m, n; m', n'] \quad (4.26)$$

where

$$\begin{aligned} A_1[m_1, n_1; m_2, n_2] \\ = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x^2 + y^2) h(x - m_1, y - n_1) h(x - m_2, y - n_2) dx dy. \end{aligned} \quad (4.27)$$

The second component minimizes the noise in the restoration:

$$S_2 = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \sum_m \sum_n f[m, n] e(x - m, y - n) dx dy \quad (4.28)$$

subject to the constraint of Equation 4.25. The filter that corresponds to this second component is

$$f[m, n] = \sum_{m'} \sum_{n'} A_2^{-1}[m, n; m', n'] \quad (4.29)$$

where A_2 is the autocorrelation matrix R_e of the image noise.

Backus and Gilbert combine these two components. The tradeoff between minimizing the spread of the composite PSF and minimizing the energy of the noise is expressed as

$$A = A_1 \cos \theta + c A_2 \sin \theta \quad (4.30)$$

where the c and θ are chosen to compromise between increasing resolution and suppressing noise. The resulting filter is

$$f[m, n] = \sum_{m'} \sum_{n'} A^{-1}[m, n; m', n'] \quad (4.31)$$

where the filter is scaled to satisfy Equation 4.25.

Unlike the Wiener filter and the constrained-least-squares filter, filters that minimize the composite PSF are determined without regard to the character of the scene. This approach is appropriate when information about likely scenes is not available, but cannot take advantage of such knowledge when it is available. Like the conventional formulation of the Wiener filter for image processing, they also fail to account for sampling and display reconstruction.

4.6 End-to-End Wiener Filter

The conventional Wiener filter fails to account for degradations in the imaging process due to sampling and display reconstruction. Sampling effects are present unless the spectrum of the continuous image is zero beyond the Nyquist frequency (i.e., the image is sufficiently sampled). Most imaging systems are designed to insufficiently sample. Systems that sample sufficiently cause excessive blurring. Aliasing caused by insufficient sampling is a source of noise that should be addressed by restoration techniques. Display devices also degrade images. The sinc function is the ideal interpolator, but only if the scene is sufficiently sampled. Even if the scene were sampled sufficiently, the sinc function can not be physically realized by a display device. As described in Chapter 3, real display devices attenuate high frequencies and pass sideband components (post-aliasing). Because restorations are usually intended for redisplay, the degradations introduced by the display device should be considered in the development of the restoration filter.

Huck et al. [3] and Fales et al. [4] have recently extended the Wiener filter to account for sampling and display using an end-to-end model of the imaging process. The imaging model described in Chapter 3 includes sampling and a LSI display. In addition to the assumptions of the conventional formulation of the Wiener filter described in Section 4.3, the derivation of the end-to-end Wiener restoration filter assumes that the sidebands of the scene spectrum are uncorrelated. (This is detailed in the mathematical development of the end-to-end Wiener filter in Chapter 6.) The end-to-end filter is more complicated than the traditional discrete Wiener filter because additional parameters are required to account for the additional sources of degradation that are considered:

$$\hat{f} = \frac{\Phi_s \hat{h}^* \hat{d}^* * \hat{\mathbb{I}}}{\left(\Phi_s |\hat{h}|^2 * \hat{\mathbb{I}} + \Phi_e \right) \left(|\hat{d}|^2 * \hat{\mathbb{I}} \right)}. \quad (4.32)$$

If the image is sufficiently sampled and the display function is the sinc interpolator, then the end-to-end Wiener filter reduces to the discrete Wiener filter.

References

- [1] R. Nathan. *Digital Video Handling*. Technical Report 32-877, NASA, 1966.
- [2] James L. Harris, Sr. Image evaluation and restoration. *Journal of the Optical Society of America*, 56(5):569–574, 1966.
- [3] Friedrich O. Huck, Carl L. Fales, Nesim Haylo, Richard W. Samms, and Kathryn Stacy. Image gathering and processing: Information and fidelity. *Journal of the Optical Society of America A*, 2(10):1644–1666, 1985.
- [4] Carl L. Fales, Friedrich O. Huck, Judith A. McCormick, and Stephen K. Park. Wiener restoration of sampled image data: end-to-end analysis. *Journal of the Optical Society of America A*, 5(3):300–314, 1988.
- [5] N. Wiener. *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*. MIT Press, Cambridge, MA, 1949.
- [6] C. W. Helstrom. Image restoration by the method of least squares. *Journal of the Optical Society of America*, 57(3):297–303, 1967.
- [7] D. Slepian. Linear least-squares filtering of distorted images. *Journal of the Optical Society of America*, 57(7):918–922, 1967.
- [8] David L. Phillips. A technique for the numerical solution of certain integral equations of the first kind. *Journal of the Association for Computing Machinery*, 9(1):84–97, 1962.
- [9] S. Twomey. On the numerical solution of fredholm integral equations of the first kind by inversion of the linear system produced by quadrature. *Journal of the Association for Computing Machinery*, 10(1):97–101, 1963.
- [10] B. R. Hunt. The application of constrained least squares estimation to image restoration by digital computer. *IEEE Transactions on Computers*, 22(9):805–812, 1973.
- [11] Harvey A. Smith. Improvement of the resolution of a linear scanning device. *SIAM Journal of Applied Mathematics*, 14(1):23–41, 1966.
- [12] G. Backus and F. Gilbert. Uniqueness in the inversion of inaccurate gross earth data. *Philosophical Transactions of the Royal Society of London A*, 266:123–192, 1970.

Chapter 5

Small-Kernel Restoration

5.1 Small-Kernel Convolution

Many imaging systems must operate within stringent time constraints, in *real time*. Because of the large number of pixels in typical images and the computational complexity of restoration, real-time image restoration is a challenging goal. As illustrated in Figure 5.1, LSI restoration techniques can be applied equivalently in either the spatial domain (with convolution) or in the spatial-frequency domain (with pointwise multiplication). The traditional techniques described in Chapter 4 are global operations—each element of the output is a function of every element of the input. Global operations are practical only when implemented in the frequency domain, but the hardware required for high-speed frequency-domain processing is too bulky and prohibitively expensive for many applications. This chapter examines some algorithms that restore an image by convolving it with a small kernel. Convolution with a kernel that has only a few values is an efficient approach that facilitates parallel implementation and adaptive processing.

Many of the most common image restoration techniques (e.g., the inverse filter, the Wiener filter, the parametric Wiener filter, and the constrained least-squares filter) are traditionally derived and implemented in the frequency domain. The restoration PSF associated with these methods is as large as the image— MN elements—so spatial convolution is $O(M^2N^2)$. For typical image sizes (e.g., 512×512 or 1024×1024), spatial convolution with a kernel as large as the image requires far too much computation and is impractical. Fast transforms allow the equivalent computation to be performed in the frequency domain with significantly fewer calculations. Frequency domain filtering requires only pointwise multiplication of the image spectrum \hat{p} and the filter transfer function \hat{f} and is therefore $O(MN)$. In most cases,

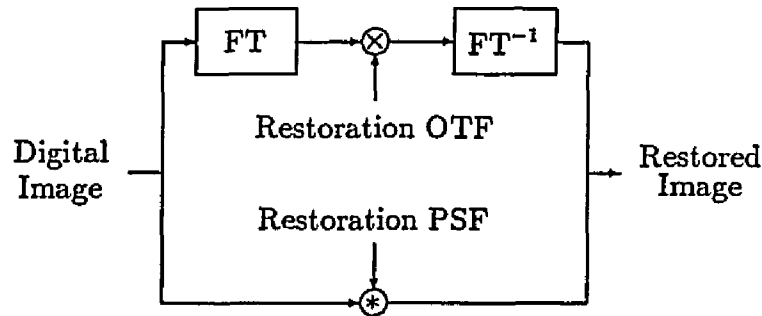


Figure 5.1: Spatial Domain and Frequency Domain Processing

this requires a Fourier transform to compute the image spectrum and an inverse Fourier transform to compute the resulting restored image. Fast Fourier transforms are $O(MN \log(MN))$.

Though fast transform algorithms significantly reduce the computation required for global restoration, substantial processing is still required. For example, for a $N \times N$ image, the Hartley transform[1], a very efficient transform for real-valued data, requires about $(3/2)N^2 \log_2 N$ multiplications and $(11/4)N^2 \log_2 N$ additions. For a 512×512 image, this is about 3.5 million multiplications and 6.5 million additions. On a Sun 3/260, a mid-priced engineering workstation, performing a forward and inverse fast two-dimensional Hartley transform on a 512×512 image and applying a filter by pointwise multiplication requires nearly four minutes. This is far too long for many applications.

Restoration can be computed more quickly in parallel—each pixel of the output can be computed independently of all other output pixels. Global operations are complicated to implement in parallel, but state-of-the-art, special-purpose hardware can perform real-time transforms on medium-sized images. In some applications, however, the bulk of this equipment or its cost is too large; in other applications, the images are too large.

The need for more efficient processing has motivated consideration of local algorithms—restoration implemented by convolution kernels that have only a few elements. If the kernel has very few elements, spatial processing requires fewer operations than frequency-domain processing. In restoration, as in most filtering operations, each output value is principally determined by the values in a relatively small region around the corresponding point in the input image. Therefore, local operations can be nearly as effective as global operations.

Spatial convolution is computationally practical only if the kernel is small. Spatial convolution of a $M \times N$ image with a K -element kernel is $O(KMN)$.

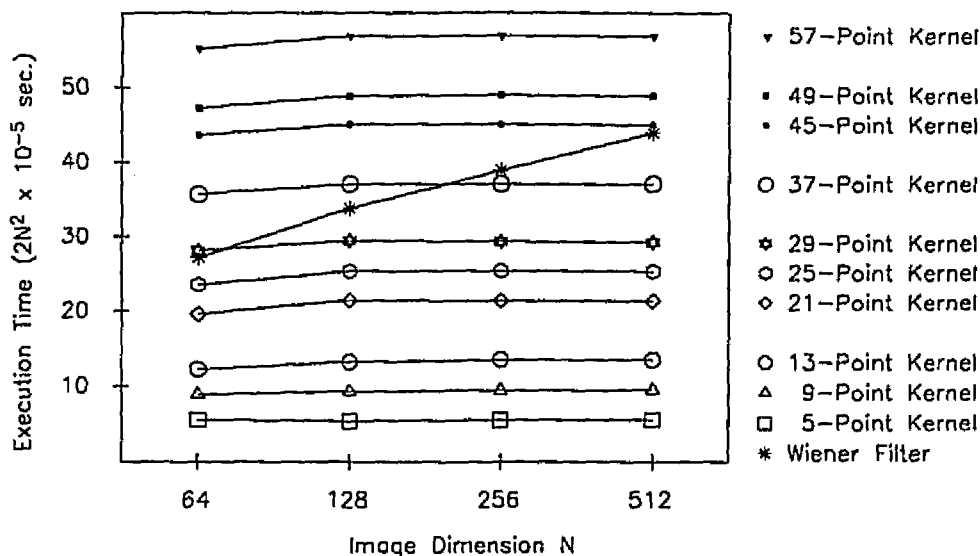


Figure 5.2: Execution Times

Execution times for small kernel convolution and frequency domain processing of a $N \times N$ image are given in Figure 5.2. The computations were executed in double precision on a Sun 3/260 with floating-point accelerator and 8MB of RAM. The frequency domain execution includes two discrete Hartley transforms[1] and a pointwise multiplication. It requires about $N^2 \lg(N) \times 10^{-4}$ seconds. Convolution requires about $2KN^2 \times 10^{-5}$ seconds. For this system, the breakeven point for convolution versus frequency-domain processing is about $K = 5 \lg(N)$ elements.¹ Small kernel restoration is efficient—inexpensive PC boards for real-time, 2D convolution with small kernels are widely available. Perhaps most importantly, local operations are easier to implement in parallel than global operations.

The challenge is to design small kernels that yield good results. This chapter reviews several techniques for designing small restoration kernels. One approach is to use traditional techniques to design a filter with a large PSF and then generate a small kernel from it. Some of these techniques are considered in Section 5.2. The methods of Section 5.3 derive the small kernels directly using measures of the composite PSF of acquisition and restoration. In Section 5.4, the minimum mean-square-error criteria of the Wiener filter is applied to the design of small restoration kernels.

¹The 64×64 images were processed most efficiently, though the difference is small. This is apparently due to the presence of 64KB of cache memory. The 64×64 image is the only image that will fit completely in the cache memory.

5.2 Small Kernels from Large Kernels

For typical imaging systems, the restoration PSF derived by any of the traditional methods has a peak at the origin and ripples outward. The most significant values of the restoration PSF are located near the origin and the magnitude of the ripples decreases with greater distance from the origin. One method for generating a small convolution kernel from a large PSF is to multiply the large PSF by a *window function* centered about the center of the PSF. The window function should leave the most significant values of the restoration PSF unchanged and replace the least significant values with zeros. The resulting kernel has fewer nonzero elements and can therefore be applied more efficiently.

A simple window function is the *pulse* (described in Appendix A). Applying a pulse window function is called *spatial truncation*. The two-dimensional pulse can be either radially symmetric or separable. The transfer function of the pulse is the sinc function. (See Appendix A.) However, the sharp cutoff of the pulse introduces an undesirable ringing in its transfer function, so several alternative window functions have been suggested[2]. One alternative, the Hanning window, is a raised-cosine function. The one-dimensional Hanning window is

$$w(x) = \begin{cases} \frac{1+\cos(\pi x/K)}{2} & \text{if } |x| < K \\ 0 & \text{otherwise} \end{cases} \quad (5.1)$$

where K is the half-width or radius of the window. The transfer function of the Hanning window is a waveform similar to the sinc, but the central lobe is wider and the magnitude of the sidelobes is greatly reduced[2].

Arguello et al.[3] subjectively compared the performance of the pulse and Hanning functions for windowing restoration PSFs and concluded that, as would be expected, the Hanning window worked better than the pulse. They also concluded that for the images they considered, restorations using 7×7 windowed kernels were subjectively not significantly different than restorations using full-size restoration kernels.

Windowing restoration PSFs is an *ad hoc* operation. The PSF is designed under the assumption that it will be used unmodified without regard to its cost or size. The windowing operation is based on processing limits without regard to the design of the PSF. There is no coordination between the restoration filter design and processing implementation. A more rigorous approach would account for processing limits in the design of the restoration kernel or for restoration results in the imposition of processing constraints.

Riemer and McGillem[4,5] modified the restoration method proposed by

Smith[6] to reduce the effect of windowing. (Smith's method is described in Chapter 4.) They added a constraint on the spread of the restoration PSF of the form

$$k = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c(x, y) |f(x, y)|^2 dx dy \quad (5.2)$$

where the empirical penalty function c weights the energy of the restoration kernel and k is a constant. The empirical penalty function is small near the origin and increases away from the origin to restrict the spread of the restoration PSF. The motivation is to keep the effective size of the restoration PSF small so that the effect of truncation is minimal. This constraint is in addition to the constraints suggested by Smith on restoration noise and composite PSF energy in minimizing the radius of gyration of the composite PSF power density.

Riemer and McGillem[4,5] account for windowing in the derivation of the restoration filter by imposing a penalty on the spread of the restoration PSF. Even so, the specific processing constraint (i.e., the actual window) is not explicit in the derivation of the restoration filter.

The least-squares method can be used to generate a small kernel to approximate a large PSF. Let \hat{h} be the transfer function of a previously defined restoration filter with a large PSF and \hat{h}' be the transfer function of a small kernel that approximates it. The least-squares approximation will minimize

$$S^2 = \sum_{\mu} \sum_{\nu} |\hat{h}[\mu, \nu] - \hat{h}'[\mu, \nu]|^2. \quad (5.3)$$

The individual elements of the small kernel h' are determined by substituting for the transfer function of the small kernel in Equation 5.3 using the equality

$$\hat{h}' = \mathcal{F}\{h'\} \quad (5.4)$$

and then differentiating S^2 with respect to the non-zero elements of h' . The result is a set of linear equations whose solution minimizes S^2 .

This procedure can be compactly represented using matrix notation:

$$\begin{aligned} S^2 &= \|\hat{\mathbf{h}}' - \hat{\mathbf{h}}\|^2 \\ &= \|\mathbf{W}\mathbf{h}' - \hat{\mathbf{h}}\|^2 \end{aligned} \quad (5.5)$$

where $\|\cdot\|$ is the complex Euclidean norm, $\hat{\mathbf{h}}$ is the matrix containing the MN terms of the transfer function of the restoration filter to be approximated, \mathbf{h}' is the matrix containing the $M'N'$ terms of the small kernel ($M' \leq M$ and $N' \leq N$), and \mathbf{W} is the $MN \times M'N'$ discrete Fourier transform matrix ($\hat{\mathbf{h}}' = \mathbf{W}\mathbf{h}'$). Differentiation yields

$$\frac{\partial S^2}{\partial \mathbf{h}'} = 2\mathbf{W}^* (\mathbf{W}\mathbf{h}' - \hat{\mathbf{h}}) \quad (5.6)$$

where the '*' exponent is the conjugate transpose. Setting this system of equations equal to zero and solving for the small kernel yields

$$\begin{aligned} \mathbf{h}' &= (\mathbf{W}^* \mathbf{W})^{-1} \mathbf{W}^* \hat{\mathbf{h}} \\ &= \mathbf{W}^\# \hat{\mathbf{h}} \end{aligned} \quad (5.7)$$

where $\mathbf{W}^\# = (\mathbf{W}^* \mathbf{W})^{-1} \mathbf{W}^*$ is the generalized inverse of the transform matrix.

The least-squares approximation in Equation 5.7 provides little more control over the imposition of spatial constraints on the kernel than does windowing. However, a weighted least-squares measure gives some control over the approximation and, indirectly, the restoration:

$$S^2 = \sum_{\mu} \sum_{\nu} \hat{c}(\mu, \nu) |\hat{h}[\mu, \nu] - \hat{h}'[\mu, \nu]|^2 \quad (5.8)$$

where \hat{c} is a penalty function. For example, Schutten and Vermeij[7] empirically set the penalty function to the inverse of the approximated transfer function \hat{h}^{-1} at frequencies where signal dominates noise (i.e., low frequencies) and to a minimal value where noise dominates signal (i.e., high frequencies). Schutten and Vermeij[7] attempt to shape the windowed approximation based on restoration quality. This approach designs a restoration filter and then designs an approximation to it; there is no direct coordination between the two steps.

5.3 Minimizing the Composite Point Spread Function

The techniques of Section 5.2 derived small kernels indirectly by first defining a restoration filter with a large PSF and then generating a small kernel approximation. The methods of this section impose a constraint on the size of the restoration kernel as a part of the problem statement so the derivation leads directly to a small restoration kernel. These techniques derive the small kernels by minimizing the composite PSF. The composite PSF g , introduced in Chapter 4, is the combined PSF of acquisition and restoration $g = h * f$. In this approach, the character of the scene is not considered; a direct method that accounts for the character of the scene is described in the next section.

Frieden[8] suggested a direct approach for generating small restoration kernels that is based on minimizing the magnitude of the side lobes of the composite PSF g while constraining the width of the central lobe. Generally, the width of the central lobe of the composite PSF is inversely related to

resolution—a PSF with a narrow central lobe has a wide frequency pass-band, while a PSF with a wide central lobe attenuates high frequencies. By fixing the width of the central lobe at x_0 , Frieden affects the resolution of the restoration. Frieden presents a one-dimensional model, so the constraint on central lobe width is

$$g(x_0) = 0. \quad (5.9)$$

Large side lobes will cause ringing adjacent to sharp edges. Frieden determines the maximum side lobe:

$$S = \frac{\max \langle |g(x)| : x > x_0 \rangle}{g(0)} \quad (5.10)$$

The value of S is minimized using a search procedure to determine the kernel values.

Stuller[9] applied the method of Smith[6] to the problem of determining a small restoration kernel. The radius of gyration of the power density of the composite PSF

$$S^2 = \frac{\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x^2 + y^2) |g(x, y)|^2 dx dy}{\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |g(x, y)|^2 dx dy} \quad (5.11)$$

is minimized subject to a constraint on the expected noise energy in the restored image:

$$E \{ |f * e|^2 \} = \sigma^2 \quad (5.12)$$

and a normalizing constraint:

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |g(x, y)|^2 dx dy = 1. \quad (5.13)$$

Posed as a discrete problem using matrices, the restoration kernel will minimize

$$S^2 = \frac{\mathbf{f}^* \mathbf{A} \mathbf{f}}{\mathbf{f}^* \mathbf{R}_h \mathbf{f}} \quad (5.14)$$

where \mathbf{A} is the matrix of weighted autocorrelation values of the acquisition PSF:

$$\begin{aligned} \mathbf{A} [m_1, n_1; m_2, n_2] \\ = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x^2 + y^2) h(x - m_1, y - n_1) h(x - m_2, y - n_2) dx dy \end{aligned} \quad (5.15)$$

and \mathbf{R}_h is the matrix of autocorrelation values of the acquisition PSF. Introducing Lagrange multipliers for the constraints of Equations 5.12 and 5.13 yields

$$F = \mathbf{f}^* \mathbf{A} \mathbf{f} - \lambda_1 (\mathbf{f}^* \mathbf{R}_h \mathbf{f} - 1) - \lambda_2 (\mathbf{f}^* \mathbf{R}_e \mathbf{f} - \sigma^2) \quad (5.16)$$

where \mathbf{R}_e is the matrix of noise autocorrelation values. Minimization requires $\partial F / \partial \mathbf{f} = 0$:

$$\mathbf{A} \mathbf{f} - \lambda_1 \mathbf{R}_h \mathbf{f} + \lambda_2 \mathbf{R}_e \mathbf{f} = \mathbf{0}. \quad (5.17)$$

Note that this is an eigenvector-eigenvalue equation:

$$\lambda_1 \mathbf{f} = \mathbf{R}_h^{-1} (\mathbf{A} + \lambda_2 \mathbf{R}_e) \mathbf{f}. \quad (5.18)$$

Solving Equation 5.17 for \mathbf{A} and substituting into Equation 5.14 yields

$$S^2 = \lambda_1 - \lambda_2 \sigma^2. \quad (5.19)$$

The procedure for determining the restoration filter values is to vary λ_2 . For each λ_2 , S^2 is minimized for the smallest eigenvalue λ_1 . The pair of values λ_1 and λ_2 that minimizes S^2 determines the values of the restoration kernel.

The techniques described in this section are based on minimizing the composite PSF. This approach does not require any knowledge of the scene, but neither can it make use of such knowledge when it is available. Nonetheless, these techniques have proven to be relatively successful in several experiments. Stuller[9] concluded that for the one-dimensional case, only a very small improvement in the composite PSF was obtained by using a restoration kernel exceeding five elements. Chu and McGillem[10] confirmed this conclusion for images with Gaussian blur using a generalized weighting function for the power density of the composite PSF—that is, in place of Equation 5.11, they used

$$S^2 = \frac{\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} w(x, y) |g(x, y)|^2 dx dy}{\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |g(x, y)|^2 dx dy}. \quad (5.20)$$

Saleh[11] used the method of Backus and Gilbert[12] to generate small kernels. Based on experiments with one-dimensional kernels of 9 elements and fewer, Saleh concluded that progressively increasing kernel size yielded diminishing improvements. All of these results indicate that small kernels can be effective at restoring images.

5.4 Small, Least-Squares Kernels

Lahart[13] used the minimum mean-square-error approach of the Wiener filter to generate spatial kernels. He derived the kernel in the spatial domain by minimizing the expected mean-square difference between the scene s and the restored image r :

$$S^2 = \sum_m \sum_n |s[m, n] - r[m, n]|^2 \quad (5.21)$$

Minimization with respect to the kernel elements requires $\partial S^2 / \partial f = 0$ and is realized when

$$E\{p * p\} * f = E\{s * p\} \quad (5.22)$$

Assuming the scene and noise are stationary processes and the noise is signal-independent yields

$$E\{p * p\} = h * R_s * h_- + R_e \quad (5.23)$$

$$E\{s * p\} = R_s * h_- \quad (5.24)$$

where R_s and R_e are the autocorrelation matrices of the scene and noise and h_- is the index reversed acquisition PSF ($h_-[m, n] = h[-m, -n]$).

Lahart limits the size of the restoration kernel by observing that $E\{p * p\}$ and $E\{s * p\}$ are small if the distance between points is greater than $2W_h + W_s - 2$ where W_h and W_s are the widths of the acquisition PSF h and the scene autocorrelation R_s , respectively. Lahart therefore limits the radius (or half-width) of the restoration kernel to $2W_h + W_s - 2$ by truncating the arrays in Equation 5.22 before solving. Lahart does not *impose* a constraint on the kernel; the size of the kernel is dictated by the character of the scene and the PSF.

While the width of the autocorrelation of the acquisition PSF is typically small, the width of the autocorrelation of the scene may be large. If the resulting restoration kernel is very large it is impractical. Lahart presents an example where the width of the acquisition PSF is 3 pixels and the width of the scene autocorrelation is 5 pixels. A scene with an autocorrelation width this small would have a great deal of fine detail. Many scenes would have wider scene autocorrelation functions. Even so, the resulting restoration kernel is 17 pixels across. A 17×17 convolution kernel cannot be efficiently applied in the spatial domain.

The algorithm developed in the next chapter derives mean-square-optimal restoration kernels, but with three significant differences from Lahart's work. First, the motivation for constraining the kernel is to limit processing. Lahart

derived kernels to implement adaptive processing and allowed the problem to dictate the size of the kernel. However, kernels as large as Lahart's are impractical for spatial processing. Smaller restoration kernels, defined for efficient processing, can yield excellent results. Second, the problem is defined so as to account for sampling and display—important considerations that Lahart and others did not address. Third, the restoration kernel is derived using a frequency-domain formulation. Frequency-domain analysis of sampling and reconstruction is straightforward. This approach is easily understood and in keeping with the traditional Wiener filter derivation.

References

- [1] Stephen E. Reichenbach, John C. Burton, and Keith W. Miller. A comparison of algorithms for computing the two-dimensional discrete Hartley transform. *Journal of the Optical Society of America A*, 6(6):818–823, 1989.
- [2] R. W. Hamming. *Digital Filters*. Prentice-Hall, Englewood Cliffs, NJ, second edition, 1983.
- [3] Roger J. Arguello, Harvey R. Sellner, and John A. Stuller. Transfer function compensation of sampled imagery. *IEEE Transactions on Computers*, 21(7):812–818, 1972.
- [4] Terry E. Riemer and Clare D. McGillem. Constrained optimization of image restoration filters. *Applied Optics*, 12(9):2027–2029, 1973.
- [5] Terry E. Riemer and Clare D. McGillem. Optimum constrained image restoration filters. *IEEE Transactions on Aerospace and Electronic Systems*, 13(2):136–146, 1977.
- [6] Harvey A. Smith. Improvement of the resolution of a linear scanning device. *SIAM Journal of Applied Mathematics*, 14(1):23–41, 1966.
- [7] R. W. Schutten and G. F. Vermeij. The approximation of image blur restoration filters by finite impulse responses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(2):176–180, 1980.
- [8] B. Roy Frieden. Image restoration by discrete convolution of minimal length. *Journal of the Optical Society of America*, 64(5):682–686, 1974.
- [9] John A. Stuller. An algebraic approach to image restoration filter design. *Computer Graphics and Image Processing*, 1:107–122, 1972.
- [10] Nim-Yau Chu and Clare D. McGillem. Image restoration filters based on a 1-0 weighting over the domain of support of the PSF. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 27(5):457–464, 1979.
- [11] Bahaa E. A. Saleh. Trade off between resolution and noise in restoration by superposition of images. *Applied Optics*, 13(8):1833–1838, 1974.
- [12] G. Backus and F. Gilbert. Uniqueness in the inversion of inaccurate gross earth data. *Philosophical Transactions of the Royal Society of London A*, 266:123–192, 1970.
- [13] Martin J. Lahart. Local image restoration by a least-squares method. *Journal of the Optical Society of America*, 69(10):1333–1339, 1979.

Chapter 6

Optimal Small-Kernel Restoration

The Wiener filter is probably the best-known and most-widely-used image restoration technique. Given a few assumptions and some knowledge of the system, the Wiener filter minimizes the expected mean-square restoration error (MSRE). While MSRE is by no means a perfect metric for restoration quality, it is a useful measure that lends itself to mathematical analysis and yields an implementable, optimal filter. As noted in Chapter 4, the Wiener filter can be criticized on several points, but no alternative technique has gained wider acceptance. In many applications, for example those requiring television-rate processing (30 images per second), the most serious drawback of the Wiener filter is its high computational cost. Although small spatial kernels, such as those described in Chapter 5, do not have as much capability or flexibility as the Wiener filter, they can be applied with much less computation. This chapter describes the design of small, Wiener-like restoration kernels that, subject to explicit spatial constraints, minimize MSRE.

Because the mathematics of this chapter is involved, Section 6.1 simplifies the problem by considering a one-dimensional, discrete system with no sampling or display. Section 6.2 addresses the one-dimensional, end-to-end restoration problem based on a more comprehensive model that includes sampling and display. Section 6.3 deals with two-dimensional restoration.

6.1 One-Dimensional Discrete Restoration

6.1.1 Formulation

The one-dimensional, discrete image formation and spatial restoration process is illustrated in Figure 6.1. The discrete image is assumed to have been formed by the convolution of the PSF and scene and the addition of noise. The objective of restoration is to produce a more accurate image of the scene by correcting for the effect of the PSF. Noise interferes with the ability of the restoration filter to accomplish this goal.

Mathematically, the process in Figure 6.1 can be described equivalently by equations in either the spatial or frequency domain. The digital image is

$$p[n] = \frac{1}{N} \sum_{n'} s[n'] h[n - n'] + e[n] \quad (6.1)$$

and the spectrum of the image is

$$\hat{p}[\nu] = \hat{s}[\nu] \hat{h}[\nu] + \hat{e}[\nu]. \quad (6.2)$$

The restored image is

$$\begin{aligned} r[n] &= \frac{1}{N} \sum_{n'} p[n'] f[n - n'] \\ &= \frac{1}{N} \sum_{n'} \left(\frac{1}{N} \sum_{n''} s[n''] h[n' - n''] + e[n'] \right) f[n - n'] \end{aligned} \quad (6.3)$$

and the spectrum of the restored image is

$$\hat{r}[\nu] = (\hat{s}[\nu] \hat{h}[\nu] + \hat{e}[\nu]) \hat{f}[\nu]. \quad (6.4)$$

The spatial kernel f (or the equivalent frequency-domain filter \hat{f}) is defined so that the restored image r is as much like the scene s as possible. Both

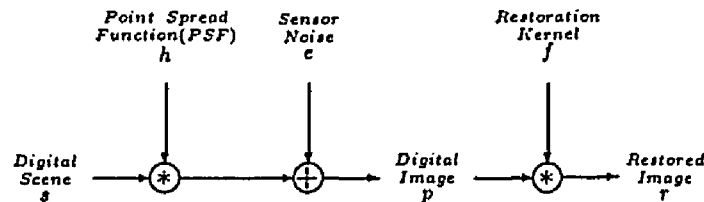


Figure 6.1: Discrete Image Formation and Spatial Restoration

the Wiener filter and the mean-square-optimal small kernel use the MSRE S^2 as the operative measure of restoration quality. Parseval's Equation equates MSRE and the energy of the error in the frequency domain:

$$\begin{aligned} S^2 &= E \left\{ \frac{1}{N} \sum_n |s[n] - r[n]|^2 \right\} \\ &= E \left\{ \sum_\nu |\hat{s}[\nu] - \hat{r}[\nu]|^2 \right\}. \end{aligned} \quad (6.5)$$

where E is the expectation operator.

The derivation assumes that the scene and noise processes are uncorrelated and that the autocorrelation functions (or equivalently the power spectra) of the scene and noise processes are known. These assumptions are motivated by mathematical convenience and may not be realistic in some applications. However, even if the assumptions do not hold, the Wiener filter and mean-square-optimal small-kernel restoration may still yield good results. Expressed as frequency domain equations, the assumptions are

$$E \{ |\hat{s}[\nu]|^2 \} = \Phi_s[\nu] \quad (6.6)$$

$$E \{ \hat{s}[\nu] \hat{e}^*[\nu] \} = 0 \quad (6.7)$$

$$E \{ \hat{s}^*[\nu] \hat{e}[\nu] \} = 0 \quad (6.8)$$

$$E \{ |\hat{e}[\nu]|^2 \} = \Phi_e[\nu]. \quad (6.9)$$

The use of the expectation operator does not necessarily mean that both the scene and noise are random processes. For example, these conditions would be satisfied by a deterministic scene with known power-spectrum corrupted by zero-mean, white noise with known energy. In that case, the expectation operator would not be needed for the scene (in Equation 6.6). Nevertheless, for generality, the expectation operator is used throughout. (As described in Chapter 2 the power-spectra of the scene and noise are real and symmetric.)

Using Equations 6.6–6.9, the expression for the expected MSRE can be written in a form that is suitable for minimization:

$$\begin{aligned} S^2 &= E \left\{ \sum_\nu |\hat{s}[\nu] - \hat{r}[\nu]|^2 \right\} \\ &= E \left\{ \sum_\nu (\hat{s}[\nu] - \hat{r}[\nu]) (\hat{s}^*[\nu] - \hat{r}^*[\nu]) \right\} \\ &= E \left\{ \sum_\nu (|\hat{s}[\nu]|^2 - \hat{s}[\nu] \hat{r}^*[\nu] - \hat{s}^*[\nu] \hat{r}[\nu] + |\hat{r}[\nu]|^2) \right\} \\ &= \sum_\nu (E \{ |\hat{s}[\nu]|^2 \} - E \{ \hat{s}[\nu] \hat{r}^*[\nu] \} - E \{ \hat{s}^*[\nu] \hat{r}[\nu] \} \\ &\quad + E \{ |\hat{r}[\nu]|^2 \}). \end{aligned} \quad (6.10)$$

The four terms of this expression can be rewritten as

$$E \{ |\hat{s}[\nu]|^2 \} = \Phi_s[\nu] \quad (6.11)$$

$$\begin{aligned} E \{ \hat{s}[\nu] \hat{r}^*[\nu] \} &= E \{ \hat{s}[\nu] (\hat{s}^*[\nu] \hat{h}^*[\nu] + \hat{e}^*[\nu]) \hat{f}^*[\nu] \} \\ &= E \{ |\hat{s}[\nu]|^2 \} \hat{h}^*[\nu] \hat{f}^*[\nu] + E \{ \hat{s}[\nu] \hat{e}^*[\nu] \} \hat{f}^*[\nu] \\ &= \Phi_s[\nu] \hat{h}^*[\nu] \hat{f}^*[\nu] \end{aligned} \quad (6.12)$$

$$\begin{aligned} E \{ \hat{s}^*[\nu] \hat{r}[\nu] \} &= E \{ \hat{s}^*[\nu] (\hat{s}[\nu] \hat{h}[\nu] + \hat{e}[\nu]) \hat{f}[\nu] \} \\ &= E \{ |\hat{s}[\nu]|^2 \} \hat{h}[\nu] \hat{f}[\nu] + E \{ \hat{s}^*[\nu] \hat{e}[\nu] \} \hat{f}[\nu] \\ &= \Phi_s[\nu] \hat{h}[\nu] \hat{f}[\nu] \end{aligned} \quad (6.13)$$

$$\begin{aligned} E \{ |\hat{r}[\nu]|^2 \} &= E \{ (\hat{s}[\nu] \hat{h}[\nu] + \hat{e}[\nu]) \hat{f}[\nu] (\hat{s}^*[\nu] \hat{h}^*[\nu] + \hat{e}^*[\nu]) \hat{f}^*[\nu] \} \\ &= E \{ |\hat{s}[\nu]|^2 \} |\hat{h}[\nu]|^2 |\hat{f}[\nu]|^2 + E \{ \hat{s}[\nu] \hat{e}^*[\nu] \} \hat{h}[\nu] |\hat{f}[\nu]|^2 \\ &\quad + E \{ \hat{s}^*[\nu] \hat{e}[\nu] \} \hat{h}^*[\nu] |\hat{f}[\nu]|^2 + E \{ |\hat{e}[\nu]|^2 \} |\hat{f}[\nu]|^2 \\ &= \Phi_s[\nu] |\hat{h}[\nu]|^2 |\hat{f}[\nu]|^2 + \Phi_e[\nu] |\hat{f}[\nu]|^2 \\ &= (\Phi_s[\nu] |\hat{h}[\nu]|^2 + \Phi_e[\nu]) |\hat{f}[\nu]|^2. \end{aligned} \quad (6.14)$$

Substituting these four expressions into Equation 6.10 yields

$$\begin{aligned} S^2 &= \sum_{\nu} (\Phi_s[\nu] - \Phi_s[\nu] \hat{h}^*[\nu] \hat{f}^*[\nu] - \Phi_s[\nu] \hat{h}[\nu] \hat{f}[\nu] \\ &\quad + (\Phi_s[\nu] |\hat{h}[\nu]|^2 + \Phi_e[\nu]) |\hat{f}[\nu]|^2). \end{aligned} \quad (6.15)$$

This expression for MSRE is a quadratic function of the filter values $\hat{f}[\nu]$ and it is convenient to write the mean-square error as

$$S^2 = \sum_{\nu} (\hat{c}[\nu] - \hat{b}[\nu] \hat{f}^*[\nu] - \hat{b}^*[\nu] \hat{f}[\nu] + \hat{a}[\nu] |\hat{f}[\nu]|^2) \quad (6.16)$$

where

$$\hat{a}[\nu] = \Phi_s[\nu] |\hat{h}[\nu]|^2 + \Phi_e[\nu] \quad (6.17)$$

$$\hat{b}[\nu] = \Phi_s[\nu] \hat{h}^*[\nu] \quad (6.18)$$

$$\hat{c}[\nu] = \Phi_s[\nu]. \quad (6.19)$$

Except for the restoration filter \hat{f} , all of the components of Equation 6.16 are assumed to be known: Φ_s is the scene power-spectrum, Φ_e is the noise power-spectrum, and \hat{h} is the system transfer function.

The expression for MSRE in Equation 6.16 is in a form that is suitable for minimization. Section 6.1.2 derives the discrete Wiener filter. Section 6.1.3 places spatial constraints on the kernel before deriving the mean-square-optimal kernel values.

6.1.2 Discrete Wiener Filter

Though the expected MSRE, S^2 in Equation 6.16, is real, the values of the filter \hat{f} may be complex. Therefore, minimization of S^2 with respect to the filter values requires consideration of both phase and amplitude. (The same approach could be followed using the real and imaginary parts instead.) Let \hat{f}_ρ denote the magnitude of the filter and \hat{f}_ϕ denote the phase:

$$\hat{f}[\nu] = \hat{f}_\rho[\nu] \exp(i\hat{f}_\phi[\nu]). \quad (6.20)$$

Substituting in the expression for MSRE (Equation 6.16) yields

$$\begin{aligned} S^2 = \sum_{\nu'} & (\hat{c}[\nu'] - \hat{b}[\nu'] \hat{f}_\rho[\nu'] \exp(-i\hat{f}_\phi[\nu']) \\ & - \hat{b}^*[\nu'] \hat{f}_\rho[\nu'] \exp(i\hat{f}_\phi[\nu']) + \hat{a}[\nu'] \hat{f}_\rho[\nu']^2). \end{aligned} \quad (6.21)$$

The optimal value at each frequency is determined independently. Consider the phase component first:

$$\begin{aligned} \frac{\partial S^2}{\partial \hat{f}_\phi[\nu]} &= i\hat{b}[\nu] \hat{f}_\rho[\nu] \exp(-i\hat{f}_\phi[\nu]) - i\hat{b}^*[\nu] \hat{f}_\rho[\nu] \exp(i\hat{f}_\phi[\nu]) \\ &= i\hat{f}_\rho[\nu] (\hat{b}[\nu] \exp(-i\hat{f}_\phi[\nu]) - \hat{b}^*[\nu] \exp(i\hat{f}_\phi[\nu])) \\ &= 0. \end{aligned} \quad (6.22)$$

Minimization with respect to phase requires that either $\hat{f}_\rho[\nu] = 0$ or

$$\hat{b}[\nu] \exp(-i\hat{f}_\phi[\nu]) = \hat{b}^*[\nu] \exp(i\hat{f}_\phi[\nu]) \quad (6.23)$$

Equation 6.23 is used in this form in the mathematics below, but it can be simplified further to reveal that the phase of the optimal filter must be the negative of the phase of the system transfer function:

$$\hat{f}_\phi[\nu] = -\hat{h}_\phi[\nu] \quad (6.24)$$

In words, the filter should undo any phase shift introduced by the system. Now consider the amplitude of the filter:

$$\begin{aligned} \frac{\partial S^2}{\partial \hat{f}_\rho[\nu]} &= -\hat{b}[\nu] \exp(-i\hat{f}_\phi[\nu]) - \hat{b}^*[\nu] \exp(i\hat{f}_\phi[\nu]) + 2\hat{a}[\nu] \hat{f}_\rho[\nu] \\ &= 0. \end{aligned} \quad (6.25)$$

At any frequency ν , where $\hat{a}[\nu] = 0$, the Wiener filter is not defined. However, it is possible to separate the question of mathematical existence from the filter's practical application. The condition $\hat{a}[\nu] = 0$ occurs only when

the system is not expected to generate a signal at that frequency. If there is no energy at that frequency, however, there should be no restoration and therefore, in practice, the Wiener filter value at that frequency is defined as zero. Otherwise, minimization with respect to amplitude requires that

$$\hat{f}_\rho[\nu] = \frac{(\hat{b}[\nu] \exp(-i\hat{f}_\phi[\nu]) + \hat{b}^*[\nu] \exp(i\hat{f}_\phi[\nu]))}{2\hat{a}[\nu]}. \quad (6.26)$$

Together, the expressions for amplitude and phase define the Wiener filter:

$$\begin{aligned} \hat{f}[\nu] &= \hat{f}_\rho[\nu] \exp(i\hat{f}_\phi[\nu]) \\ &= \frac{(\hat{b}[\nu] \exp(-i\hat{f}_\phi[\nu]) + \hat{b}^*[\nu] \exp(i\hat{f}_\phi[\nu])) \exp(i\hat{f}_\phi[\nu])}{2\hat{a}[\nu]} \\ &= \frac{2\hat{b}[\nu] \exp(-i\hat{f}_\phi[\nu]) \exp(i\hat{f}_\phi[\nu])}{2\hat{a}[\nu]} \\ &= \frac{\hat{b}[\nu]}{\hat{a}[\nu]} \\ &= \frac{\Phi_s[\nu] \hat{h}^*[\nu]}{\Phi_s[\nu] |\hat{h}[\nu]|^2 + \Phi_e[\nu]}. \end{aligned} \quad (6.27)$$

This completes the derivation of the traditional, discrete Wiener filter \hat{f} . Given the model in Figure 6.1 and knowledge of the power-spectra of the scene and noise and the system transfer function, this is the filter that minimizes expected mean-square restoration error.

The model in Figure 6.1 depicts restoration using spatial convolution. For spatial restoration, the Wiener filter \hat{f} must be transformed to the spatial domain using the inverse Fourier transform. However, the resulting spatial kernel f is as large as the image p , so the computational cost of the convolution is prohibitive. Therefore, the Wiener filter is almost always applied in the frequency domain by taking the transform of the digital image \hat{p} , multiplying by the Wiener filter \hat{f} , and then taking the inverse transform of the product $\hat{p}\hat{f}$. This procedure is depicted in Figure 6.2.

6.1.3 Discrete Spatially-Constrained Kernel

The computational cost of convolution is proportional to the number of nonzero elements in the kernel. Because of this, spatial kernels are not practical unless they have only a few nonzero elements. If the spatial kernel is restricted to a few non-zero values, spatial convolution requires significantly less computation than frequency-domain restoration. The algorithm

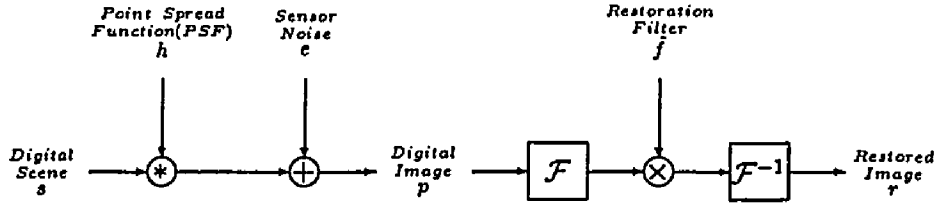


Figure 6.2: Discrete Image Formation and Frequency Restoration

presented in this section constrains the kernel to a few specific locations. The (unconstrained) Wiener filter is mean-square-optimal, so the spatially-constrained kernel cannot do better. However, it may perform nearly as well. In practice, the most significant elements of restoration kernels are located near the kernel center and elements far from the center are small. Therefore, it is reasonable to expect that a small spatial kernel with a few well-chosen values near the kernel center can be nearly as effective as the Wiener filter.

In the derivation of the previous section, the Wiener filter is defined in the frequency domain by the equations

$$\hat{a}[\nu] \hat{f}[\nu] = \hat{b}[\nu] \quad \nu = 0, 1, \dots, N-1 \quad (6.28)$$

where

$$\hat{a}[\nu] = \Phi_s[\nu] \left| \hat{h}[\nu] \right|^2 + \Phi_e[\nu] \quad (6.29)$$

$$\hat{b}[\nu] = \Phi_s[\nu] \hat{h}^*[\nu]. \quad (6.30)$$

The spatial domain equivalent of this frequency domain product is the convolution equation

$$a * f = b \quad (6.31)$$

where

$$a[n] = \sum_{\nu} \hat{a}[\nu] W_N^{\nu n} \quad (6.32)$$

$$b[n] = \sum_{\nu} \hat{b}[\nu] W_N^{\nu n}. \quad (6.33)$$

This convolution can be equivalently expressed as a linear system of N equations in N variables (the spatial kernel values) as

$$\mathbf{A} \mathbf{f} = \mathbf{b} \quad (6.34)$$

where the $N \times N$ coefficient matrix A is

$$A[n_1, n_2] = \frac{1}{N} a[n_1 - n_2], \quad (6.35)$$

the $N \times 1$ result matrix b is defined as in Equation 6.33, and f is the $N \times 1$ matrix of kernel values to be determined. (This system of equations is conditioned on periodic images and circular convolution.)

In Equation 6.34, there are as many equations as image pixels. However, if the size of the kernel is constrained, the system of independent equations can only be as large as the number of nonzero elements in the kernel. The optimal spatially-constrained kernel is specified by the system of linear equations whose solution will minimize MSRE subject to the constraints.

The spatial constraints on the kernel are expressed as a nonempty set of spatial locations, $C \subseteq \{0 \dots N - 1\}$, for which the kernel can be nonzero. The elements not in the constraint set must be zero:

$$f[n] = 0 \quad \text{if } (n \bmod N) \notin C. \quad (6.36)$$

If all of the elements in the kernel are allowed to be nonzero (i.e., $C = \{0 \dots N - 1\}$), then the optimal kernel is the inverse transform of the Wiener filter (i.e., the solution of Equation 6.34). (This observation is proven in Section 6.1.4.)

The expression for expected MSRE in Equation 6.16 is defined in terms of frequency components:

$$S^2 = \sum_{\nu} \left(\hat{c}[\nu] - \hat{b}[\nu] \hat{f}^*[\nu] - \hat{b}^*[\nu] \hat{f}[\nu] + \hat{a}[\nu] |\hat{f}[\nu]|^2 \right). \quad (6.37)$$

Before this expression can be minimized with respect to the kernel elements in the constraint set, it must be expressed in terms of those values. The transfer function of the kernel is

$$\hat{f}[\nu] = \frac{1}{N} \sum_{n \in C} f[n] W_N^{-\nu n}. \quad (6.38)$$

Substituting this expression into Equation 6.37, yields the expected MSRE in terms of the (unrestricted) kernel values:

$$S^2 = \sum_{\nu} \left(\hat{c}[\nu] - \hat{b}[\nu] \left(\frac{1}{N} \sum_{n' \in C} f^*[n'] W_N^{\nu n'} \right) - \hat{b}^*[\nu] \left(\frac{1}{N} \sum_{n' \in C} f[n'] W_N^{-\nu n'} \right) + \hat{a}[\nu] \left| \frac{1}{N} \sum_{n' \in C} f[n'] W_N^{-\nu n'} \right|^2 \right)$$

$$\begin{aligned}
&= \sum_{\nu} \hat{c}[\nu] - \frac{1}{N} \sum_{n' \in \mathcal{C}} f^*[n'] \sum_{\nu} \hat{b}[\nu] W_N^{\nu n'} - \frac{1}{N} \sum_{n' \in \mathcal{C}} f[n'] \sum_{\nu} \hat{b}[\nu] W_N^{-\nu n'} \\
&\quad + \frac{1}{N^2} \sum_{n' \in \mathcal{C}} \sum_{n'' \in \mathcal{C}} f^*[n'] f[n''] \sum_{\nu} \hat{a}[\nu] W_N^{\nu(n'-n'')} \\
&= \sum_{\nu} \hat{c}[\nu] - \frac{1}{N} \sum_{n' \in \mathcal{C}} f^*[n'] b[n'] \\
&\quad - \frac{1}{N} \sum_{n' \in \mathcal{C}} f[n'] b^*[n'] + \frac{1}{N^2} \sum_{n' \in \mathcal{C}} \sum_{n'' \in \mathcal{C}} f^*[n'] f[n''] a[n' - n'']. \quad (6.39)
\end{aligned}$$

For real systems (i.e., the scene s , system h , and noise e are real-valued), the optimal kernel f and matrices a and b are real-valued. (A proof of this is given in Section 6.1.4.) In the interest of generality, however, Equation 6.39 allows for complex values. Let f_r and f_i be the real and imaginary parts of the kernel:

$$\begin{aligned}
S^2 &= \sum_{\nu} \hat{c}[\nu] - \frac{1}{N} \sum_{n' \in \mathcal{C}} f^*[n'] b[n'] - \frac{1}{N} \sum_{n' \in \mathcal{C}} f[n'] b^*[n'] \\
&\quad + \frac{1}{N^2} \sum_{n' \in \mathcal{C}} \sum_{n'' \in \mathcal{C}} f^*[n'] f[n''] a[n' - n''] \\
&= \sum_{\nu} \hat{c}[\nu] - \frac{1}{N} \sum_{n' \in \mathcal{C}} (f_r[n'] - i f_i[n']) b[n'] \\
&\quad - \frac{1}{N} \sum_{n' \in \mathcal{C}} (f_r[n'] + i f_i[n']) b^*[n'] \\
&\quad + \frac{1}{N^2} \sum_{n' \in \mathcal{C}} \sum_{n'' \in \mathcal{C}} (f_r[n'] - i f_i[n']) (f_r[n''] + i f_i[n'']) a[n' - n'']. \quad (6.40)
\end{aligned}$$

The array a is Hermitian:

$$\begin{aligned}
a[n' - n''] &= \sum_{\nu} \left(\Phi_s[\nu] |\hat{h}[\nu]|^2 + \Phi_e[\nu] \right) W_N^{\nu(n'-n'')} \\
&= \sum_{\nu} \left(\Phi_s[\nu] |\hat{h}[\nu]|^2 + \Phi_e[\nu] \right) W_N^{-\nu(n''-n')} \\
&= a^*[n'' - n'], \quad (6.41)
\end{aligned}$$

a property that is used below in determining the optimal spatial filter.

Minimization with respect to a kernel element $f[n]$ proceeds by differentiating S^2 with respect to the real and imaginary components. Differentiating with respect to the imaginary component of a kernel element yields

$$\frac{\partial S^2}{\partial f_i[n]} = \frac{i}{N} b[n] - \frac{i}{N} b^*[n] - \frac{i}{N^2} \sum_{n' \in \mathcal{C}} f_r[n'] a[n - n']$$

$$\begin{aligned}
& + \frac{1}{N^2} \sum_{n' \in C} f_i[n'] a[n - n'] + \frac{i}{N^2} \sum_{n' \in C} f_r[n'] a[n' - n] \\
& + \frac{1}{N^2} \sum_{n' \in C} f_i[n'] a[n' - n] \\
= & \frac{i}{N} b[n] - \frac{i}{N} b^*[n] - \frac{i}{N^2} \sum_{n' \in C} f_r[n'] a[n - n'] \\
& + \frac{1}{N^2} \sum_{n' \in C} f_i[n'] a[n - n'] + \frac{i}{N^2} \sum_{n' \in C} f_r[n'] a^*[n - n'] \\
& + \frac{1}{N^2} \sum_{n' \in C} f_i[n'] a^*[n - n'] \\
= & -\frac{2}{N} \Im \{b[n]\} + \frac{2}{N^2} \sum_{n' \in C} \Im \{f[n'] a[n - n']\} \\
= & 0. \tag{6.42}
\end{aligned}$$

Similarly, differentiating with respect to the real component of a kernel element yields

$$\begin{aligned}
\frac{\partial S^2}{\partial f_r[n]} & = -\frac{1}{N} b[n] - \frac{1}{N} b^*[n] + \frac{1}{N^2} \sum_{n' \in C} f_r[n'] a[n - n'] \\
& + \frac{i}{N^2} \sum_{n' \in C} f_i[n'] a[n - n'] + \frac{1}{N^2} \sum_{n' \in C} f_r[n'] a[n' - n] \\
& - \frac{i}{N^2} \sum_{n' \in C} f_i[n'] a[n' - n] \\
= & -\frac{1}{N} b[n] - \frac{1}{N} b^*[n] + \frac{1}{N^2} \sum_{n' \in C} f_r[n'] a[n - n'] \\
& + \frac{i}{N^2} \sum_{n' \in C} f_i[n'] a[n - n'] + \frac{1}{N^2} \sum_{n' \in C} f_r[n'] a^*[n - n'] \\
& - \frac{i}{N^2} \sum_{n' \in C} f_i[n'] a^*[n - n'] \\
= & -\frac{2}{N} \Re \{b[n]\} + \frac{2}{N^2} \sum_{n' \in C} \Re \{f[n'] a[n - n']\} \\
= & 0. \tag{6.43}
\end{aligned}$$

The real and imaginary constraints are combined in the equation

$$\frac{1}{N} \sum_{n' \in C} f[n'] a[n - n'] = b[n] \quad n \in C. \tag{6.44}$$

This is an equation with a number of unknowns equal to the number of unconstrained kernel elements. There are $|C|$ equations in $|C|$ unknowns

(the $|C|$ kernel values). This system of equations can be written as the matrix equation

$$\mathbf{A}_C \mathbf{f}_C = \mathbf{b}_C \quad (6.45)$$

where \mathbf{A}_C is the $|C| \times |C|$ coefficient matrix, \mathbf{f}_C is the $|C| \times 1$ vector of kernel values, and \mathbf{b}_C is the $|C| \times 1$ result vector.

The matrices of Equation 6.45 are submatrices of the matrices of Equation 6.34 for the Wiener filter. The vector \mathbf{f}_C has values for the subset of kernel elements C that are not restricted to have zero value. The vector \mathbf{b}_C contains only the elements of \mathbf{b} that are named in the constraint set C . Similarly, \mathbf{A}_C is a principal submatrix[1] of the coefficient matrix \mathbf{A} —consisting only of the rows and columns of \mathbf{A} named in the constraint set C .

Equation 6.45 is the key result of this section. Its solution is the mean-square-optimal kernel that conforms to the spatial constraints. The following algorithm summarizes the process of calculating the optimal spatially-constrained kernel.

Algorithm 6.1 *Mean-Square-Optimal, Spatially-Constrained, Kernel for One-Dimensional, Discrete Restoration*

Input:

- N —Image dimension.
- $\Phi_s [N]$ —Power spectrum of the scene.
- $\Phi_e [N]$ —Power spectrum of the noise.
- $\hat{h} [N]$ —Optical transfer function of the system.
- $|C|$ —Number of kernel elements.
- C —Constraint set (an ordered set of the locations of the kernel elements).

Output:

- $f [N]$ —Mean-square-optimal spatially-constrained kernel.

Declarations:

- $\hat{a} [N]$ —Wiener filter denominator.
- $\hat{b} [N]$ —Wiener filter numerator.
- $a [N]$ —Spatial transform of Wiener filter denominator.
- $b [N]$ —Spatial transform of Wiener filter numerator.
- $A_C [|C|, |C|]$ —Coefficient matrix.
- $f_C [|C|]$ —Input matrix.
- $b_C [|C|]$ —Output matrix.
- ν —Frequency index.
- n_1, n_2 —Spatial indices.

fourier_transform⁻¹()—Inverse Fourier transform subroutine.
 linear_solver()—Subroutine for solving a system of linear equations.

Execution:

```

begin
  for  $\nu \leftarrow 0$  to  $N - 1$ 
    begin
       $\hat{a}[\nu] \leftarrow \Phi_s[\nu] |\hat{h}[\nu]|^2 + \Phi_t[\nu]$ 
       $\hat{b}[\nu] \leftarrow \Phi_s[\nu] \hat{h}^*[\nu]$ 
    end
     $a \leftarrow \text{fourier\_transform}^{-1}(\hat{a})$ 
     $b \leftarrow \text{fourier\_transform}^{-1}(\hat{b})$ 
    foreach  $n_1$  s.t.  $n_1 \in C$ 
      begin
         $b_C[\text{ord}(n_1 \text{ in } C)] \leftarrow b[n_1]$ 
        foreach  $n_2$  s.t.  $n_2 \in C$ 
           $A_C[\text{ord}(n_1 \text{ in } C), \text{ord}(n_2 \text{ in } C)] \leftarrow \frac{1}{N} a[n_1 - n_2]$ 
        end
         $f_C \leftarrow \text{linear\_solver}(A_C, b_C)$ 
        for  $n_1 \leftarrow 0$  to  $N - 1$ 
          if  $n_1 \in C$ 
             $f[n_1] \leftarrow f_C[\text{ord}(n_1 \text{ in } C)]$ 
          else
             $f[n_1] \leftarrow 0$ 
        end
      end
    end
  end

```

6.1.4 Theorems

Several important observations can be made. First, if none of the values of the kernel are restricted (i.e., the kernel can contain all of the spatial elements), the optimal kernel is the inverse Fourier transform of the Wiener filter. Second, if the Wiener filter exists, then for any spatial constraints there is a unique optimal constrained kernel. Third, if the system is real-valued, the optimal kernel is real-valued. These three observations are expressed in the following theorems.

Theorem 6.1 *If none of the kernel values are restricted, then the optimal, spatially-constrained kernel is the inverse Fourier transform of the Wiener filter.*

Proof: If $C = \{0 \dots N - 1\}$, all of the rows and columns of A are retained in the submatrix A_C , so $A = A_C$. Likewise, all of the rows of b are retained and $b = b_C$. Therefore, both vectors f and f_C are defined by the same set of equations.

□

When the matrix A_C is invertible, Algorithm 6.1 will generate the optimal kernel. If the matrix A_C is not invertible, the algorithm will fail. When is A_C invertible? In general, this question cannot be answered *a priori*, but for the important (and typical) case that the Wiener filter is defined at all frequencies, there is an optimal kernel for any spatial constraints. If the Wiener filter is not defined at some frequencies, then for a particular spatial constraint the optimal kernel may or may not be defined.

Theorem 6.2 *If the Wiener filter is defined at all frequencies (i.e., for all ν , $\hat{a}[\nu] \neq 0$), then for any spatial constraint set C there exists a unique optimal kernel.*

Proof: The eigenvalues of the circulant matrix A are the values of its transform \hat{a} , which are non-negative reals (Equation 6.17). Therefore, A is positive semi-definite. If the Wiener filter exists at all frequencies then none of the eigenvalues are zero and A is positive definite. Matrix A is positive definite if and only if $x^*Ax > 0$ for all nonzero vectors x . Let x be any nonzero vector whose elements that are not in the constraint set are zero. Let x_C be the submatrix of x containing only the elements in the constraint set. (The matrix A_C is the principal submatrix of the matrix A whose rows and columns are named in C .) Then $x_C^*A_Cx_C = x^*Ax > 0$. Because $x_C \neq 0$ is arbitrary, A_C is positive definite and hence invertible. (This is an application of the theorem that any principal submatrix of a positive definite matrix is positive definite[1].) Under these conditions, the optimal, constrained kernel exists and is defined as

$$f_C = A_C^{-1}b_C. \quad (6.46)$$

□

Note that the presence of noise at all frequencies (i.e., $\Phi_e[\nu] > 0$ for all ν) is sufficient for the Wiener filter to be defined at all frequencies and hence is a sufficient condition for the existence of a unique optimal filter for any spatial constraint.

Inversion is an ill-conditioned problem. The condition number of a matrix bounds the relative error in the inverse in terms of the relative error in the data. The smaller the condition number, the less ill-conditioned the problem.

It is appropriate to consider how the condition number of the coefficient matrix \mathbf{A}_C compares to the condition number of matrix A —that is, how does constraining the size of the kernel affect the conditioning of the defining equations?

Corollary 6.1 *The coefficient matrix of the system of equations defining the constrained kernel is at least as well-conditioned as the coefficient matrix of the system of equations defining the Wiener filter.*

Proof: The condition number (relative to the spectral norm) of the Wiener filter coefficient matrix is the ratio of the largest eigenvalue to the smallest eigenvalue:

$$\kappa_A = \frac{\max \langle \lambda(A) \rangle}{\min \langle \lambda(A) \rangle} \quad (6.47)$$

where λ is an eigenvalue. The coefficient matrix \mathbf{A}_C is a principal submatrix of the positive-definite, Hermitian matrix A . Therefore, from the inclusion principle (or interlacing eigenvalues)[1], the largest eigenvalue of \mathbf{A}_C is bounded above by the largest eigenvalue of A :

$$\max \langle \lambda(\mathbf{A}_C) \rangle \leq \max \langle \lambda(A) \rangle. \quad (6.48)$$

and the smallest eigenvalue of \mathbf{A}_C is bounded below by the smallest eigenvalue of A :

$$\min \langle \lambda(\mathbf{A}_C) \rangle \geq \min \langle \lambda(A) \rangle. \quad (6.49)$$

The matrix \mathbf{A}_C is normal, so its condition number is also the ratio of its largest to smallest eigenvalue:

$$\kappa_{\mathbf{A}_C} = \frac{\max \langle \lambda(\mathbf{A}_C) \rangle}{\min \langle \lambda(\mathbf{A}_C) \rangle} \quad (6.50)$$

Therefore,

$$\begin{aligned} \kappa_{\mathbf{A}_C} &= \frac{\max \langle \lambda(\mathbf{A}_C) \rangle}{\min \langle \lambda(\mathbf{A}_C) \rangle} \\ &\leq \frac{\max \langle \lambda(A) \rangle}{\min \langle \lambda(A) \rangle} \\ &= \kappa_A \end{aligned} \quad (6.51)$$

□

Though the derivation of the optimal kernel allows for complex-valued systems, imaging systems are real-valued. For real-valued systems, the following theorem can be established.

Theorem 6.3 *For real systems, if a unique optimal kernel exists, then it is real.*

Proof: Recall the definitions of a and b :

$$a[n] = \sum_{\nu} \left(\Phi_s[\nu] |\hat{h}[\nu]|^2 + \Phi_c[\nu] \right) W_N^{\nu n} \quad (6.52)$$

$$b[n] = \sum_{\nu} \Phi_s[\nu] \hat{h}^*[\nu] W_N^{\nu n}. \quad (6.53)$$

If the scene, PSF, and noise are real, their power spectra are real and symmetric. The pointwise product or sum of real, symmetric arrays is real and symmetric, so the product of the power spectra of the scene and PSF plus the power spectrum of the noise is real and symmetric. The inverse Fourier transform of a real and symmetric array is real and symmetric, so a is real and symmetric. If a is real, \mathbf{A} is also real. Any submatrix of a real array is real, so \mathbf{A}_C is real. If \mathbf{A}_C is real, so is its inverse \mathbf{A}_C^{-1} (if it exists).

If the scene is real, its power spectrum is symmetric and real. If the PSF is real, its transform is Hermitian (i.e., a symmetric real part and an anti-symmetric imaginary part). The conjugate of a Hermitian is Hermitian, so the conjugate of the transform of the PSF is Hermitian. The pointwise product of a symmetric function and a Hermitian function is Hermitian, so the product of the power spectrum of the scene and the conjugate of the transform of the PSF is Hermitian. The inverse transform of a Hermitian is real, so b is real. Therefore, \mathbf{b} and \mathbf{b}_C are also real.

The optimal kernel constrained by the constraint set C is defined by the linear system of equations $\mathbf{A}_C f = \mathbf{b}_C$. If a unique solution exists, \mathbf{A}_C can be inverted and $f_C = \mathbf{A}_C^{-1} \mathbf{b}_C$. If \mathbf{A}_C^{-1} and \mathbf{b}_C are real, then f_C is real.

□

6.2 End-to-End Restoration

6.2.1 Formulation

Sampling and display are important components of the end-to-end imaging process. The discrete imaging model used in the previous section simplifies the derivation of the restoration kernel by disregarding sampling and display. A more accurate, end-to-end model of the imaging process is pictured in Figure 6.3. The formulation of the Wiener filter based on this model is detailed in Huck et al.[2] and Fales et al.[3].

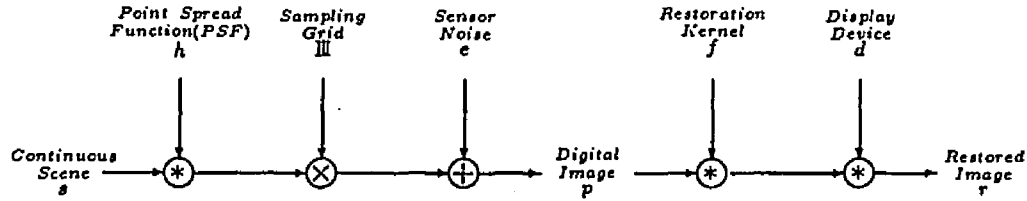


Figure 6.3: End-to-End Imaging Model and Spatial Restoration

The digital image is

$$p[n] = \int_{-\infty}^{\infty} s(x') h(n - x') dx' + e[n]. \quad (6.54)$$

As in the previous section, periodicity of the scene is assumed. This assumption is dictated by computational considerations—periodicity of the scene allows the derivation to be restricted to discrete frequencies. The discrete spectrum \hat{s} of the periodic scene s is given by the Fourier coefficients

$$\hat{s}[\nu] = \frac{1}{N} \int_{-\infty}^{\infty} s(x) W_N^{-\nu x} dx. \quad \nu = 0, \pm 1, \pm 2, \dots \quad (6.55)$$

(Recall that the frequency index $[\nu]$ corresponds to the spatial frequency (ν/N) .) The transfer function \hat{h} of the system is the continuous Fourier transform of the PSF h :

$$\hat{h}(\nu) = \int_{-\infty}^{\infty} h(x) W^{-\nu x} dx. \quad -\infty < \nu < \infty \quad (6.56)$$

The acquisition transfer function \hat{h} is *not* discrete, but its value is only of interest at the component frequencies of the periodic scene. For these discrete frequency components, $\hat{h}[\nu] \equiv \hat{h}(\nu/N)$.

The frequency-domain equation corresponding to Equation 6.54 is then

$$\hat{p}[\nu] = \sum_{k=-\infty}^{\infty} \hat{s}[\nu - kN] \hat{h}[\nu - kN] + \hat{e}[\nu]. \quad (6.57)$$

The arrays \hat{p} and \hat{e} are the (periodic) discrete Fourier transforms of the arrays p and e respectively. In Equation 6.57, sampling is manifested as the folding of the sideband frequencies into the baseband.

After restoration, the continuous, displayed image is

$$r(x) = \sum_n \left(\frac{1}{N} \sum_{n'} p[n'] f[n - n'] \right) d(x - n) \quad (6.58)$$

The spectrum \hat{r} of the displayed image r is discrete and aperiodic:

$$\hat{r}[\nu] = \hat{p}[\nu] \hat{f}[\nu] \hat{d}[\nu] \quad \nu = 0, \pm 1, \pm 2, \dots \quad (6.59)$$

where the restoration filter transfer function \hat{f} is the DFT of the kernel f and the display transfer function \hat{d} is the continuous Fourier transform of the display PSF:

$$\hat{d}(v) = \int_{-\infty}^{\infty} d(x) W^{-vx} dx. \quad -\infty < v < \infty \quad (6.60)$$

The display transfer function \hat{d} is not discrete. However, just as for the acquisition transfer function, only its values at discrete frequencies are used to display the periodic result ($\hat{d}[\nu] \equiv \hat{d}(\nu/N)$).

As in the previous section, the expression for MSRE must be rewritten in terms of the unknown filter values \hat{f} . The expected MSRE is

$$\begin{aligned} S^2 &= E \left\{ \frac{1}{N} \int_0^N |s(x) - r(x)|^2 dx \right\} \\ &= E \left\{ \sum_{\nu=-\infty}^{\infty} |\hat{s}[\nu] - \hat{r}[\nu]|^2 \right\} \end{aligned} \quad (6.61)$$

In addition to the assumptions in Section 6.1 (Equations 6.6–6.9), the elements of the scene spectrum that will be aliased to a single frequency (the sidebands) in the sampled image are assumed to be uncorrelated:

$$E \{ \hat{s}[\nu] \hat{s}^*[\nu + kN] \} = \begin{cases} \Phi_s[\nu] & \text{if } k = 0 \\ 0 & \text{otherwise} \end{cases} \quad (6.62)$$

$$E \{ \hat{s}[\nu] \hat{e}^*[\nu] \} = 0 \quad (6.63)$$

$$E \{ \hat{s}^*[\nu] \hat{e}[\nu] \} = 0 \quad (6.64)$$

$$E \{ |\hat{e}[\nu]|^2 \} = \Phi_e[\nu]. \quad (6.65)$$

After a great deal of algebra, it can be shown that again

$$S^2 = \sum_{\nu=0}^{N-1} \left(\hat{c}[\nu] - \hat{b}[\nu] \hat{f}^*[\nu] - \hat{b}^*[\nu] \hat{f}[\nu] + \hat{a}[\nu] |\hat{f}[\nu]|^2 \right) \quad (6.66)$$

where now

$$\begin{aligned} \hat{a}[\nu] &= \left(\sum_{k=-\infty}^{\infty} \Phi_s[\nu - kN] |\hat{h}[\nu - kN]|^2 + \Phi_e[\nu] \right) \\ &\quad \times \left(\sum_{k=-\infty}^{\infty} |\hat{d}[\nu - kN]|^2 \right) \end{aligned} \quad (6.67)$$

$$\hat{b}[\nu] = \sum_{k=-\infty}^{\infty} \Phi_s[\nu - kN] \hat{h}^*[\nu - kN] \hat{d}^*[\nu - kN] \quad (6.68)$$

$$\hat{c}[\nu] = \sum_{k=-\infty}^{\infty} \Phi_s[\nu - kN]. \quad (6.69)$$

These equations are consistent with those of Fales et al.[3]. The expression for the MSRE for end-to-end restoration in Equation 6.66 is identical to the expression for MSRE for discrete restoration in Equation 6.16, except that the definitions of \hat{a} , \hat{b} , and \hat{c} are different.

6.2.2 End-to-End Wiener Filter

The derivation of the Wiener filter in Section 6.1.2 is not conditioned on the definitions of \hat{a} , \hat{b} , \hat{c} , so the derivation for the end-to-end Wiener restoration filter is identical. The result is again

$$\hat{f}[\nu] = \frac{\hat{b}[\nu]}{\hat{a}[\nu]} \quad (6.70)$$

where $\hat{a}[\nu]$ and $\hat{b}[\nu]$ are defined in Equations 6.67 and 6.68.

If there is no aliasing expected (i.e., $\Phi_s[\nu] \hat{h}[\nu] = 0$ for $|\nu| \geq N/2$) and the display is "ideal" (i.e., \hat{d} is the unit pulse), then the optimal, end-to-end filter is identical to the traditional formulation of the Wiener image restoration filter. Of course, in practice this is seldom if ever the case.

6.2.3 End-to-End, Spatially-Constrained Kernel

As in Section 6.1.3, the optimal kernel is calculated by solving the system of equations

$$\mathbf{A}_C \mathbf{f}_C = \mathbf{b}_C \quad (6.71)$$

where \mathbf{A}_C and \mathbf{b}_C are constructed from the arrays a and b which are the inverse discrete Fourier transforms of \hat{a} and \hat{b} defined in Equations 6.67 and 6.68.

Only a few changes in Algorithm 6.1 are needed to accommodate the end-to-end model. The scene spectrum \hat{s} defined in Equation 6.55 may have an arbitrarily large number of frequency components. In practice, the series must be finite, but the effect of any frequency truncation beyond two to four times the Nyquist limit is typically minimal. Most scenes have little

energy at high frequencies and the magnitude of the transfer function far beyond the Nyquist limit is effectively zero. Let N_s be an integer-multiple of N ($N_s = S_N N$ for some integer S_N) such that the scene spectrum has no components at or beyond $(\pm N_s/2)$. That is:

$$\hat{s}[\nu] = 0 \quad \text{if } |\nu| \geq N_s/2 \quad (6.72)$$

It is convenient to store the N_s coefficients of the scene spectrum in the same data structure as a DFT, storing the negative components ($-N_s/2 \leq \nu < 0$) in the top half of a N_s -element array. The corresponding values of the acquisition and display transfer functions can also be stored in this manner. Then, the calculation in the following algorithm of \hat{a} and \hat{b} folds the sideband components into a N -element array as defined in Equations 6.67 and 6.68.

Algorithm 6.2 *Mean-Square-Optimal, Spatially-Constrained, Kernel for One-Dimensional, End-to-End Restoration*

Input:

- N_s —Scene spectrum dimension.
- N —Image dimension.
- $\Phi_s [N_s]$ —Power spectrum of the scene.
- $\Phi_e [N]$ —Power spectrum of the noise.
- $\hat{h} [N_s]$ —Acquisition transfer function.
- $\hat{d} [N_s]$ —Display transfer function.
- $|C|$ —Number of kernel elements.
- C —Constraint set (an ordered set of the locations of the kernel elements).

Output:

- $f [N]$ —Mean-square-optimal spatially-constrained kernel.

Declarations:

- $\hat{a} [N]$ —Wiener filter denominator.
- $\hat{b} [N]$ —Wiener filter numerator.
- t_a, t_b, t_d —Temporary variables.
- $a [N]$ —Spatial transform of Wiener filter denominator.
- $b [N]$ —Spatial transform of Wiener filter numerator.
- $A_C [|C|, |C|]$ —Coefficient matrix.
- $f_C [|C|]$ —Input matrix.
- $b_C [|C|]$ —Output matrix.
- ν_1, ν_2 —Frequency indices.
- n_1, n_2 —Spatial indices.
- `fourier_transform-1()`—Inverse Fourier transform subroutine.
- `linear_solver()`—Subroutine for solving a system of linear equations.

Execution:

```

begin
  for  $\nu_1 \leftarrow 0$  to  $N - 1$ 
    begin
       $t_a \leftarrow 0$ 
       $t_b \leftarrow 0$ 
       $t_d \leftarrow 0$ 
      for  $\nu_2 \leftarrow \nu_1$  to  $N_S - 1$  step  $N$ 
        begin
           $t_a \leftarrow t_a + \Phi_s[\nu_2] |\hat{h}[\nu_2]|^2$ 
           $t_b \leftarrow t_b + \Phi_s[\nu_2] \hat{h}^*[\nu_2] \hat{d}^*[\nu_2]$ 
           $t_d \leftarrow t_d + |\hat{d}[\nu_2]|^2$ 
        end
       $\hat{a}[\nu_1] \leftarrow (t_a + \Phi_e[\nu_1]) t_d$ 
       $\hat{b}[\nu_1] \leftarrow t_b$ 
    end
   $a \leftarrow \text{fourier\_transform}^{-1}(\hat{a})$ 
   $b \leftarrow \text{fourier\_transform}^{-1}(\hat{b})$ 
  foreach  $n_1$  s.t.  $n_1 \in C$ 
    begin
       $b_C[\text{ord}(n_1 \text{ in } C)] \leftarrow b[n_1]$ 
      foreach  $n_2$  s.t.  $n_2 \in C$ 
         $A_C[\text{ord}(n_1 \text{ in } C), \text{ord}(n_2 \text{ in } C)] \leftarrow \frac{1}{N} a[n_1 - n_2]$ 
      end
       $f_C \leftarrow \text{linear\_solver}(A_C, b_C)$ 
      for  $n_1 \leftarrow 0$  to  $N - 1$ 
        if  $n_1 \in C$ 
           $f[n_1] \leftarrow f_C[\text{ord}(n_1 \text{ in } C)]$ 
        else
           $f[n_1] \leftarrow 0$ 
      end
    end
  end

```

6.3 Two-Dimensional Restoration

6.3.1 Two-Dimensional Wiener Filter

The derivations of the two-dimensional discrete and end-to-end Wiener filters parallel those of Sections 6.1 and 6.2 and so are not presented. The two-

dimensional Wiener filter is

$$\hat{f}[\mu, \nu] = \frac{\hat{b}[\mu, \nu]}{\hat{a}[\mu, \nu]} \quad (6.73)$$

where for discrete restoration

$$\hat{a}[\mu, \nu] = \Phi_s[\mu, \nu] |\hat{h}[\mu, \nu]|^2 + \Phi_e[\mu, \nu] \quad (6.74)$$

$$\hat{b}[\mu, \nu] = \Phi_s[\mu, \nu] \hat{h}^*[\mu, \nu] \quad (6.75)$$

and for end-to-end restoration

$$\hat{a}[\mu, \nu] = \left(\sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} \Phi_s[\mu - k_1M, \nu - k_2N] |\hat{h}[\mu - k_1M, \nu - k_2N]|^2 + \Phi_e[\mu, \nu] \right) \left(\sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} |\hat{d}[\mu - k_1M, \nu - k_2N]|^2 \right) \quad (6.76)$$

$$\hat{b}[\mu, \nu] = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} \Phi_s[\mu - k_1M, \nu - k_2N] \hat{h}^*[\mu - k_1M, \nu - k_2N] \times \hat{d}^*[\mu - k_1M, \nu - k_2N]. \quad (6.77)$$

6.3.2 Two-Dimensional, Spatially-Constrained Kernel

The two-dimensional Wiener filter is defined for each frequency $[\mu, \nu]$ by the equation

$$\hat{a}[\mu, \nu] \hat{f}[\mu, \nu] = \hat{b}[\mu, \nu]. \quad (6.78)$$

The spatial-domain equivalent is the convolution

$$\frac{1}{MN} \sum_{m'} \sum_{n'} f[m', n'] a[m - m', n - n'] = b[m, n]. \quad (6.79)$$

The constraint set C is the set of locations in the two-dimensional kernel that are not restricted to have zero value. The constraints on the kernel reduces this system of equations to $|C|$ equations in $|C|$ variables:

$$\frac{1}{MN} \sum_{[m', n'] \in C} f[m', n'] a[m - m', n - n'] = b[m, n] \quad [m, n] \in C. \quad (6.80)$$

For discrete restoration, the values of a and b are defined from Equations 6.74 and 6.75. For end-to-end restoration, the values of a and b are defined from Equations 6.76 and 6.77. Because discrete restoration is a special case of end-to-end restoration, only the algorithm for the two-dimensional, end-to-end, mean-square-optimal, spatially-constrained restoration kernel is given. The $M, \times N,$ arrays for the scene spectrum and the acquisition and display transfer functions are defined in the manner described in Section 6.2.3.

Algorithm 6.3 Mean-Square-Optimal, Spatially-Constrained, Kernel for Two-Dimensional, End-to-End Restoration

Input:

M_s, N_s —Scene spectrum dimensions.
 M, N —Image dimensions.
 $\Phi_s [M_s, N_s]$ —Power spectrum of the scene.
 $\Phi_e [M, N]$ —Power spectrum of the noise.
 $\hat{h} [M_s, N_s]$ —Acquisition transfer function.
 $\hat{d} [M_s, N_s]$ —Display transfer function.
 $|C|$ —Number of kernel elements.
 C —Constraint set (an ordered set of the locations of the kernel elements).

Output:

$f [M, N]$ —Mean-square-optimal spatially-constrained kernel.

Declarations: $\hat{a} [M, N]$ —Wiener filter denominator.

$\hat{b} [M, N]$ —Wiener filter numerator.

t_a, t_b, t_d —Temporary variables.

$a [M, N]$ —Spatial transform of Wiener filter denominator.

$b [M, N]$ —Spatial transform of Wiener filter numerator.

$A_C [|C|, |C|]$ —Coefficient matrix.

$f_C [|C|]$ —Input matrix.

$b_C [|C|]$ —Output matrix.

$\mu_1, \mu_2, \nu_1, \nu_2$ —Frequency indices.

m_1, m_2, n_1, n_2 —Spatial indices.

$\text{twod_fourier_transform}^{-1}()$ —Inverse two-dimensional Fourier transform.

$\text{linear_solver}()$ —Subroutine for solving a system of linear equations.

Execution:

```
begin
  for  $\mu_1 \leftarrow 0$  to  $M - 1$ 
    for  $\nu_1 \leftarrow 0$  to  $N - 1$ 
      begin
         $t_a \leftarrow 0$ 
         $t_b \leftarrow 0$ 
         $t_d \leftarrow 0$ 
        for  $\mu_2 \leftarrow \mu_1$  to  $M_s - 1$  step  $M$ 
          for  $\nu_2 \leftarrow \nu_1$  to  $N_s - 1$  step  $N$ 
            begin
               $t_a \leftarrow t_a + \Phi_s [\mu_2, \nu_2] |\hat{h} [\mu_2, \nu_2]|^2$ 
               $t_b \leftarrow t_b + \Phi_s [\mu_2, \nu_2] \hat{h}^* [\mu_2, \nu_2] \hat{d}^* [\mu_2, \nu_2]$ 
            end
          end
        end
      end
    end
  end
```



```

         $t_d \leftarrow t_d + |\hat{d}[\mu_2, \nu_2]|^2$ 
    end
     $\hat{a}[\mu_1, \nu_1] \leftarrow (t_a + \Phi_e[\mu_1, \nu_1]) t_d$ 
     $\hat{b}[\mu_1, \nu_1] \leftarrow t_b$ 
end
 $a \leftarrow \text{twod\_fourier\_transform}^{-1}(\hat{a})$ 
 $b \leftarrow \text{twod\_fourier\_transform}^{-1}(\hat{b})$ 
foreach  $[m_1, n_1]$  s.t.  $[m_1, n_1] \in C$ 
    begin
         $b_C[\text{ord}([m_1, n_1] \text{ in } C)] \leftarrow b[m_1, n_1]$ 
        foreach  $[m_2, n_2]$  s.t.  $[m_2, n_2] \in C$ 
             $A_C[\text{ord}([m_1, n_1] \text{ in } C), \text{ord}([m_2, n_2] \text{ in } C)]$ 
                 $\leftarrow \frac{1}{MN} a[m_1 - m_2, n_1 - n_2]$ 
        end
         $f_C \leftarrow \text{linear\_solver}(A_C, b_C)$ 
        for  $m_1 \leftarrow 0$  to  $M - 1$ 
            for  $n_1 \leftarrow 0$  to  $N - 1$ 
                if  $[m_1, n_1] \in C$ 
                     $f[m_1, n_1] \leftarrow f_C[\text{ord}([m_1, n_1] \text{ in } C)]$ 
                else
                     $f[m_1, n_1] \leftarrow 0$ 
                end
            end
        end
    end
end

```

6.4 Asides

6.4.1 Symmetry

Imaging system functions often exhibit symmetry. When symmetry is present, the size of the linear system of equations that defines the optimal restoration kernel can be reduced. Consider the one-dimensional, discrete equations:

$$\mathbf{A} \mathbf{f} = \mathbf{b} \quad (6.81)$$

where

$$\mathbf{A}[n_1, n_2] = \frac{1}{N} \sum_{\nu} \left(\Phi_s[\nu] |\hat{h}[\nu]|^2 + \Phi_e[\nu] \right) W_N^{\nu(n_1 - n_2)} \quad (6.82)$$

$$\mathbf{b}[n] = \sum_{\nu} \Phi_s[\nu] \hat{h}^*[\nu] W_N^{\nu n}. \quad (6.83)$$

The power spectra of the scene and noise are symmetric. If the transfer function \hat{h} is symmetric, then both a and b are symmetric and half the

equations are redundant. In two-dimensions, radial symmetry can reduce the number of equations by nearly a factor of 8.

As a practical matter, these savings in solving the system of equations that defines the optimal kernel are not very significant. First, only small kernels are practical, so the system of equations is always small. For example, a 49-element kernel is about as large as is practical. In this case, there are at most 49 linear equations in 49 variables. Second, the restoration kernel is usually pre-computed, before the image is actually restored. Therefore, the cost of computing the kernel values is typically not important.

6.4.2 Generalized Inverse

The Moore-Penrose generalized inverse is another method for generating a small kernel. This technique is described in Chapter 5. The generalized inverse bears some similarity to the technique developed in this chapter. Both are *mean-square optimal*. However, the generalized inverse minimizes the mean-square difference between the small kernel and the inverse transform of the (unconstrained) Wiener filter. The technique described in this chapter minimizes the expected mean-square restoration error. The purpose of the kernel is to restore images, so restoration error is a more pertinent criterion.

References

- [1] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, New York, NY, 1985.
- [2] Friedrich O. Huck, Carl L. Fales, Nesim Haylo, Richard W. Samms, and Kathryn Stacy. Image gathering and processing: information and fidelity. *Journal of the Optical Society of America A*, 2(10):1644–1666, 1985.
- [3] Carl L. Fales, Friedrich O. Huck, Judith A. McCormick, and Stephen K. Park. Wiener restoration of sampled image data: end-to-end analysis. *Journal of the Optical Society of America A*, 5(3):300–314, 1988.

Chapter 7

Artificial Scenes and Simulated Imaging

7.1 Experimental Methodology

Proper experimental methods are as important to digital image processing research as to other scientific pursuits. All scientific reports of experimental work should include a precise statement of the experiment, a complete presentation of any non-standard algorithms, a clear description of the controlled environment, and a full discussion of results that addresses issues such as measures of success and generality of use. Two recent surveys of computer vision and image processing research conclude that many published papers do not adhere to this standard[1,2]. This failure leaves questions about the validity and reliability of results and makes replicating experiments and building on previous results difficult.

This chapter describes a software simulation environment for controlled image processing research. The simulation is based on the end-to-end imaging model developed in Chapter 3. This model is appropriate for a variety of problems that involve image acquisition and display (e.g., image restoration, enhancement, and compression/decompression). By using a model-based simulation, research can be conducted with greater precision, flexibility, and portability than is possible using real systems[3,4]. Of course, the validity of the research depends on the completeness of the model, the accuracy of the simulation, and the correctness of the implementation. It is always important to confirm simulation results by comparison with actual data.

Section 7.2 describes methods for generating artificial scenes. Section 7.3 details techniques for simulating the components of the digital image processing model developed in Chapter 3—image formation, sampling, noise,

and image display. Simulated restoration results are presented in Chapter 8.

7.2 Artificial Scenes

Computer-generated artificial scenes offer greater accuracy, precision, flexibility, and portability than physical scenes. Research using physical scenes entails an inevitable degree of uncertainty and imprecision. The characteristics of physical scenes cannot be known directly; they must be measured indirectly using devices that may themselves be components of the research environment. Computer-generated artificial scenes are completely and precisely known. Though it is not difficult to use a variety of physical scenes, it is very difficult to specify an ensemble with specific statistics. Ensembles of artificial scenes can be created using relatively simple techniques that give full control over important statistics. Many physical scenes are not portable. Artificial scenes can be easily communicated from one computer to another and exactly reproduced.

Artificial scenes should accurately model the important characteristics of physical scenes. Physical scenes fill the spatial continuum, but artificial scenes must be represented with a discrete data structure. Physical scenes may have details finer than can be resolved by digital imaging systems. This fact has important implications for image acquisition—sampling is not invertible because high-frequencies alias at lower frequencies obscuring the actual components at those frequencies. This is seen as a loss of subpixel detail. Also, sampling is shift-variant—subpixel shifts of the scene relative to the sampling grid (sample/scene phase shift) produce varying images[5,6]. If artificial scenes are to accurately simulate physical scenes, they must have the capacity to contain subpixel details (frequency components beyond the Nyquist frequency).

Both types of two-dimensional artificial scenes described in this section have the capacity for high-frequency components. The first type, called *Fourier scenes*, are two-dimensional Fourier series. Techniques for generating Fourier scenes provide control over important mathematical and statistical properties of the scene, but no direct control over spatial structure. The second type of artificial scenes, *digital scenes*, are digital images with *super-resolution*—resolution higher than the sampling resolution. Techniques for generating digital scenes provide control over the spatial structure of the scene, but no direct control over important mathematical and statistical properties.

7.2.1 Fourier Scenes

Fourier scenes are two-dimensional Fourier series. A scene specified by a Fourier series is periodic, is defined at every point (not just at the sample points), and can contain arbitrarily high spatial-frequencies. A Fourier scene is stored as a two-dimensional, complex-valued array \hat{s} . The continuous scene is then defined by the array values as

$$s(x, y) = \sum_{\mu=-\infty}^{\infty} \sum_{\nu=-\infty}^{\infty} \hat{s}[\mu, \nu] W_M^{\mu x} W_N^{\nu y}. \quad (7.1)$$

For real-valued scenes, the complex Fourier series is Hermitian ($\hat{s}[\mu, \nu] = \hat{s}^*[-\mu, -\nu]$).

A Fourier scene can have details finer than any sampling grid (i.e., it can contain frequency components above the Nyquist limits, $|\mu| > M/2$ or $|\nu| > N/2$). As a practical matter, the size of the computer imposes a bound on the Fourier series; however, this limitation is seldom a problem, because the significance of the terms of the Fourier series generally decreases as frequency increases and typical imaging systems virtually eliminate frequency components that are well beyond the Nyquist limit. Therefore, truncating the infinite series in Equation 7.1 results in negligible error. Defining a bound for a Fourier series involves a tradeoff between accuracy on one hand and computation and storage costs on the other. A series with frequencies two to four times the Nyquist limit in each dimension is usually sufficient for accurate simulation—the contribution of higher frequencies in most real systems is negligible.

Equation 7.1 is the Cartesian form for the Fourier series. It can be written equivalently in magnitude-phase notation as

$$s(x, y) = \sum_{\mu=-\infty}^{\infty} \sum_{\nu=-\infty}^{\infty} \hat{s}_\rho[\mu, \nu] \cos\left(2\pi\left(\frac{\mu x}{M} + \frac{\nu y}{N}\right) - \hat{s}_\phi[\mu, \nu]\right). \quad (7.2)$$

Again assuming real-valued scenes and Hermitian Fourier series, the magnitude \hat{s}_ρ is an even function and the phase \hat{s}_ϕ is odd.

Specifying the magnitudes of the terms of the Fourier scene of Equation 7.2 affords direct control over important statistics of the scene. Randomizing the phase components generates an ensemble of scenes. Two important ensemble statistics are the mean

$$\eta_s(x, y) = E\{s(x, y)\} \quad (7.3)$$

and the autocorrelation

$$R_s(x_1, y_1; x_2, y_2) = E\{s(x_1, y_1) s(x_2, y_2)\}. \quad (7.4)$$

The Fourier transform of the autocorrelation is the power spectrum Φ_s . Fourier scenes are real and periodic, so the power spectra are real, symmetric, and discrete.

As defined in Chapter 2, a stochastic scene is wide-sense (or weakly or second-order) stationary if the mean and autocorrelation are independent of position—that is, the mean is constant:

$$E\{s(x, y)\} = \eta_s \quad (7.5)$$

and the autocorrelation is a function of the distance between points:

$$E\{s(x_1, y_1)s(x_2, y_2)\} = R_s(x_1 - x_2, y_1 - y_2). \quad (7.6)$$

The stochastic scene is strict-sense stationary if all statistics are spatially invariant[7].

The following algorithm for producing scenes provides direct control over the mean and autocorrelation and generates a stationary, stochastic Fourier scene.

Algorithm 7.1 Generating Fourier Scenes

1. Set bound on the frequencies of the Fourier series. It is convenient to truncate the series at frequencies that are integer multiples of the sampling frequency (twice the Nyquist limit). That is, bound the series at $\pm M_s/2$ cycles per M pixels vertically and $\pm N_s/2$ cycles per N pixels horizontally where

$$M_s = S_M M \quad (7.7)$$

$$N_s = S_N N \quad (7.8)$$

and S_M and S_N are integers, typically 2 or 4.

2. Set the amplitude of the zero-frequency term of the Fourier series to the expected value η_s that is desired:

$$\hat{s}_\rho[0, 0] \leftarrow \eta_s. \quad (7.9)$$

3. Set the amplitude of the other spatial frequencies of the Fourier series to the square-root of the power spectrum Φ_s that is desired:

$$\hat{s}_\rho[\mu, \nu] \leftarrow \sqrt{\Phi_s[\mu, \nu]}. \quad (7.10)$$

(The power spectrum Φ_s is real-valued and symmetric.)

4. Set the phase of each spatial frequency term to an independent random variable with a uniform probability density between $-\pi$ and π . (For real-valued images, the phase must be antisymmetric, so only about half of the phase values are freely set.) In practice, the two-dimensional phase array \hat{s}_ϕ is created using a pseudo-random number generator. Randomizing the phase is motivated in part by the observation that the phase of physical scenes typically looks random. More fundamentally, as proven later, this process for generating scenes is wide-sense stationary.
5. The values of the scene itself, s , are given by the Fourier series as in Equation 7.2.

This algorithm resembles a common analog procedure used by electrical engineers to generate physical signals by passing the output of a white-noise generator (a signal with constant spectral magnitude \hat{s}_ρ) through a filter with a transfer function equal to the desired spectral magnitude[8, pp. 250-2]. The choice of a uniformly distributed random phase is an obvious one (e.g., Gonsalves[9]), but the statistical implications of this choice should be analyzed.

Lemma 7.1 *The expected value of a stochastic scene produced by Algorithm 7.1 is independent of position.*

Proof: From the linearity of the expectation operator:

$$\begin{aligned} E\{s(x, y)\} &= E\left\{\sum_{\mu, \nu} \hat{s}_\rho[\mu, \nu] \cos\left(2\pi\left(\frac{\mu x}{M} + \frac{\nu y}{N}\right) - \hat{s}_\phi[\mu, \nu]\right)\right\} \\ &= \sum_{\mu, \nu} \hat{s}_\rho[\mu, \nu] E\left\{\cos\left(2\pi\left(\frac{\mu x}{M} + \frac{\nu y}{N}\right) - \hat{s}_\phi[\mu, \nu]\right)\right\} \end{aligned} \quad (7.11)$$

Except at the zero-frequency, the distribution of the phase \hat{s}_ϕ is uniform, so:

$$\begin{aligned} E\left\{\cos\left(2\pi\left(\frac{\mu x}{M} + \frac{\nu y}{N}\right) - \hat{s}_\phi[\mu, \nu]\right)\right\} &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \cos\left(2\pi\left(\frac{\mu x}{M} + \frac{\nu y}{N}\right) - \hat{s}_\phi\right) d\hat{s}_\phi \\ &= \begin{cases} 1 & \text{if } [\mu, \nu] = [0, 0] \\ 0 & \text{otherwise} \end{cases} \end{aligned} \quad (7.12)$$

Therefore:

$$\begin{aligned} E\{s(x, y)\} &= \hat{s}_\rho[0, 0] \\ &= \eta_s \end{aligned} \quad (7.13)$$

for any spatial location (x, y) .

□

Lemma 7.2 *The autocorrelation of a stochastic scene produced by Algorithm 7.1 is independent of position.*

Proof: From the linearity of the expectation operator:

$$\begin{aligned}
 E \{s(x_1, y_1) s(x_2, y_2)\} &= E \left\{ \left(\sum_{\mu_1, \nu_1} \hat{s}_\rho[\mu_1, \nu_1] \cos \left(2\pi \left(\frac{\mu_1 x_1}{M} + \frac{\nu_1 y_1}{N} \right) - \hat{s}_\phi[\mu_1, \nu_1] \right) \right) \right. \\
 &\quad \left. \times \left(\sum_{\mu_2, \nu_2} \hat{s}_\rho[\mu_2, \nu_2] \cos \left(2\pi \left(\frac{\mu_2 x_2}{M} + \frac{\nu_2 y_2}{N} \right) - \hat{s}_\phi[\mu_2, \nu_2] \right) \right) \right\} \\
 &= \sum_{\mu_1, \nu_1} \sum_{\mu_2, \nu_2} \hat{s}_\rho[\mu_1, \nu_1] \hat{s}_\rho[\mu_2, \nu_2] E \left\{ \cos \left(2\pi \left(\frac{\mu_1 x_1}{M} + \frac{\nu_1 y_1}{N} \right) - \hat{s}_\phi[\mu_1, \nu_1] \right) \right. \\
 &\quad \left. \times \cos \left(2\pi \left(\frac{\mu_2 x_2}{M} + \frac{\nu_2 y_2}{N} \right) - \hat{s}_\phi[\mu_2, \nu_2] \right) \right\} \quad (7.14)
 \end{aligned}$$

The phase values are independent, so:

$$\begin{aligned}
 E \left\{ \cos \left(2\pi \left(\frac{\mu_1 x_1}{M} + \frac{\nu_1 y_1}{N} \right) - \hat{s}_\phi[\mu_1, \nu_1] \right) \cos \left(2\pi \left(\frac{\mu_2 x_2}{M} + \frac{\nu_2 y_2}{N} \right) - \hat{s}_\phi[\mu_2, \nu_2] \right) \right\} \\
 &= E \left\{ \cos \left(2\pi \left(\frac{\mu_1 x_1}{M} + \frac{\nu_1 y_1}{N} \right) - \hat{s}_\phi[\mu_1, \nu_1] \right) \right\} \\
 &\quad \times E \left\{ \cos \left(2\pi \left(\frac{\mu_2 x_2}{M} + \frac{\nu_2 y_2}{N} \right) - \hat{s}_\phi[\mu_2, \nu_2] \right) \right\} \\
 &= 0 \quad \text{where } [\mu_1, \nu_1] \neq [\mu_2, \nu_2] \text{ and } [\mu_1, \nu_1] \neq [-\mu_2, -\nu_2] \quad (7.15)
 \end{aligned}$$

The remaining terms of the sum yield:

$$\begin{aligned}
 E \{s(x_1, y_1) s(x_2, y_2)\} &= 2 \sum_{\mu, \nu} (\hat{s}_\rho[\mu, \nu])^2 E \left\{ \cos \left(2\pi \left(\frac{\mu x_1}{M} + \frac{\nu y_1}{N} \right) - \hat{s}_\phi[\mu, \nu] \right) \right. \\
 &\quad \left. \times \cos \left(2\pi \left(\frac{\mu x_2}{M} + \frac{\nu y_2}{N} \right) - \hat{s}_\phi[\mu, \nu] \right) \right\} - \hat{s}_\rho[0, 0] \quad (7.16)
 \end{aligned}$$

Because $2 \cos(\alpha) \cos(\beta) = \cos(\alpha + \beta) + \cos(\alpha - \beta)$:

$$\begin{aligned}
 E \left\{ \cos \left(2\pi \left(\frac{\mu x_1}{M} + \frac{\nu y_1}{N} \right) - \hat{s}_\phi[\mu, \nu] \right) \cos \left(2\pi \left(\frac{\mu x_2}{M} + \frac{\nu y_2}{N} \right) - \hat{s}_\phi[\mu, \nu] \right) \right\} \\
 &= \frac{1}{2} E \left\{ \cos \left(2\pi \left(\frac{\mu(x_1+x_2)}{M} + \frac{\nu(y_1+y_2)}{N} \right) - 2\hat{s}_\phi[\mu, \nu] \right) \right\} \\
 &\quad + \frac{1}{2} \cos \left(2\pi \left(\frac{\mu(x_1-x_2)}{M} + \frac{\nu(y_1-y_2)}{N} \right) \right) \\
 &= \frac{1}{2} \cos \left(2\pi \left(\frac{\mu(x_1-x_2)}{M} + \frac{\nu(y_1-y_2)}{N} \right) \right) \quad [\mu, \nu] \neq [0, 0] \quad (7.17)
 \end{aligned}$$

Therefore:

$$\begin{aligned}
 E \{s(x_1, y_1) s(x_2, y_2)\} &= \sum_{\mu, \nu} (\hat{s}_\rho[\mu, \nu])^2 \cos \left(2\pi \left(\frac{\mu(x_1-x_2)}{M} + \frac{\nu(y_1-y_2)}{N} \right) \right) \quad (7.18)
 \end{aligned}$$

This expression depends only on the distance between (x_1, y_1) and (x_2, y_2) and is therefore independent of spatial position (shift-invariant).

□

As an example, Algorithm 7.1 can be used to generate white noise (to an arbitrary spatial frequency). A noise process is white if the values in the scene are not correlated with each other. That is

$$E \{s(x_1, y_1) s(x_2, y_2)\} = E \{s(x_1, y_1)\} E \{s(x_2, y_2)\} \quad (7.19)$$

where $(x_1, y_1) \neq (x_2, y_2)$. Zero-mean white noise is generated with Algorithm 7.1 by setting the amplitude at all non-zero frequencies to the square root of the expected of the spatial noise process normalized by the number of terms in the Fourier series $M_s N_s$:

$$\hat{s}_\rho[\mu, \nu] = \begin{cases} 0 & \text{if } [\mu, \nu] = [0, 0] \\ \sqrt{R_s(0, 0)/(M_s N_s)} & \text{otherwise.} \end{cases} \quad (7.20)$$

In practice, the series cannot be infinite, but if the highest frequency of interest in the scene is known (e.g., the sampling frequency is specified), the Fourier series can be made large enough to insure summation over a full period for all implied scene elements. Under these conditions, the process is white (from the orthogonality of the cosine):

$$\begin{aligned} E \{s(x_1, y_1) s(x_2, y_2)\} &= \sum_{\mu, \nu} (\hat{s}_\rho[\mu, \nu])^2 \cos \left(2\pi \left(\frac{\mu(x_1 - x_2)}{M} + \frac{\nu(y_1 - y_2)}{N} \right) \right) \\ &= \begin{cases} R_s(0, 0) & \text{if } (x_1, y_1) = (x_2, y_2) \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \quad (7.21)$$

The proofs of Lemmas 7.1 and 7.2 illustrate the control of Algorithm 7.1 over the expected value and autocorrelation of the scene. Together, these lemmas establish that stochastic scenes defined by Algorithm 7.1 are wide-sense stationary, but a stronger theorem holds—stochastic scenes defined by Algorithm 7.1 are strict-sense stationary.

Theorem 7.1 *Stochastic scenes produced by Algorithm 7.1 are strictly stationary.*

Proof: Consider a scene s_1 defined by Algorithm 7.1:

$$s_1(x, y) = \sum_{\mu, \nu} \hat{s}_\rho[\mu, \nu] \cos(\theta_1[\mu, \nu] - \hat{s}_\phi[\mu, \nu]) \quad (7.22)$$

where

$$\theta_1[\mu, \nu] = 2\pi \left(\frac{\mu x}{M} + \frac{\nu y}{N} \right). \quad (7.23)$$

The same process with a spatial shift is

$$s_2(x, y) = \sum_{\mu, \nu} \hat{s}_\rho[\mu, \nu] \cos(\theta_2[\mu, \nu] - \hat{s}_\phi[\mu, \nu]) \quad (7.24)$$

where

$$\theta_2[\mu, \nu] = 2\pi \left(\frac{\mu(x+a)}{M} + \frac{\nu(y+b)}{N} \right). \quad (7.25)$$

The spatial shift changes only the phase term θ .

The statistics of the scenes s_1 and s_2 are determined by their probability density functions or PDFs. The function θ plays no part in the probability density. Regardless of θ , the PDF of the spatial contribution of each term of the Fourier series

$$Z_{\mu, \nu} = \hat{s}_\rho[\mu, \nu] \cos(\theta[\mu, \nu] - \hat{s}_\phi[\mu, \nu]) \quad (7.26)$$

is[7, p. 100]

$$\text{pdf}_{Z_{\mu, \nu}}(z) = \begin{cases} \left(\pi \sqrt{\hat{s}_\rho[\mu, \nu]^2 - z^2} \right)^{-1} & \text{if } |z| < \hat{s}_\rho[\mu, \nu] \\ 0 & \text{otherwise.} \end{cases} \quad (7.27)$$

The phase values of the Fourier series are independent and functions of independent random variables are independent[7, p. 132], so the spatial contribution of each term of the Fourier series is independent. Therefore, if $[\mu_1, \nu_1] \neq [\mu_2, \nu_2]$, Z_{μ_1, ν_1} and Z_{μ_2, ν_2} are independent. The PDF of a sum of independent random variables is the convolution of their densities[7, p. 134], so the PDF for a point in the scene is equal to the convolution of functions of the form of Equation 7.27. Therefore, spatial shift does not affect the PDF and the stochastic scene is strictly stationary.

□

Corollary 7.1 *The probability density function of a stochastic scene generated by Algorithm 7.1 is*

$$\text{pdf}_s(z) = \mathbf{K}_{\mu, \nu} \text{pdf}_{Z_{\mu, \nu}}(z) \quad (7.28)$$

where \mathbf{K} indicates a convolution series (or cascade) and $\text{pdf}_{Z_{\mu, \nu}}(z)$ is defined in Equation 7.27.

Unlike most physical scenes, Fourier scenes are portable. Fourier scenes are easily specified and communicated. In the worst case, it may be necessary to specify all of the elements of the two-dimensional arrays of the

Fourier scenes, but this is often unnecessary. Frequently, the power spectrum has a standard functional form (e.g., a Gaussian) that can be specified by only a few parameters. Following Algorithm 7.1 and using a shared, standard random-number generator[10], the phase array can be specified by the seed for the random-number generator. Widespread use of Fourier scenes would facilitate image processing and computer vision research by providing a virtually unlimited common body of scenes for evaluating algorithms.

7.2.2 Digital Scenes

Dictating the spatial structure of Fourier scenes is difficult. Most image processing applications, however, focus on spatial structure. For example, a radiologist is concerned with recognizing tumors, not with image ensemble statistics. Fourier scenes are useful, but conclusions for specific applications will undoubtedly be more subjective. Digital scenes are digital images used to simulate scenes. Because they are defined in the spatial domain, they provide direct control over spatial structure. If a digital scene is specified to super-resolution—resolution higher than the subsequent sampling lattice—it has the capacity for details finer than the sampling lattice and can be used to simulate aliasing and sample-scene phase shift.

A relatively small collection of digital images with recognizable spatial structure have long been used as artificial scenes[11]. Many researchers use a digital scene with the same resolution as the digital image. For example, in their text on image restoration, Andrews and Hunt[12, p. 128] blurred a 500×500 digital scene and added noise to produce a 500×500 digital image. The blurred and noisy image was then restored. If the artificial scene has the same resolution as the image, there are no sampling effects. This approach ignores aliasing and sample/scene phase—important problems for image restoration[13].

If the digital scene has finer resolution than the digital image produced from it, the simulation can model the important effects of sampling. Just as a Fourier scene on a digital computer has a cutoff frequency, the ratio of digital scene elements (*scenels*) to digital picture elements (*pixels*) must be finite. The best scenel:pixel ratio depends on the desired level of detail in the scene relative to the sampling grid—the finer the detail, the greater the ratio must be. There is a tradeoff between accuracy on one hand and computational and storage costs on the other. A digital scene that has twice the resolution of the sampling lattice is sufficient to exhibit aliasing at all frequencies of the digital image. Typically, spatial frequencies beyond four times the resolution of the sampling lattice are effectively lost (cutoff by prefiltering) before sampling.

Digital images of any kind can be used as digital scenes. Because they are to be resampled, digital scenes must be fairly large. Large digital images can be obtained from many sources. Landsat images are very large and commonly available. Large charged-coupled device (CCD) photosensor arrays are now available. Digital scenes can be generated artificially on a computer. Modestino and Freis[14] suggest a stochastic process to generate images with spatial edge structure. Similar spatial processes could be devised for other spatial structures. Image generation tools such as the *movie.byu* software developed at Brigham Young University offer flexibility and control over three-dimensional digital scene generation.

7.2.3 Relationship Between Fourier Scenes and Digital Scenes

Fourier scenes and digital scenes complement one another. Fourier scenes are specified in the frequency domain and provide control over important scene statistics. Digital scenes are specified in the spatial domain and provide control over spatial structure. Both are able to model the subpixel details (or high-frequency components) of physical scenes and exhibit aliasing and sample/scene phase shift during sampling. Fourier scenes are defined over the spatial continuum, but digital scenes are defined only at the discrete points (scenels) of a spatial array. It is possible to specify a method for converting from one representation to the other consisting of an interpolator to convert digital scenes to Fourier scenes and a sampling lattice to convert Fourier scenes to digital scenes.

Digital scenes are defined only at the scenels. Interpolating between these points results in a continuous scene. The Fourier series of the resulting continuous scene can be calculated from the discrete Fourier transform of the digital scene and the transfer function of the interpolator. Of particular interest is the relationship between the DFT of a digital scene and the Fourier series of the continuous scene formed from it using the "ideal" sinc interpolator. The spectrum of a digital scene is periodic. The spectrum of the sinc function is a perfect low-pass filter—a two-dimensional pulse about the origin that is the size of the period of the digital scene spectrum. (See Appendix A.) The product of the scene spectrum and this two-dimensional pulse is the digital scene DFT—the complex Fourier series of the interpolated scene is the array of values of the DFT of the digital scene.

The inverse process generates a digital scene from a Fourier scene. Spatially sampling a Fourier scene with superresolution yields a digital scene. Spatial sampling is equivalent to spectral convolution. The two-dimensional shah function is its own Fourier transform and spectral convolution with the

shah function yields a periodic spectrum. If the frequency of the conversion lattice is twice the highest frequency of the Fourier scene (i.e., provides the necessary superresolution), the spectrum of the result is the periodic extension of the Fourier series. Therefore, superresolution sampling and ideal interpolation are invertible methods for converting between Fourier and digital scenes.

All of this may sound more complicated than it is: in practice, the conversion is nothing more than computing a DFT. For example, suppose the image period is $M = N = 256$ and consider a 1024×1024 element Fourier series \hat{s} with frequencies $|\mu| < 2M$ and $|\nu| < 2N$. The inverse DFT of the Fourier scene yields a digital scene with 1024×1024 elements covering the image space with four-fold superresolution (i.e., values every one-fourth pixel in each dimension). The forward DFT of the digital scene is the Fourier scene.

The complex-valued Fourier series and the DFT are inefficient data structures for real-valued scenes—half of the values are redundant. Similarly, fast Fourier transforms for complex data perform about twice the number of computations required for real-valued data. The traditional Fourier transform is used in the text of this dissertation because more readers are familiar with it. However, in actual computations the Hartley transform was employed. The Hartley transform is an equivalent transform for efficiently computing and representing the spectra of real-valued scenes[15].

7.3 Simulated Imaging

This section describes techniques for simulating the imaging process. Software imaging simulation is an exact, flexible, and portable environment for image processing research. The characteristics of real, physical devices can be measured or estimated, but such characterizations are inexact. Moreover, the characteristics of a real device may differ from image to image (changing over time). A simulated imaging system is exactly known and fully controllable. A real imaging system provides only a limited environment for testing image processing techniques. Software simulation of imaging systems provides a flexible image acquisition and display testbed that can be easily modified and adapted. There are no “standard” physical imaging devices, but shared, simulated devices can be installed in any imaging software environment.

The model of Chapter 3 is useful for many, but not all, applications. For example, in some applications, display is not be required. In other applications, additional components may be necessary (e.g., variable illumination

or viewing geometry). It is difficult to define a single environment flexible enough for all applications, but model-based simulation can be adapted for specific applications.

Simulated imaging can be computed equivalently in either the spatial domain or the frequency domain. Figure 7.1 illustrates the alternate paths. Though the result in either case is the same, the computational costs can be quite different. Often, frequency domain computation is more efficient even if the scene is specified in the spatial domain (as a digital scene) and the result is presented in the spatial domain (as a displayed image). The tradeoffs in the choice of domain are considered in the following examination of imaging simulation.

7.3.1 Image Formation

The model shown in Figure 7.1 begins with image formation. As discussed in Chapter 3, many imaging devices are accurately modeled as linear, shift-invariant systems. A LSI system is fully characterized by its point spread function (PSF) or its optical transfer function (OTF). In software simulation, these functions can be exactly specified and easily varied. A wide variety of electro-optical imaging systems can be modeled by OTFs of the form:

$$\hat{h}(u, v) = \exp\left(-\left(\sqrt{u^2 + v^2}/\alpha\right)^\beta\right) \quad (7.29)$$

where α is the spatial frequency at which $\hat{h}(u, v) = e^{-1} \approx 0.37$ and β is a shape parameter typically ranging from 1 to slightly more than 2[16]. All of the simulated results presented in Chapter 8 were produced using this OTF model.

Image formation can be computed in the spatial domain as the convolution of the scene and PSF or in the frequency domain as the product of the scene spectrum and OTF. The scencels of a digital scene with superresolution ratios S_M and S_N are spaced $1/S_M$ units vertically and $1/S_N$ units horizontally. There are $M_s \times N_s$ scencels where $M_s = S_M M$ and $N_s = S_N N$. The spatial convolution $s' = s * h$ is

$$s'\left(\frac{m}{S_M}, \frac{n}{S_N}\right) = \frac{1}{M_s N_s} \sum_{m'=0}^{M_s-1} \sum_{n'=0}^{N_s-1} s\left(\frac{m-m'}{S_M}, \frac{n-n'}{S_N}\right) h\left(\frac{m'}{S_M}, \frac{n'}{S_N}\right) \quad (7.30)$$

where h is the PSF and s is a digital scene. Though s and s' are arrays, the parenthetical notation of continuous images is retained in this equation to emphasize that the arrays simulate continuous images. The Fourier scene

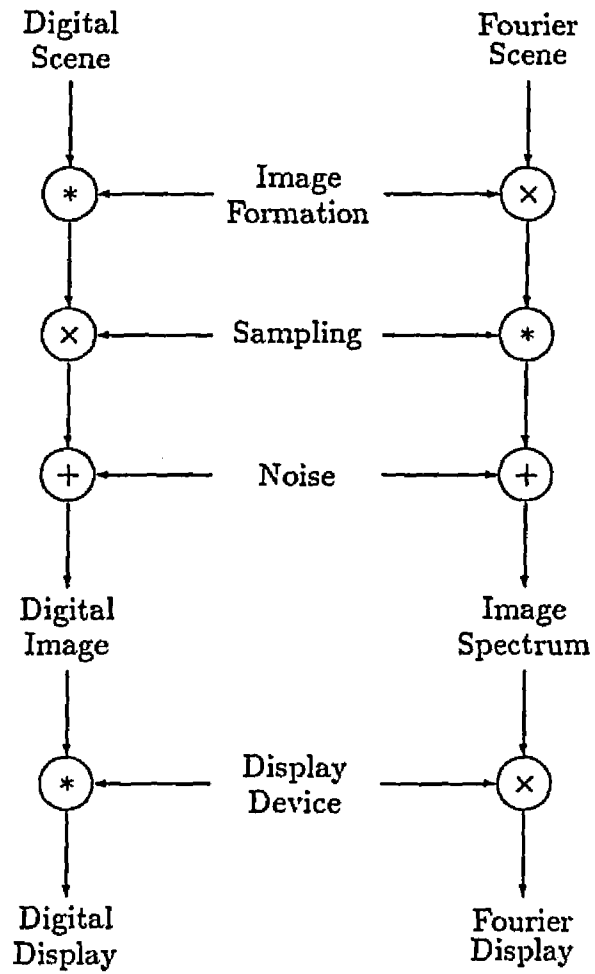


Figure 7.1: Simulation Processing

array is also $M_s \times N_s$ with frequencies to $\pm S_M/2$ and $\pm S_N/2$ times the sampling rates (M and N). In the frequency domain, the product $\hat{s}' = \hat{s}\hat{h}$ is

$$\hat{s}'[\mu, \nu] = \hat{s}[\mu, \nu] \hat{h}(\mu, \nu) \quad (7.31)$$

where \hat{h} is the OTF and \hat{s} is complex Fourier scene.

The computational complexity of Equation 7.30 is $O(M_s^2 N_s^2) - O(M_s N_s)$ for each of $M_s N_s$ points. In practice, the result of the convolution need only be calculated at the $M \times N$ points that are subsequently sampled, reducing the computation to $O(M_s N_s MN)$. If the acquisition device can be accurately modeled by a small PSF kernel, with $M_h \times N_h$ non-zero terms, then the spatial computation is further reduced to $O(M_h N_h MN)$. The computation of Equation 7.31 is $O(M_s N_s)$. If a digital scene s is to be processed in the frequency domain, the fast Fourier transform to calculate the spectrum \hat{s} is $O(M_s N_s \log_2(M_s N_s))$. The inverse transform is also \hat{s} is $O(M_s N_s \log_2(M_s N_s))$.

The execution times for convolution and frequency domain processing on a Sun3/260 are presented in Chapter 5. The size of the PSF is the most variable of the values. The complexity of frequency multiplication and the Fourier transform are independent of the PSF size. The cost of spatial convolution is directly proportional to the number of elements in the PSF. If the PSF is very small, spatial convolution is more efficient than frequency domain processing. For large PSFs, frequency computation is significantly more efficient than spatial convolution. Also, convolution is relatively more efficient for large digital scenes. In the simulations of Chapter 8, the PSFs are the same size as the images, so frequency domain processing is used.

7.3.2 Sampling

The sampling lattice of almost every digital imaging system is a rectangular array of uniformly-spaced points. This sampling lattice can be simulated in software and applied to an artificial scene. Sampling can be computed as the spatial product of the continuous image and the sampling function or as the convolution of the image spectrum and the transform of the sampling function. Considered independently of other steps, sampling is more efficiently computed in the spatial domain, but in sequence with the other steps, frequency computation is often more efficient.

Sampling the acquired image s' of Equation 7.30 with the comb (or shah) sampling function III is given by spatial multiplication $p' = s' \text{III}$ where

$$p'[m, n] = s'(m, n). \quad (7.32)$$

(Because the digital scene has superresolution, the scene $s'(m, n)$ is stored in the array location $s'[mS_M, nS_N]$.) The resulting image is $M \times N$, so the sampling operation is $O(MN)$. If spatial sampling is implemented as frequency convolution, the result is

$$\hat{p}'[\mu, \nu] = \sum_{\mu'=0}^{S_M-1} \sum_{\nu'=0}^{S_N-1} \hat{s}'[\mu - \mu'M, \nu - \nu'N]. \quad (7.33)$$

The frequency computation is $O(M_s N_s)$.

Though the complexity of the spatial computation is less than the frequency computation, both expressions are dominated by the cost of simulating image formation. As explained in Section 7.3.1, image formation is usually more efficiently simulated in the frequency domain. In that case, spatial sampling would require an inverse transform, more than offsetting any savings in the sampling computation. Even if the inverse transform must subsequently be executed, the folding of frequencies in Equation 7.33 reduces the cost from $O(M_s N_s \log_2(M_s N_s))$ to $O(MN \log_2(MN))$.

In physical images, the amount of aliasing or aliased noise is difficult to calculate. Doing so requires *a priori* knowledge of the scene spectrum beyond the Nyquist limit or a measure of it using a higher resolution system. In contrast, the aliased components of the simulated image are exactly known. For example, the energy of the aliased components is

$$\epsilon_a^2 = \sum_{|\mu| > \frac{M}{2}} \sum_{|\nu| > \frac{N}{2}} |\hat{s}'[\mu, \nu]|^2 \quad (7.34)$$

Sample-scene shift is simulated as

$$s' \left(\frac{m-a}{S_M}, \frac{n-b}{S_N} \right) \quad (7.35)$$

where the row-shift is a/S_M and the column-shift is b/S_N . In this equation, there are effectively $S_M \times S_N$ possible different subpixel-unit shifts of the sampling lattice relative to the scene. In the frequency domain, sample/scene shift is simulated as

$$\hat{s}'(\mu, \nu) W_M^{\mu a} W_N^{\nu b}. \quad (7.36)$$

The spatial domain computation is straightforward only if the shifts are multiples of the subpixel superresolution units; fractional shifts require interpolation. Arbitrary shifts are easily computed in the frequency domain.

7.3.3 Noise

In the simulations in this dissertation, additive, signal-independent, white noise is assumed because it is a good model of many real sources of noise

and the assumption simplifies many image processing problems. This is a common assumption. White-noise can be generated in the frequency domain as described in Section 7.2.1 (p. 113) or equivalently as a random spatial process with normal distribution[17]. Then the noise is added to the image as $p = p' + e$ where

$$p[m, n] = p'[m, n] + e[m, n] \quad (7.37)$$

or to the image spectrum as $\hat{p} = \hat{p}' + \hat{e}$ where

$$\hat{p}[\mu, \nu] = \hat{p}'[\mu, \nu] + \hat{e}[\mu, \nu]. \quad (7.38)$$

7.3.4 Display

Real display devices are difficult to precisely characterize and characterization errors can confound research results. Results for different displays or even the same display device with different settings may be quite different. Controlled adjustments to test a variety of display parameters are difficult. Simulated display devices are accurate, precise, flexible, and portable. A simulated display image can be exactly compared to the desired result (e.g., the digital scene). Algorithms can be easily tested for a variety of simulated display devices. Software simulations of display devices are portable.

Displayed images are spatially continuous and as such cannot be directly represented on a digital computer. Continuous displays can be simulated using the same structures used to simulate continuous scenes: Fourier series and super-resolution digital images. Display devices are accurately modeled as LSI systems. A super-resolution display simulation is calculated by spatial convolution $r = d * p$ defined by

$$r\left(\frac{m}{D_M}, \frac{n}{D_N}\right) = \frac{1}{M_D N_D} \sum_{m'=0}^{M-1} \sum_{n'=0}^{N-1} p[m', n'] d\left(\frac{m}{D_M} - m', \frac{n}{D_N} - n'\right) \quad (7.39)$$

where d is the super-resolution array (D_M and D_N are the display super-resolution ratios) that models the PSF of the display device. Using the same super-resolution ratio for the display device as for the scene (so that the array sizes are identical— $M_d = M_s$ and $N_d = N_s$) simplifies comparisons between the scene and the result, but the display device should be accurately modeled. The displayed image can also be calculated in the frequency domain as $\hat{r} = \hat{d}\hat{p}$ given by

$$\hat{r}[\mu, \nu] = \hat{p}[\mu, \nu] \hat{d}(\mu, \nu) \quad (7.40)$$

where \hat{d} is the two-dimensional complex Fourier series modeling the transfer function of the display.

The same comments about computational complexity that were made in Section 7.3.1 are appropriate here. A small display PSF and a large image favors convolution; otherwise, frequency-domain processing is more efficient. In the simulations of Chapter 8, frequency domain processing was used to provide for a display PSF as large as the image.

References

- [1] Robert M. Haralick. Computer vision theory: The lack thereof. *Computer Vision, Graphics, and Image Processing*, 36:372–386, 1986.
- [2] Keith Price. Anything you can do, I can do better (no you can't)... *Computer Vision, Graphics, and Image Processing*, 36:387–390, 1986.
- [3] Stephen K. Park. *Discrete Linear Systems*. 1988. Unpublished Lecture Notes (Computer Science Department, College of William and Mary).
- [4] Stephen E. Reichenbach and Stephen K. Park. Computer generated scenes and simulated imaging. In *Technical Digest of the Optical Society of America Annual Meeting*, page 170, 1988.
- [5] Stephen K. Park and Robert A. Schowengerdt. Image sampling, reconstruction, and the effect of sample-scene phasing. *Applied Optics*, 21(17):3142–3151, 1982.
- [6] Stephen K. Park, Robert A. Schowengerdt, and Mary-Anne Kaczynski. Modulation-transfer-function analysis for sampled image systems. *Applied Optics*, 23(15):2572–2582, 1984.
- [7] Athansios Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, New York, NY, 1984.
- [8] Richard A. Roberts and Clifford T. Mullis. *Digital Signal Processing*. Addison-Wesley, Reading, MA, 1987.
- [9] Robert A. Gonsalves, Thomas A. Lianza, and Andrew Masia. Generation of random scenes with controlled statistics. In *Image Understanding Systems II*, pages 146–152, SPIE, 1979.
- [10] Stephen K. Park and Keith W. Miller. Random number generators: Good ones are hard to find. *Communications of the ACM*, 31(10), 1988.
- [11] Ray Schmidt. *The USC-Image Processing Institute Data Base*. Technical Report 780, University of Southern California Image Processing Institute, October 1977.
- [12] H. C. Andrews and B. R. Hunt. *Digital Image Restoration*. Prentice-Hall, Englewood Cliffs, NJ, 1977.
- [13] Carl L. Fales, Friedrich O. Huck, Judith A. McCormick, and Stephen K. Park. Wiener restoration of sampled image data: end-to-end analysis. *Journal of the Optical Society of America A*, 5(3):300–314, 1988.

- [14] J. W. Modestino and R. W. Fries. Edge detection in noisy images using recursive digital filtering. *Computer Graphics and Image Processing*, 6:409-433, 1977.
- [15] Stephen E. Reichenbach, John C. Burton, and Keith W. Miller. A comparison of algorithms for computing the two-dimensional discrete Hartley transform. *Journal of the Optical Society of America A*, 6(6):818-823, 1989.
- [16] C. B. Johnson. A method for characterizing electro-optical device modulation transfer functions. *Photographic Science and Engineering*, 14(6):413-415, 1970.
- [17] Donald E. Knuth. *The Art of Computer Programming: Seminumerical Algorithms*. Volume 2, Addison-Wesley, Reading, MA, second edition, 1981.

Chapter 8

Experimental Results

8.1 One-Dimensional Simulation Results

This section presents experimental restoration results for one-dimensional artificial scenes degraded by simulated imaging devices (as described in Chapter 7). The experiment design included three variables: the smoothness of the scene, the width of the acquisition transfer function, and the noise level. Three levels for each variable were considered, producing a total of 27 experimental restoration problems. For each of the 27 experiments, 32 specific images were generated. Each of the images was restored with kernels constrained to a number of sizes. The mean-square restoration error (MSRE) of the constrained restorations was compared to the mean-square error of the unrestored display and the MSRE of the end-to-end Wiener restoration. The results demonstrate that small kernels can substantially restore an image in a variety of situations.

One-dimensional Fourier scenes were generated as described in Chapter 7 by specifying the spectral magnitude and randomizing the phase. The spectral magnitude \hat{s}_ρ was set to

$$\hat{s}_\rho[\nu] = \begin{cases} K \exp\left(-(|\nu|/\alpha_s)^{\beta_s}\right) & \text{if } 0 < |\nu| < 2N \\ 0 & \text{otherwise} \end{cases} \quad (8.1)$$

with $N = 256$. The scene magnitude is zero at the origin ($\hat{s}_\rho[0] = 0$), so the resulting ensemble of scenes is zero-mean. The constant K was defined so that the scenes had unit RMS energy ($\gamma_s = 1$). Three distinct ensembles with different degrees of smoothness were considered. In all three $\alpha_s = N/16$; only β_s was varied:

- With $\beta_s = 1.50$ (and $K = 0.0581392$), the scenes are relatively smooth with little high-frequency energy.

- With $\beta_s = 0.75$ (and $K = 0.0704946$), the scenes are somewhat jagged with a moderate amount of high-frequency energy.
- With $\beta_s = 0.50$ (and $K = 0.0658505$), the scenes are more jagged with substantial high-frequency energy.

The power spectra and corresponding autocorrelation functions of these three ensembles are pictured in Figure 8.1 ($N = 256$). Representative scenes from each of the three ensembles are shown in Figure 8.2.

The three simulated image acquisition transfer functions and the corresponding PSFs are pictured in Figure 8.3. The model of the device transfer functions was suggested by Johnson[1]:

$$\hat{h}[\nu] = \exp\left(-(|\nu|/\alpha_h)^{\beta_h}\right). \quad (8.2)$$

All three of the transfer functions in this section are Gaussian bell curves ($\beta_h = 2$).

- With $\alpha_h = 0.75$, the transfer function roll-off is mostly above the Nyquist frequency. This function attenuates frequency components within the Nyquist limit only slightly and will therefore cause little blurring. However, the transfer function significantly passes components above the Nyquist limit and the system is therefore vulnerable to aliasing.
- With $\alpha_h = 0.50$, the transfer function rolls off at a lower frequency and therefore causes somewhat more blurring, but the system is less vulnerable to aliasing.
- With $\alpha_h = 0.25$, the transfer function is nearly zero beyond the Nyquist frequency, virtually eliminating aliasing, but causing substantial blurring.

Three levels of additive noise were used. Signal-to-noise ratio (SNR) is the ratio of root-mean-square (RMS) energy of the scene to RMS energy of the noise:

$$\begin{aligned} \text{SNR} &\triangleq \frac{\gamma_s}{\gamma_e} \\ &= \sqrt{\frac{\sum_{\nu} \hat{s}_{\rho}[\nu]^2}{\sum_{\nu} \hat{e}_{\rho}[\nu]^2}}. \end{aligned} \quad (8.3)$$

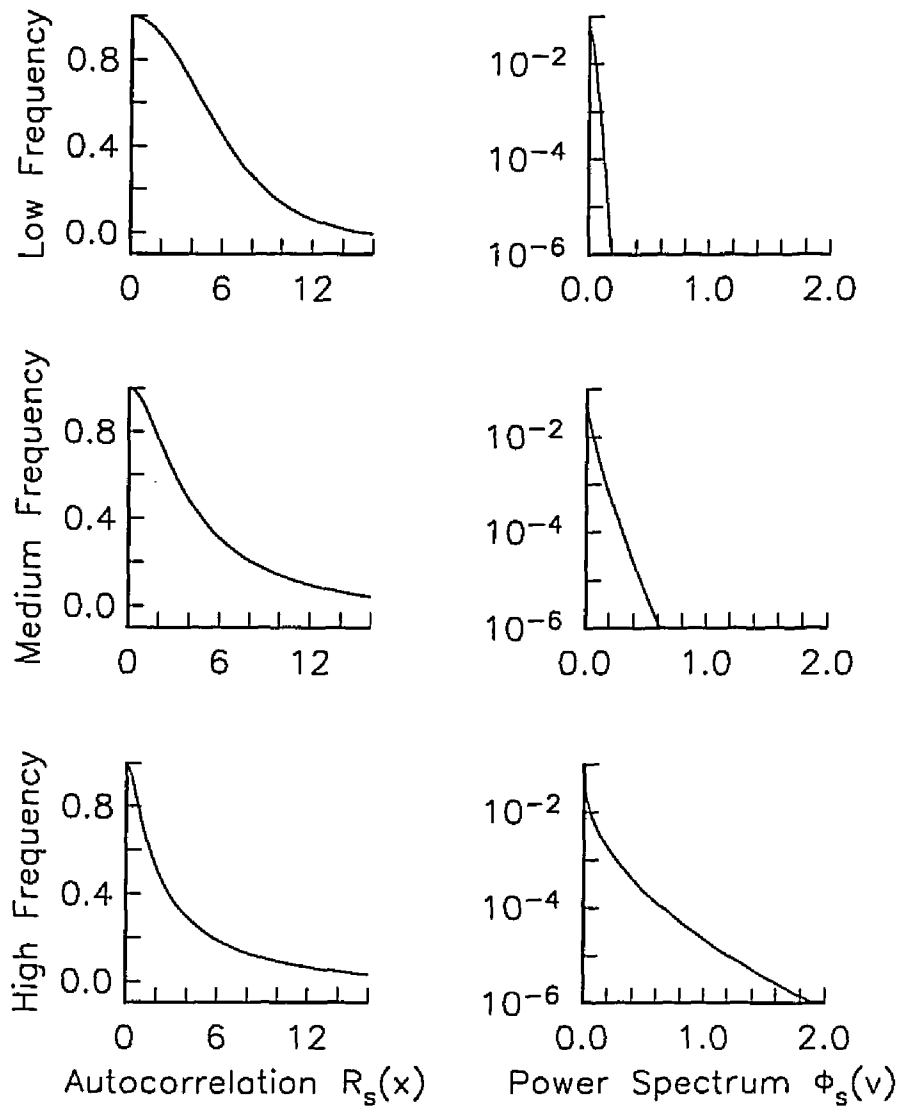


Figure 8.1: Artificial Scene Statistics

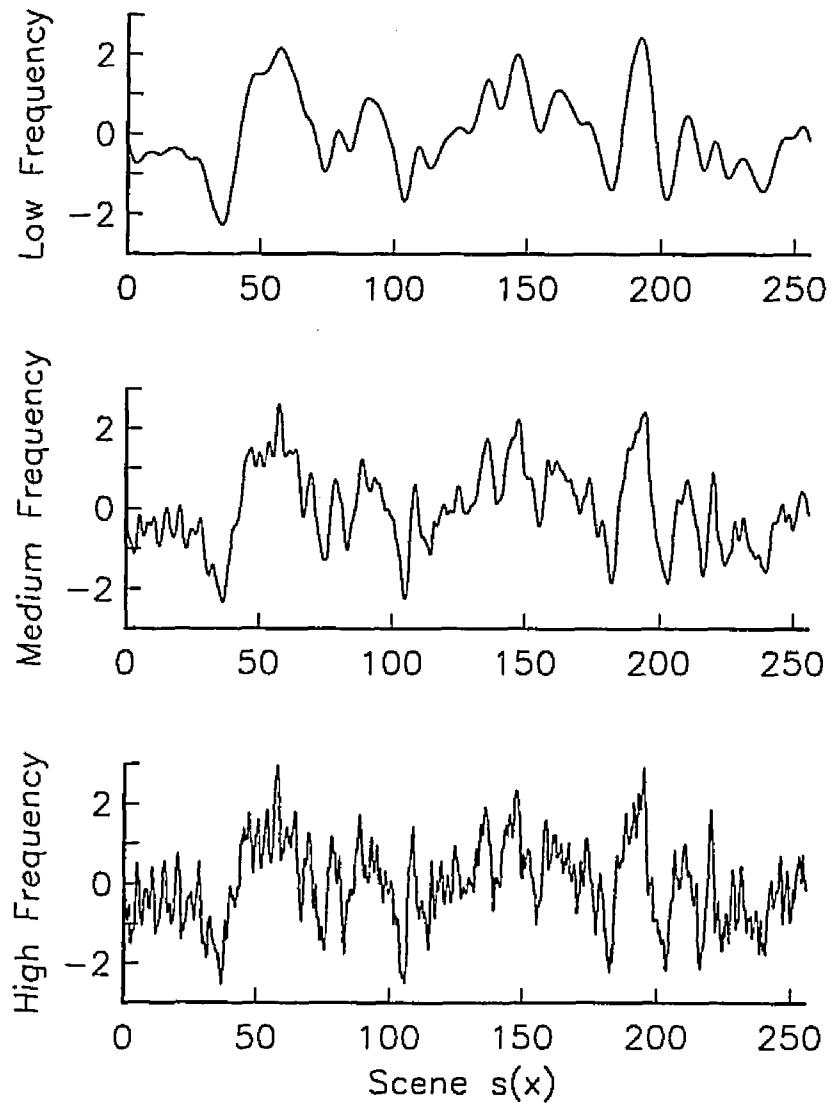


Figure 8.2: Representative Artificial Scenes

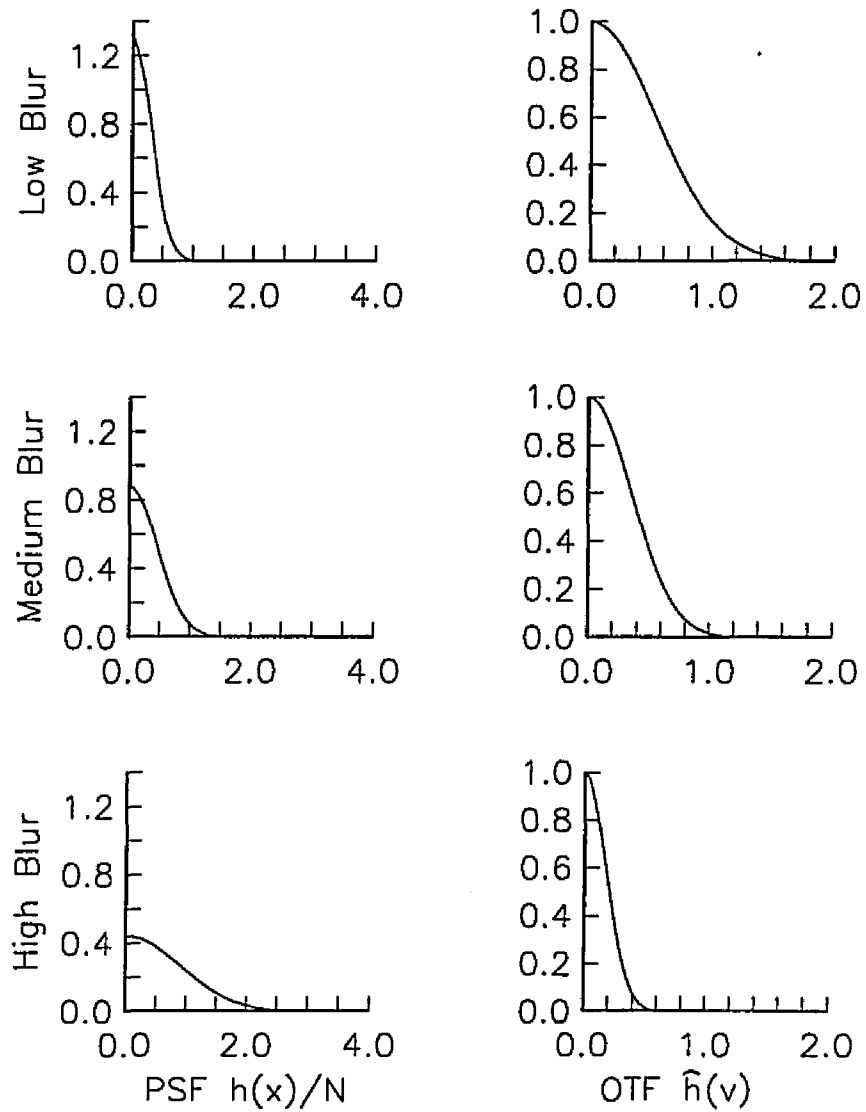


Figure 8.3: Simulated Acquisition Functions

In this simulation, the scenes are normalized to have unit RMS energy ($\gamma_s = 1$), so the SNR is equal to the reciprocal of the RMS energy of the noise. The spectral magnitude \hat{e}_ρ of white noise is constant. Therefore, the spectral magnitude of the additive white noise for a given SNR is

$$\hat{e}_\rho[\nu] = (\text{SNR}\sqrt{N})^{-1} \quad (\nu \neq 0). \quad (8.4)$$

(Because $\hat{e}_\rho[0] = 0$, the noise is zero-mean.)

- For the low-noise images (high SNR), SNR=100.
- For the moderate-noise images, SNR=25.
- For the high-noise images (low SNR), SNR=5.

Figure 8.4 illustrates representative sensor noise.

Real display devices are a significant component of the end-to-end imaging process, but are not usually a source of much variability. Therefore, the simulated display function was not varied in these experiments—a single display model was used for all of the simulations. Schade[2] suggested a display model consisting of the sum of two Gaussians—the *nucleus*, a strongly-peaked central spot that contains most of the energy, and a broad *flare* spot around the nucleus. The composite display transfer function is

$$\hat{d}[\nu] = D_1 \exp\left(-(|\nu|/\alpha_1)^2\right) + D_2 \exp\left(-(|\nu|/\alpha_2)^2\right). \quad (8.5)$$

The parameters for the functions are taken from Schade's results: for the nucleus, $D_1 = 0.76$ and $\alpha_1 = 0.4301484$; for the flare, $D_2 = 0.24$ and $\alpha_2 = 0.0323814$. For practical reasons, the display transfer function is cutoff at twice the sampling rate $\pm 2N$ (the same length as the Fourier series used to generate the artificial scenes). The effect of the truncation is insignificant. The nucleus, flare, and composite display spot and their transfer functions are illustrated in Figure 8.5.

All of the components of the simulation have now been described. Figure 8.6 illustrates the end-to-end imaging simulation for a representative scene from the medium frequency ensemble. The top graph is the scene ($\alpha_s = 0.75$). Directly below it is the image created by applying the acquisition function with medium blur ($\alpha_h = 0.50$) to the scene. The third graph is the sampled image. Next is the sampled scene plus moderate noise (SNR = 25). The bottom graph of Figure 8.6 shows the unrestored displayed image. Acquisition blurring, aliasing due to sampling, additive sensor noise, and display degradation are all present in the output of the system. The net effect of all of these degradations is primarily a loss of high frequencies. The

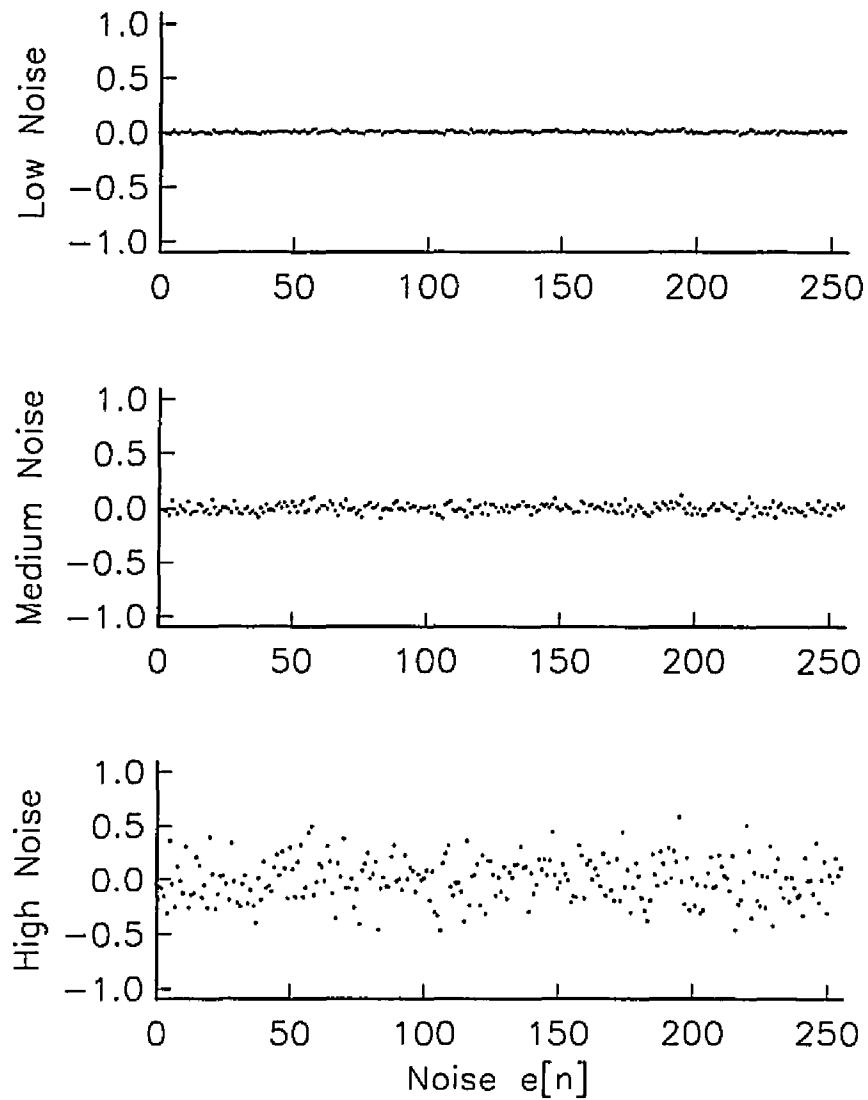


Figure 8.4: Representative Sensor Noise

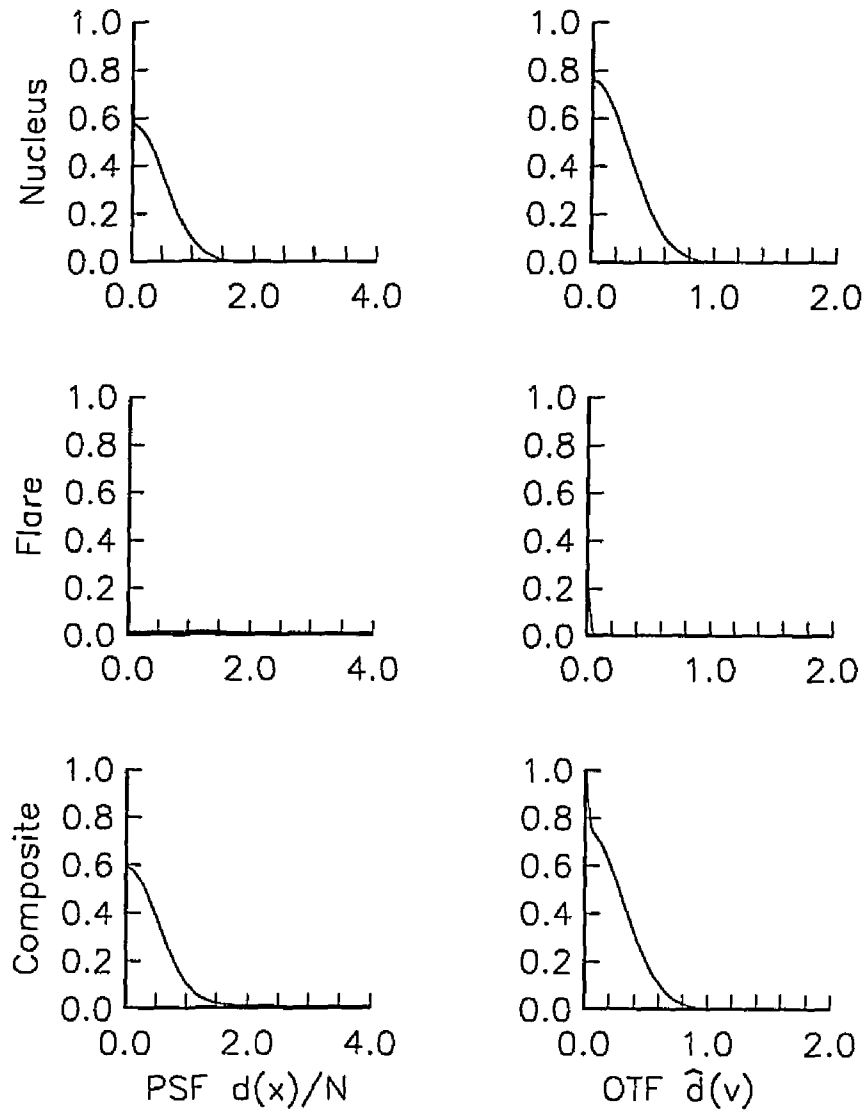


Figure 8.5: Simulated Display Functions

goal of restoration is to process the noisy digital image (the fourth graph) so that when it is displayed, the output (the bottom graph) is a more accurate representation of the input (the top graph).

The optimal, spatially-constrained restoration kernels and the end-to-end Wiener filter (that accounts for sampling and display) were calculated from the power spectra of the scene and noise, and the image acquisition and display transfer functions. The spatial kernels were constrained to have zero value at all but an odd number of locations centered at the origin—the smallest kernel, with three elements, was allowed non-zero values only where $|n| \leq 1$; the next smallest, with five elements, was allowed non-zero values only where $|n| \leq 2$; and so on. The largest constrained kernel has $(N - 1)$ elements; only the element at $n = N/2$ was constrained to 0. The next-largest optimal kernel (no elements constrained to 0) is the spatial kernel of the Wiener filter.

The optimal three-point and five-point kernels for the example of Figure 8.6 and the corresponding transfer functions are shown in Figure 8.7. The Wiener filter transfer function and part of the corresponding spatial kernel are also illustrated. Only the first few elements of the Wiener kernel are shown; the magnitude of the Wiener kernel elements beyond 6 pixels from the origin is less than $0.01N$. Clearly, the optimal small kernels are quite different than the kernels produced by truncating the Wiener PSF. As can be seen by comparing the transfer functions, the optimal three-point kernel does a fair job of approximating the Wiener filter at low frequencies but amplifies high-frequency components where SNR is lower much more than does the Wiener filter. The transfer function of the optimal five-point kernel more closely approximates the Wiener filter, but is still quite different.

Note that the magnitude of the small kernels' transfer function at the origin is greater than zero (about 1.09 for the three-point kernel and 1.08 for the five-point kernel). The artificial scene in this simulation is a zero-mean process. The acquisition function blurred the scene and decreased contrast. Increasing the contrast of the displayed image will decrease mean-square error. Pointwise multiplication by a constant greater than one increases the contrast of a zero-mean process. Therefore, even a one-point kernel can slightly restore a zero-mean image (i.e., reduce the mean-square error). Contrast stretching for an image that does not have a zero mean requires addition of a constant to all pixels, a non-linear operation. This issue is considered more fully in the next section where the two-dimensional scenes do not have zero mean.

Figure 8.8 shows the original scene, the unrestored output, the output with three-point restoration, the output with five-point restoration, and the output with Wiener restoration. Visual comparison is a subjective process,

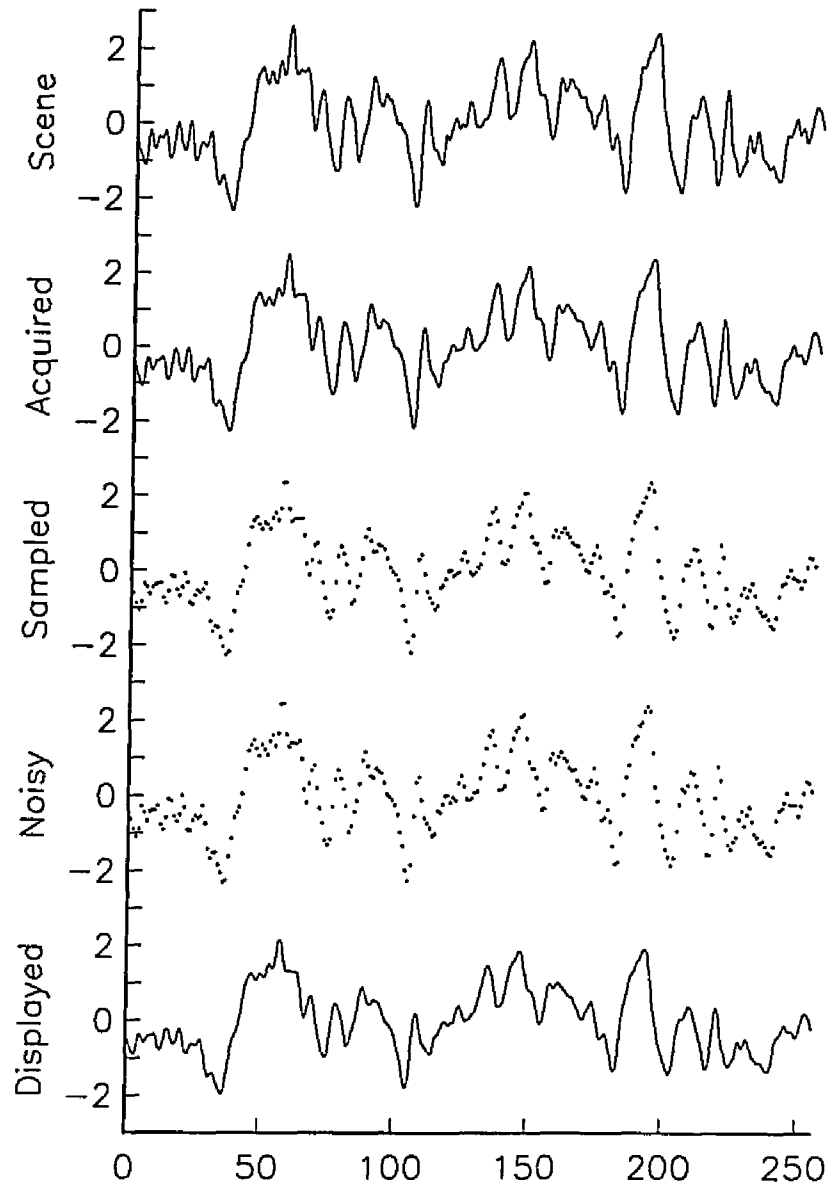


Figure 8.6: Simulated End-to-End Processing of a Representative Scene

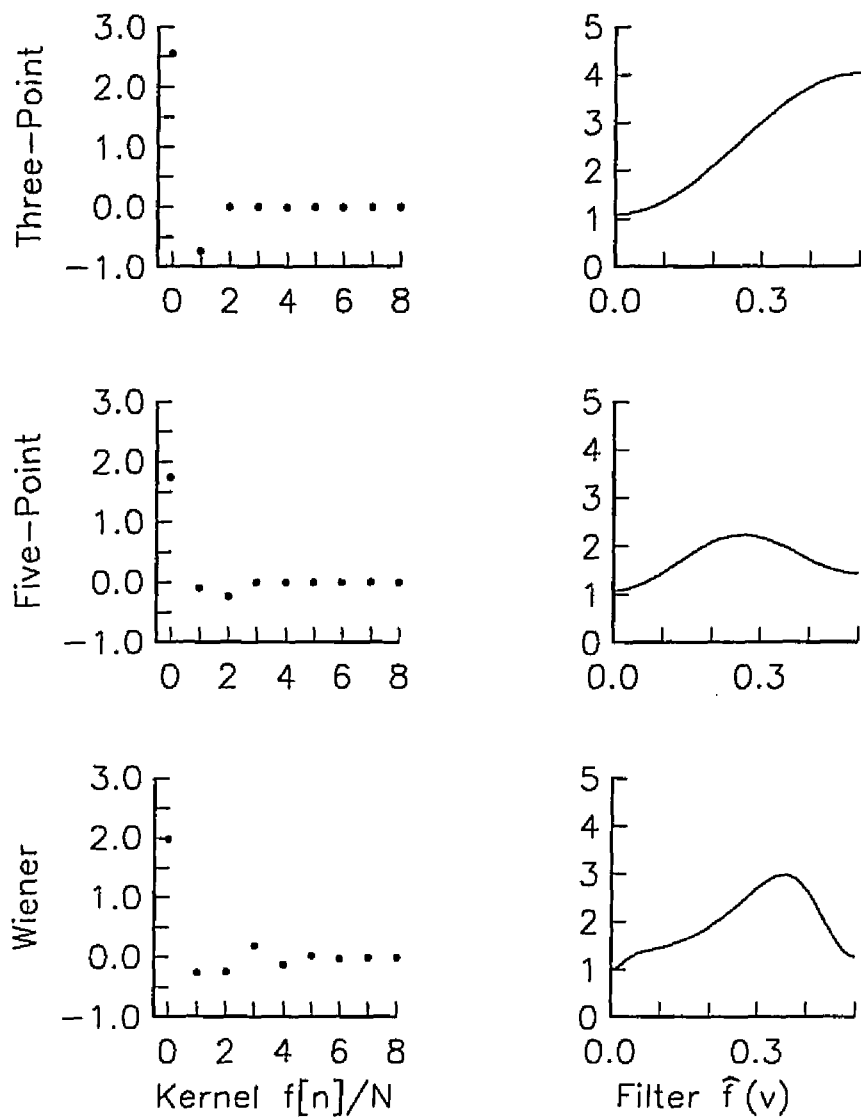


Figure 8.7: Restoration Functions

but it is clear that all of the restored images are more like the original scene than the unrestored image. It is more difficult to conclude from visual inspection which of the restorations is the best.

Figures 8.9, 8.10, and 8.11 present numeric error measures as a function of kernel size. The relative MSRE is the ratio of the RMS difference between the displayed image and the scene to the RMS energy of the scene:

$$\text{Relative RMS Error} = \sqrt{\frac{\sum_{\nu} |\hat{s}[\nu] - \hat{r}[\nu]|^2}{\sum_{\nu} |\hat{s}[\nu]|^2}}. \quad (8.6)$$

Each of the 27 restoration experiments was performed 32 times—that is, each execution used a different scene from the ensemble and different random noise. The plots show the relative RMS error averaged over all 32 executions. The standard deviations of the relative RMS error were so small that plotting them on these graphs proved impractical. The plots are shown only for kernels with 65 elements or fewer (radius 32). In all cases, only negligible improvement occurred beyond 19 elements (radius 9). The solid lines show the relative RMS error for the Wiener restoration. (The kernel of the Wiener filter has 255 elements, a radius of 128.)

In many cases, the three-point and five-point kernels yielded results that are nearly as accurate as the Wiener filter. This is particularly true when there is little noise (e.g., SNR=100—the leftmost column in Figures 8.9, 8.10, and 8.11) and/or there is significant high-frequency energy in the scene (e.g., $\beta_s = 0.50$ —Figure 8.11). Small kernels are relatively less successful in low SNR situations (e.g., SNR=5—the rightmost column in Figures 8.9, 8.10, and 8.11). In low SNR problems, the restoration kernel should suppress noise by local averaging, but small kernels are restricted in doing so by their size. Low-SNR images are difficult to restore—the inverse filter performs very badly and the Wiener filter has only limited success. Imaging system design changes can increase SNR, for example, with a slower shutter speed, pre-sample filtering, decreased spatial resolution, and increased brightness resolution.

In the experiment with a medium-frequency scene, moderate blur, and moderate noise ($\beta_s = 0.75$, $\alpha_h = 0.50$, and SNR=25)—the middle graph of Figure 8.10—the average unrestored relative RMS error was 0.204613. The Wiener filter reduced this to 0.051149, a decrease of 75%. The three-point kernel resulted in a relative RMS error of 0.091685, a decrease of 55%. The three-point kernel (radius 1) achieved 73% of the improvement of the Wiener filter. The five-point kernel (radius 2) reduced the relative RMS error to 0.083614, a decrease of 59%. This is 79% of the improvement of the Wiener

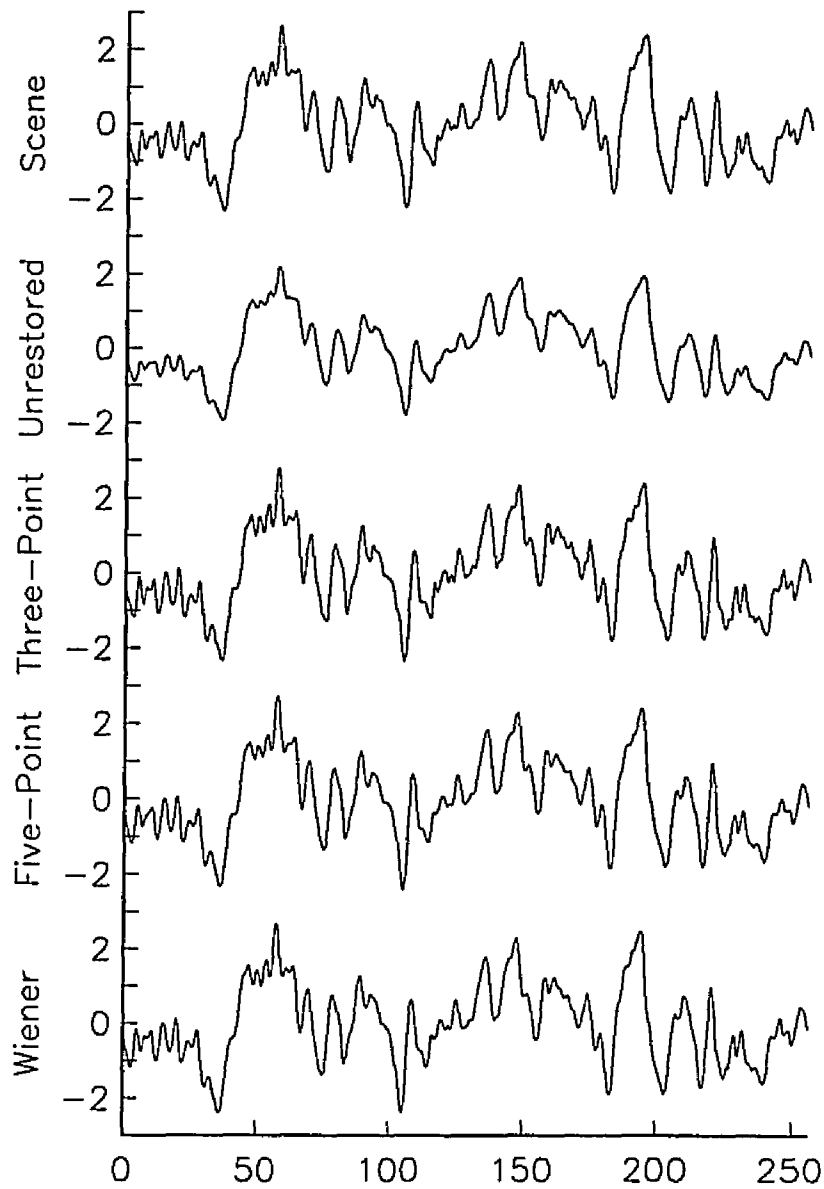


Figure 8.8: Representative Restoration Results

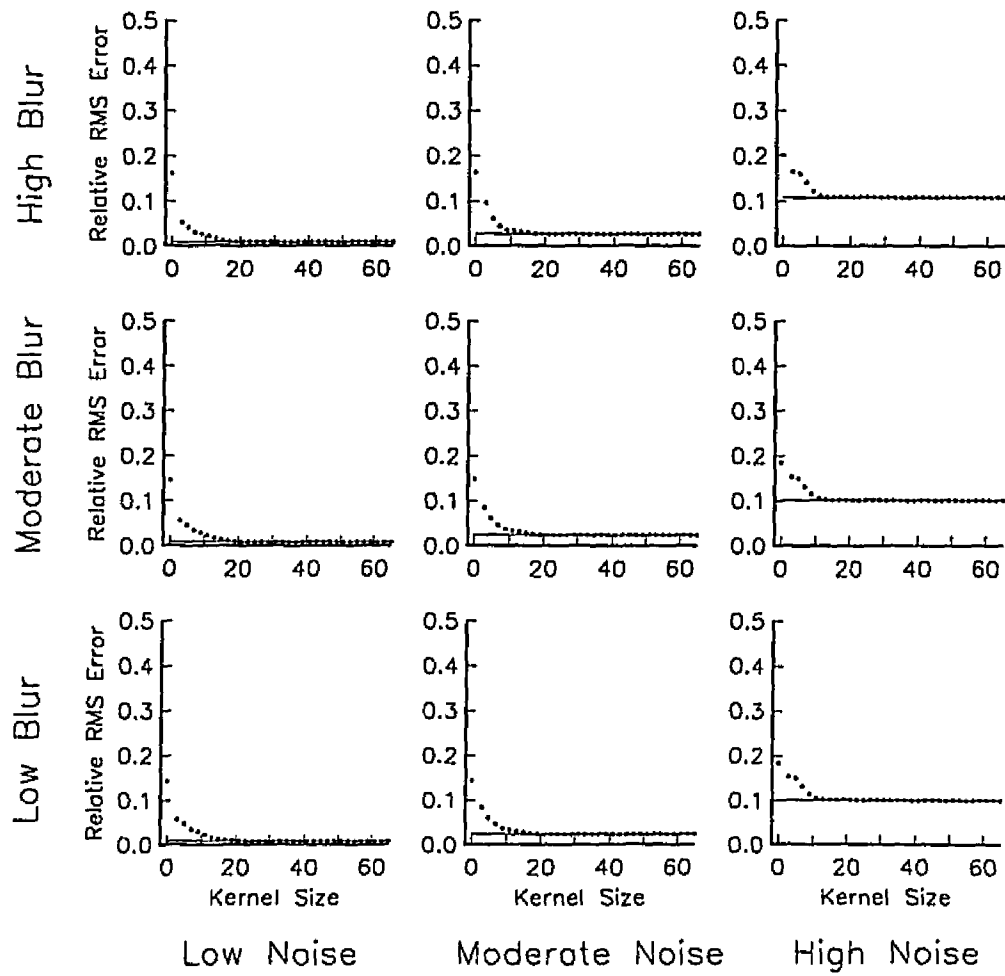


Figure 8.9: Relative Restoration Error (Low Frequency Scenes)

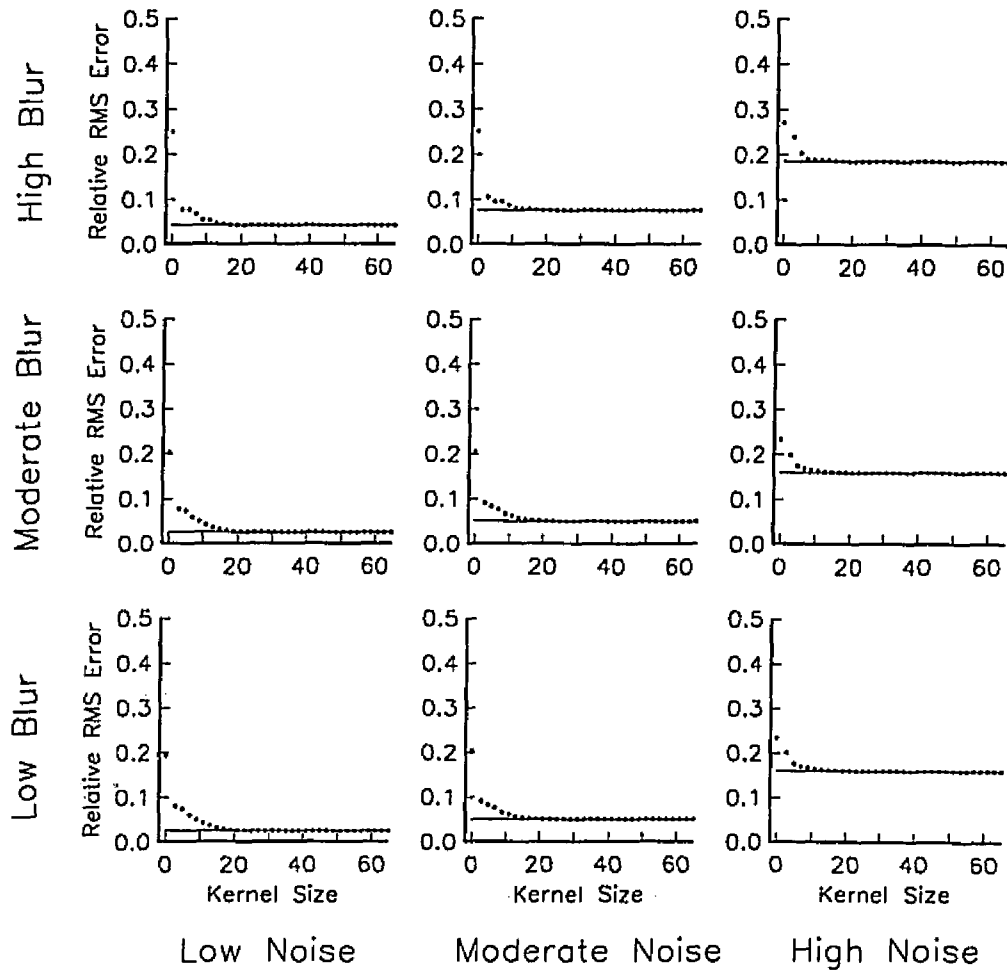


Figure 8.10: Relative Restoration Error (Medium Frequency Scenes)

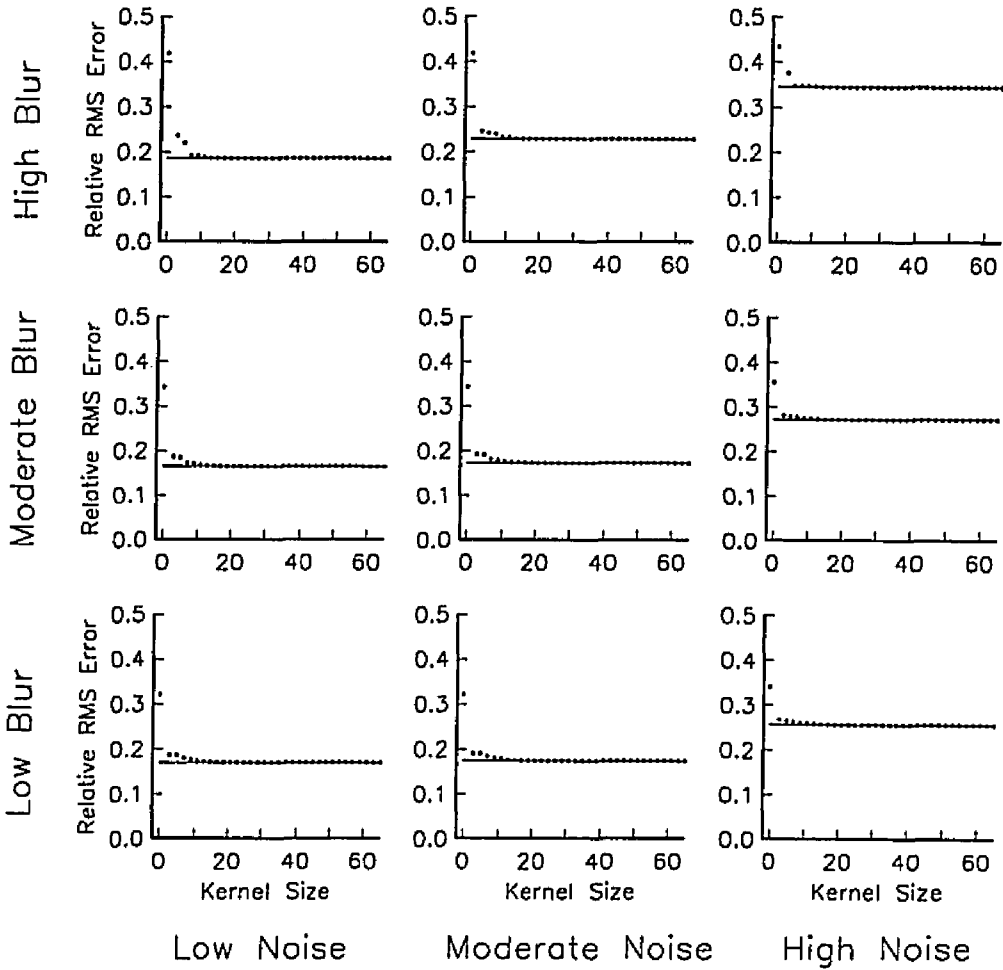


Figure 8.11: Relative Restoration Error (High Frequency Scenes)

filter. These small kernels achieve a large portion of the improvement of the Wiener filter with substantially less computation.

8.2 Two-Dimensional Simulation Results

This section presents two-dimensional simulation results. The two-dimensional simulation parallels the one-dimensional simulation described in Section 8.1. A single digital scene was used for all two-dimensional simulation experiments; the OTF width and the SNR were varied to produce nine experimentals.

The digital scene is a 1024×1024 , 12-bit image of several United States one-dollar bills. This "scene" is familiar and widely available; it has regions with great detail, regions of little detail, edges, and texture. The digital scene was acquired with a Photometrics slow-scan, three-stage, thermoelectric-cooled CCD camera system with a Texas Instruments VP1M array sensor and standard 50mm lens. The target was placed 45cm from the camera, the camera was focused on the target, the f -stop was set to 8, and the exposure time was 0.1 seconds. The brightness values of the image range from 286 to 16383 with a mean \bar{s} of 2926 and contrast σ_s of 1167.

Before the digital scene was subsampled at every fourth pixel, it was blurred by a simulated image acquisition device. The acquisition OTF was modeled as a two-dimensional Gaussian:

$$\hat{h}[\mu, \nu] = \exp\left(-\left(\sqrt{\mu^2 + \nu^2}/\alpha_h\right)^{\beta_h}\right) \quad (8.7)$$

with $\beta_h = 2$ and α_h set to 0.25, 0.50, and 0.75 in different experiments. (One-dimensional cross-sections of the simulated acquisition OTFs and corresponding PSFs are shown in Figure 8.3.)

The blurred digital scene was subsampled at every fourth pixel to produce a 256×256 effectively noiseless digital image. The four-fold superresolution in the digital scene accommodates spectral frequencies to ± 2.0 , ample for simulating aliasing.

Three different levels of additive white noise were used. The noise processes were defined to yield SNRs of 5, 25, and 100 relative to a zero-mean scene with the same contrast (standard deviation) as the actual scene. (The contrast was used in place of energy to control for the non-zero mean of the scene.) The spectral magnitude is of the zero-mean noise is

$$\hat{e}_p[\mu, \nu] = \sigma_s (\text{SNR}\sqrt{MN})^{-1} \quad ([\mu, \nu] \neq [0, 0]). \quad (8.8)$$

			10	9	10			
		8	7	6	7	8		
	8	5	4	3	4	5	8	
10	7	4	2	1	2	4	7	10
9	6	3	1	0	1	3	6	9
10	7	4	2	1	2	4	7	10
	8	5	4	3	4	5	8	
		8	7	6	7	8		
			10	9	10			

Figure 8.12: Two-Dimensional Kernel Elements

The noise in Figure 8.4 scaled by the contrast of the scene (1167) is representative.

The display transfer function used in Section 8.1 was generalized to two dimensions as

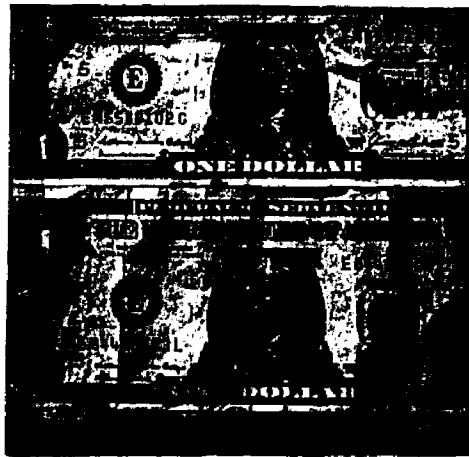
$$\hat{d}[\mu, \nu] = D_1 \exp\left(-\left(\sqrt{\mu^2 + \nu^2}/\alpha_1\right)^2\right) + D_2 \exp\left(-\left(\sqrt{\mu^2 + \nu^2}/\alpha_2\right)^2\right) \quad (8.9)$$

with $D_1 = 0.76$, $\alpha_1 = 0.4301484$, $D_2 = 0.24$, and $\alpha_2 = 0.0323814$. (Figure 8.5 shows a cross-section of the OTF and PSF of the simulated display device.)

Ten different-sized restoration kernels were considered. The smallest was constrained to 5 elements, the center element and its four nearest neighbors. The next smallest 9 elements, 3×3 . Figure 8.12 shows the order in which new elements are added to form larger kernels. Table 8.1 gives the sizes of the constrained kernels used in the simulation.

The digital scene, the unrestored display, the restored images for the 9-point, 25-point, and 49-point kernels, and the end-to-end Wiener restoration for three of the nine experiments are shown in Figures 8.13, 8.14, and 8.15. Figure 8.13 shows the results for the experiment with low blur ($\alpha_h = 0.75$) and high noise (SNR=5). Figure 8.14 shows the results for the experiment with moderate blur ($\alpha_h = 0.50$) and moderate noise (SNR=25). Figure 8.15 shows the results for the experiment with high blur ($\alpha_h = 0.25$) and low noise (SNR=100).

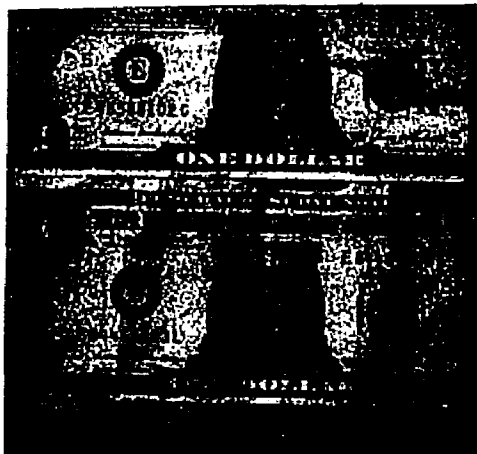
The restoration accuracy as measured by relative RMS error for each of the nine experiments is plotted in Figure 8.16. The relative RMS error is graphed as a function of the restoration kernel size. The solid horizontal line in each graph is the relative RMS error for Wiener restoration. The success of the small kernels follows the pattern of the one-dimensional simulation—the



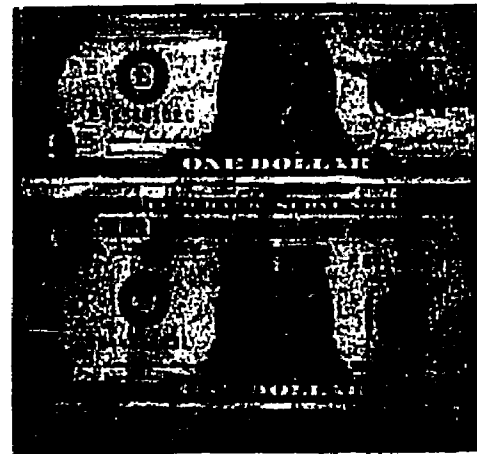
A. Digital Scene



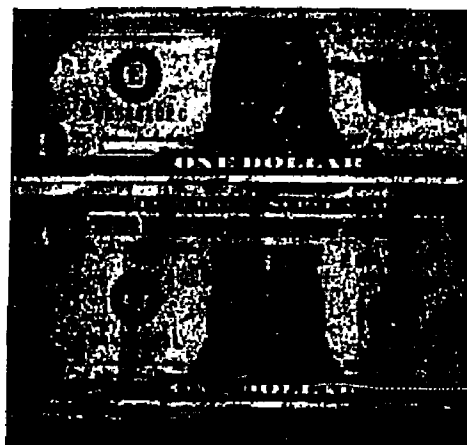
B. Unrestored, Display



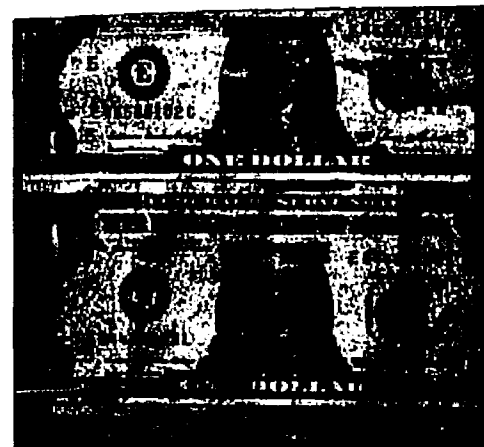
C. 9-Point Restoration



D. 25-Point Restoration



E. 49-Point Restoration

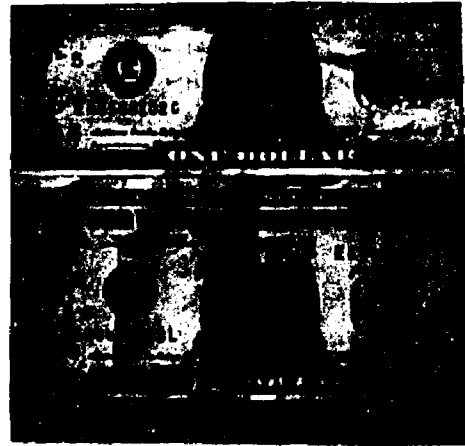


F. Wiener Restoration

Figure 8.13: Restoration for Low Blur and High Noise



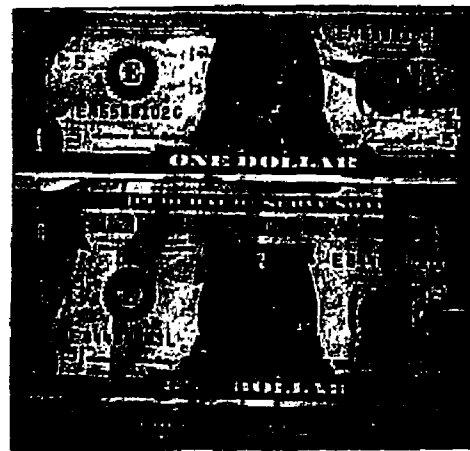
A. Digital Scene



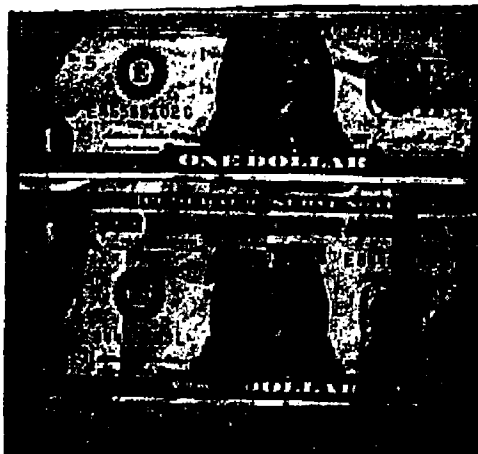
B. Unrestored, Display



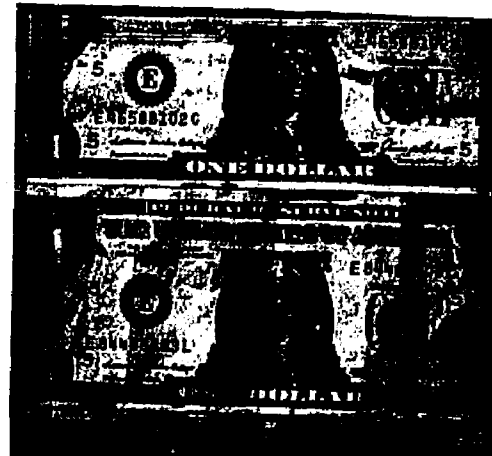
C. 9-Point Restoration



D. 25-Point Restoration

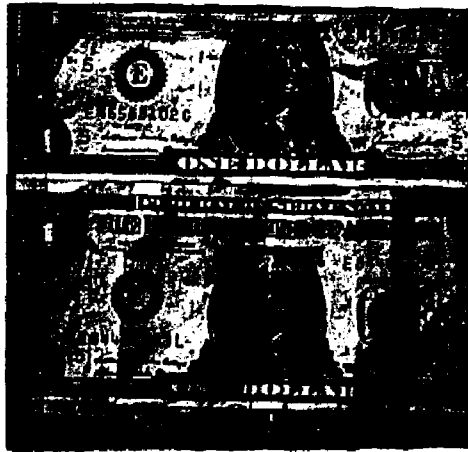


E. 49-Point Restoration



F. Wiener Restoration

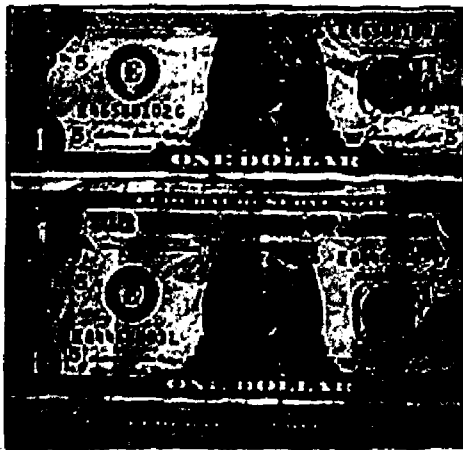
Figure 8.14: Restoration for Moderate Blur and Moderate Noise



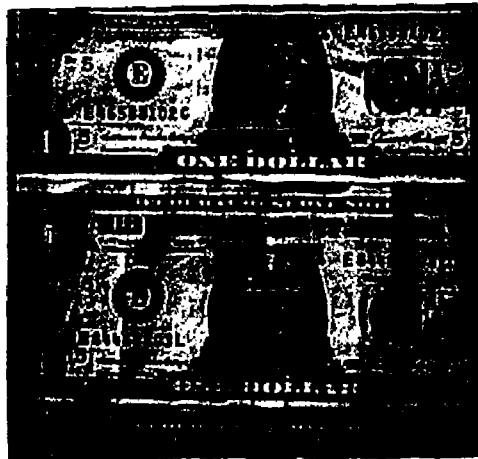
A. Digital Scene



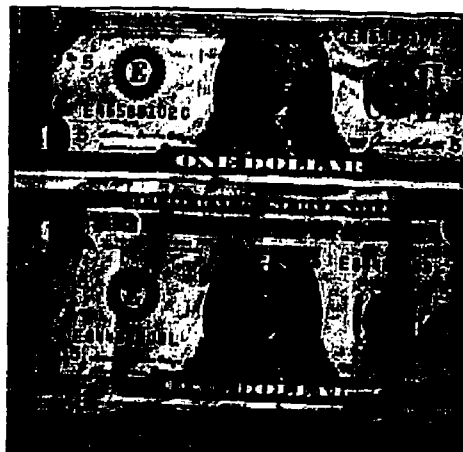
B. Unrestored, Display



C. 9-Point Restoration



D. 25-Point Restoration



E. 49-Point Restoration



F. Wiener Restoration

Figure 8.15: Restoration for High Blur and Low Noise

Number	Kernel Size
1	5
2	9
3	13
4	21
5	25
6	29
7	37
8	45
9	49
10	57

Table 8.1: Two-Dimensional Kernel Sizes

small kernels are more successful in low-noise problems and less successful in higher-noise problems. The high level of relative RMS error in the low-noise, low-blur image indicates that the scene in these problems has significant high-frequency information and was therefore significantly degraded by aliasing and sampling-reconstruction blur[3] during processing.

For the image with moderate blur ($\alpha_h = 0.50$) and moderate noise (SNR=25), the unrestored display had a relative RMS error of 0.466674. The Wiener filter reduced the relative RMS error to 0.340891, a decrease of 27%. The 9-point kernel reduced the relative RMS error to 0.358369, a decrease of 23%. This is 86% of the reduction achieved by the Wiener filter. The 25-point kernel reduced the relative RMS error to .354646, a decrease of 24%. This is 89% of the reduction achieved by the Wiener filter. The 49-point kernel reduced the RMS error to .346512, a decrease of 26%. This is 96% of the reduction achieved by the Wiener filter. The small kernels can be applied more efficiently than the Wiener filter, yet are nearly as effective in reducing RMS error.

The images in the two-dimensional simulation are not zero-mean. In this, they are like real images. It is possible however to create a zero-mean image by subtracting the image mean from each pixel value. As was mentioned in the discussion of the one-dimensional simulation, RMS error in zero-mean images can be reduced somewhat by the simple operation of pointwise multiplication. Small kernels can similarly increase contrast and reduce RMS error in zero-mean images. How much more can RMS error be reduced by such a linear stretch? The same two-dimensional simulation was run but the images were normalized by subtracting out the mean before being restored.

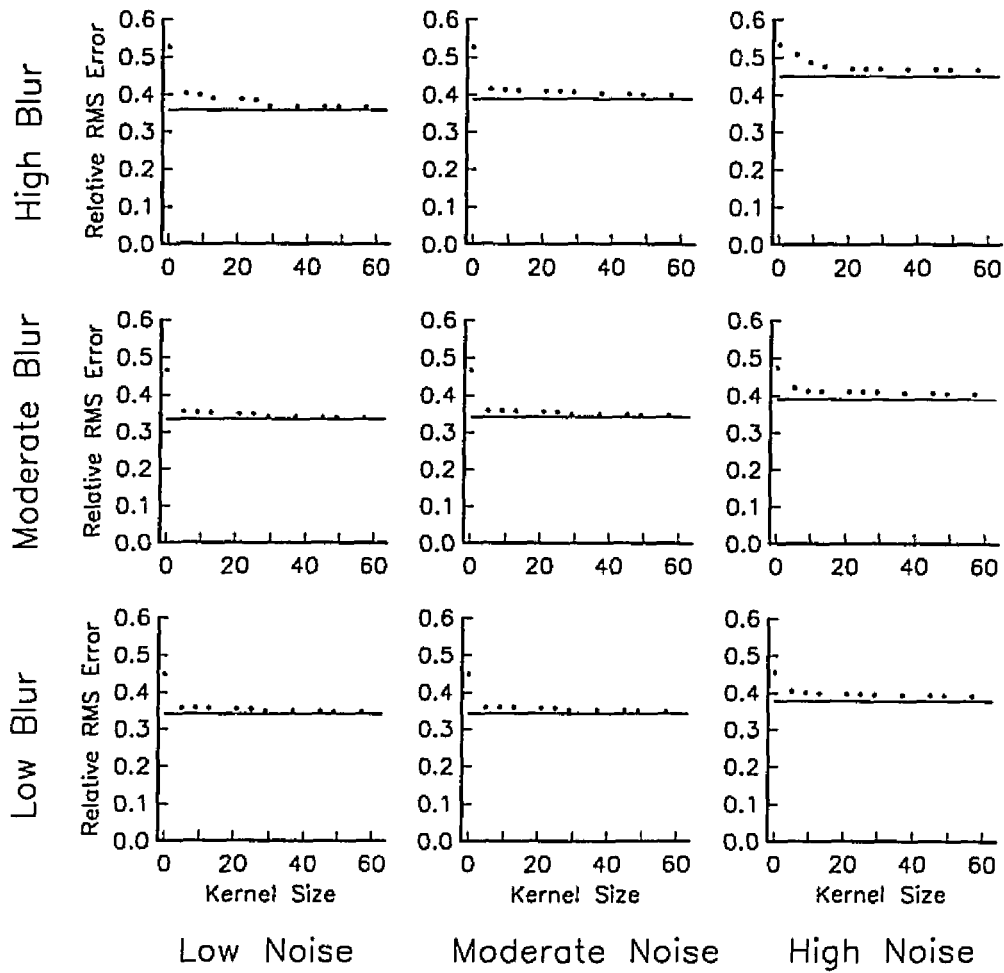


Figure 8.16: Relative Restoration Error

The results indicate that this procedure can result in significant relative improvements but little absolute improvement. For example, for the smallest kernel (5 elements) restoring the worst image (high blur, $\alpha_h = 0.25$, and high noise, SNR=5), the zero-mean restoration performed 38% better than the unnormalized restoration in reducing relative RMS error. However, this kernel is not very effective and the actual improvement in reducing relative RMS error is less than 1%. Overall, the relative gains were far less than 38%, averaging only 10% for all 5-point restorations and 5% for all restorations. The relative gain for 49-point kernels was only 2%. In no cases was the actual reduction in relative RMS error more than 1% and averaged less than 0.4% overall. The relative RMS error plots for normalized restoration are shown in Figure 8.17. Because the relative error is improved no more than 1% in any case, Figures 8.16 and 8.17 are virtually indistinguishable.

8.3 Restoration of Digital Images

This section shows restoration results for unretouched digital images. The target of United States one-dollar bills was photographed using the acquisition system described in Section 8.2. For the images in this section, the bills were placed 180cm from the camera (four times the distance of the target in Section 8.2). The camera f -stop was set to 8 and the exposure time was 0.1 seconds. Three images were acquired: one with the focus at 180cm (i.e., in focus), one with the focus at 135cm (defocus), and one with the focus at 90cm (extreme defocus). As stated above, these images were 1024×1024 with 12-bit pixels. From each of the three images, a 256×256 portion was clipped and restored.

Restoration of a real image is more difficult than restoration of a simulated image. In a real image, the scene and noise power spectra and the acquisition and display transfer functions are not known. Estimating these functions is difficult and estimation errors can be a significant problem. A restoration based on the best estimates that can be made will usually benefit from *ad hoc* adjustments to correct for *over-restoration* (excessive high-pass accentuation or sharpening) or *under-restoration* (an image that is still too blurred). Huck, McCormick et al.[5,4] have illustrated the impact of over-restoration and under-restoration on the visual quality of the result.

The higher-resolution digital image of the target that was used as the digital scene in Section 8.2 provided the basis for estimating the scene power spectrum. This image, taken at 45cm, has four-fold superresolution relative to the images taken at 180cm, so the square of the spectral magnitude (spectral energy) of this single image provides an estimate of the scene power

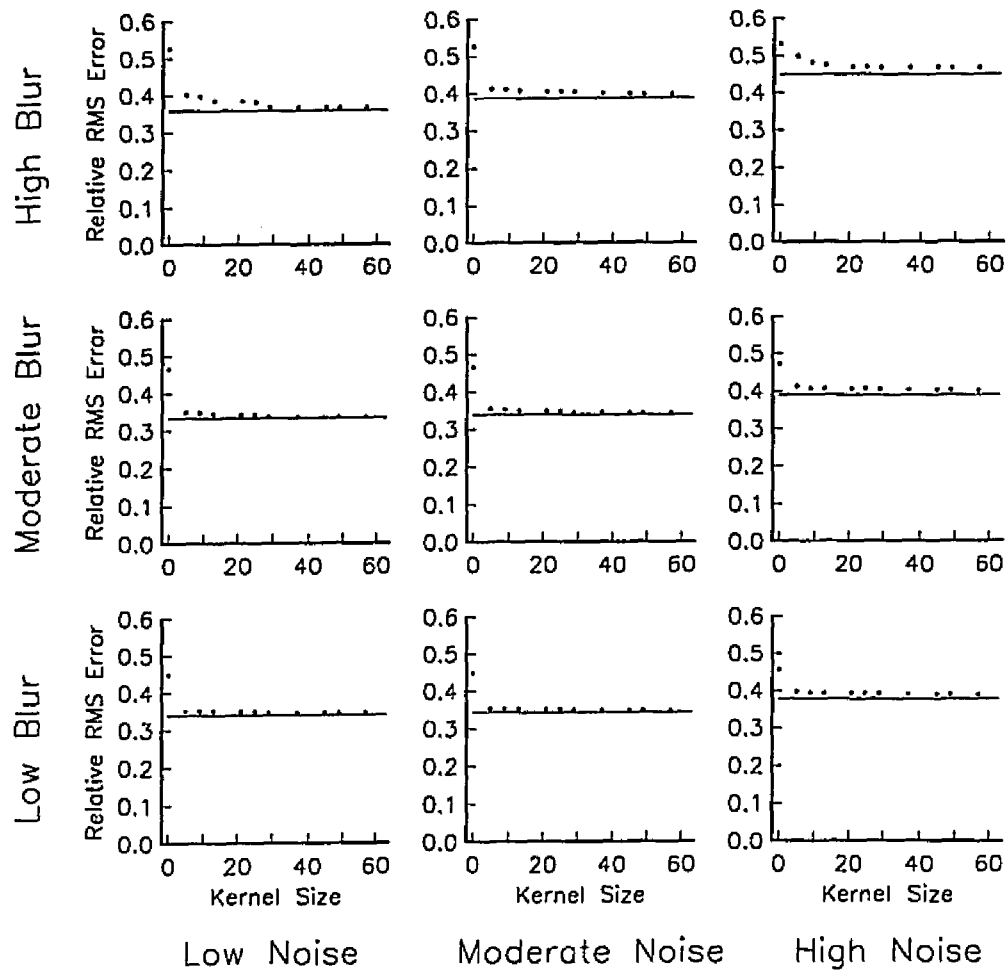


Figure 8.17: Relative Restoration Error (Normalized Images)

spectrum to twice the sampling resolution at 180cm. The spectral energy from a single image is a noisy estimate, so it was smoothed and diminished by a rough estimate of the noise energy.

The acquisition system OTF for each of the three images was characterized using the technique described in Appendix B. The one-dimensional estimates indicated that the system functions were nearly circularly symmetric. Therefore, the average of the horizontal and vertical estimates shown in Figure 8.18 were used to define a radially-symmetric acquisition function estimates.

The noise energy was estimated using the contrast of images of flat fields (targets of a solid color). This analysis indicated that the actual noise in the image acquisition system was not completely signal independent—the contrast of images of a bright field was about three times greater than that of images of a dark field. The contrast of images of the bright field was about one-tenth the contrast of the dollar-bill images. The estimate of the additive, white-noise energy used in deriving the restoration filters was based on the contrast of the bright-field images. Relative to a zero-mean image, this would be a SNR of 10.

The photographs were taken with a Matrix filmwriter. The model of the display device functions described in Sections 8.1 and 8.2 was used as the estimate of the display device function. The spots are placed farther apart than is required for flat-field response[6]. The spot spacing estimate was set to 10/7 of that used in the simulations.

The in-focus image and the 9-point, 25-point, and end-to-end Wiener restorations are shown in Figure 8.19. The end-to-end system is fairly well designed, so it is difficult to significantly improve the image. However, all three restored images are sharper than the unrestored image.

The defocused image (target at 180cm; focus at 135cm) and the 9-point, 25-point, and end-to-end Wiener restorations are shown in Figure 8.20. Viewed from a distance, the restored images are sharper than the unrestored image. However, when viewed closer, the image restored by the Wiener filter has obvious artifacts such as ringing around sharp edges. These artifacts are usually intolerable. The artifacts in the image restored with the 25-point kernel are less severe. The image restored with the 9-point kernel does not have significant artifacts and is sharper than the unrestored image.

The extremely defocused image (target at 180cm; focus at 90cm) and the 9-point, 25-point, and Wiener restorations are shown in Figure 8.21. The artifacts in these restorations are even more pronounced than those in the previous example. In the image restored with the Wiener filter, the sharpened serial numbers and signatures almost blend in with the artifacts.

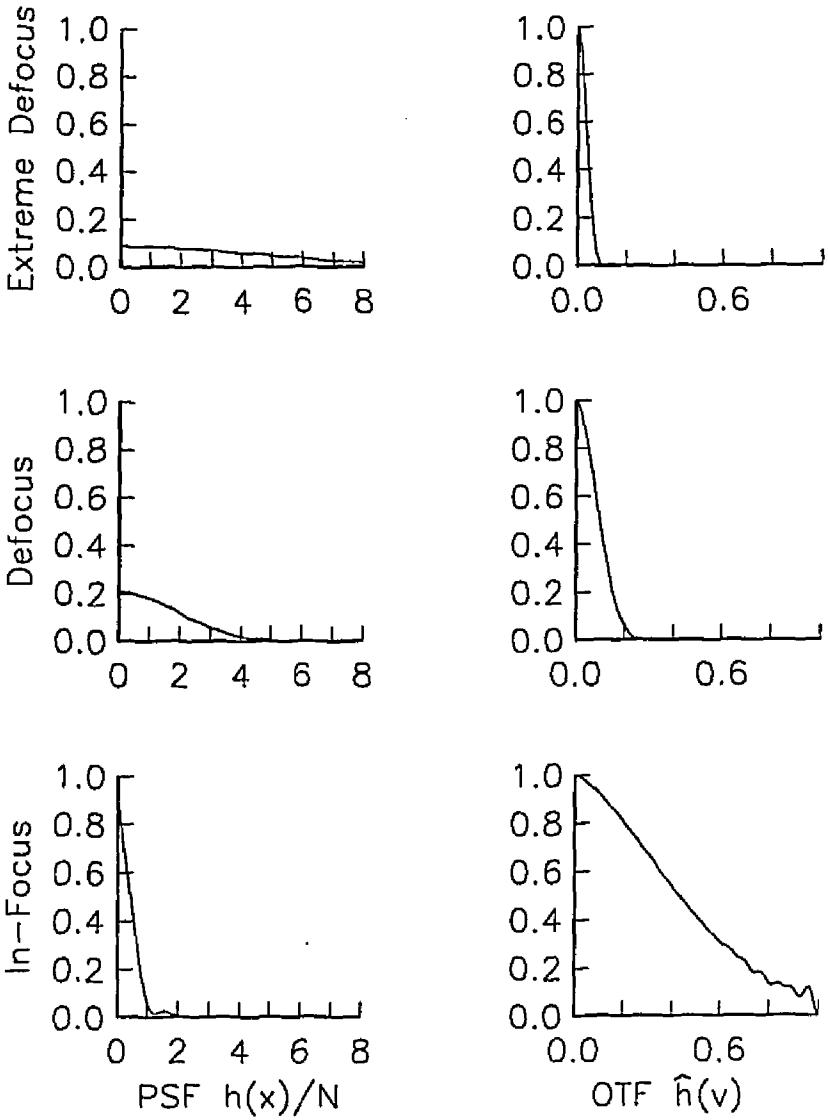
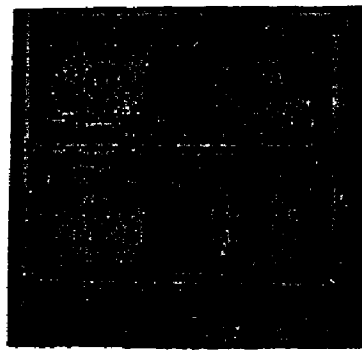
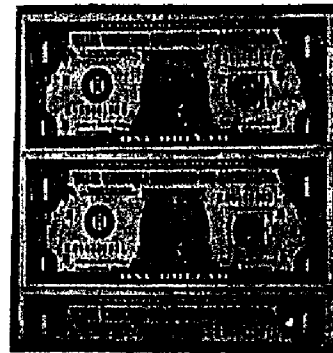


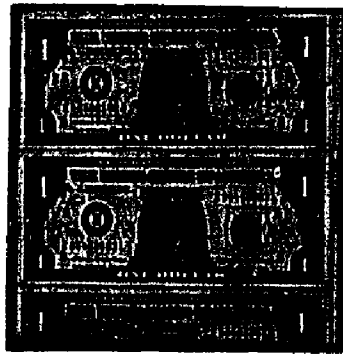
Figure 8.18: Acquisition Functions



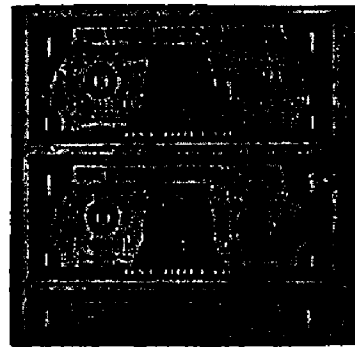
A. Unretouched Image



B. 9-Point Restoration

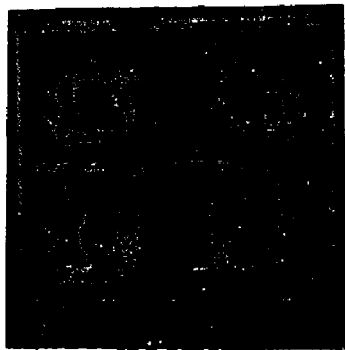


C. 25-Point Restoration

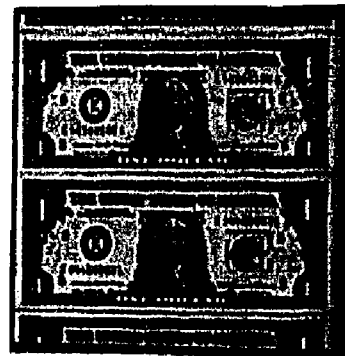


D. Wiener Restoration

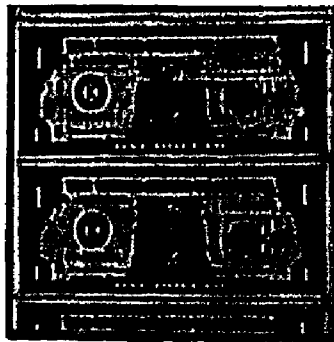
Figure 8.19: Restoration of In-Focus Image



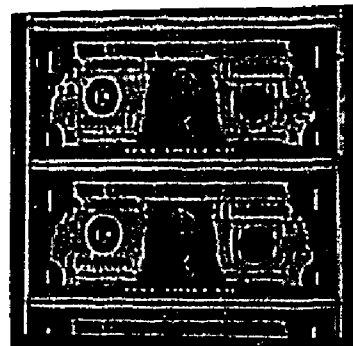
A. Unretouched Image



B. 9-Point Restoration

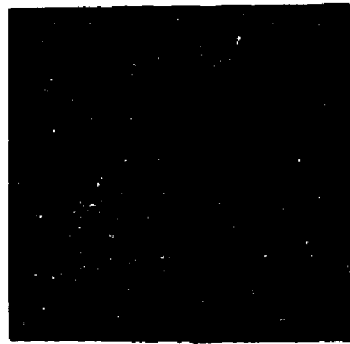


C. 25-Point Restoration

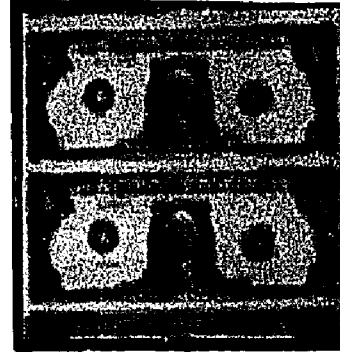


D. Wiener Restoration

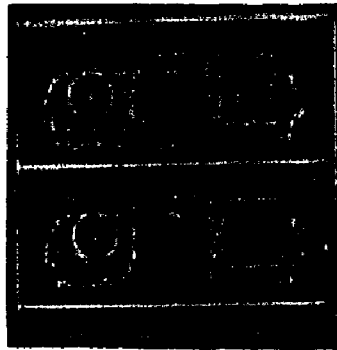
Figure 8.20: Restoration of Defocused Image



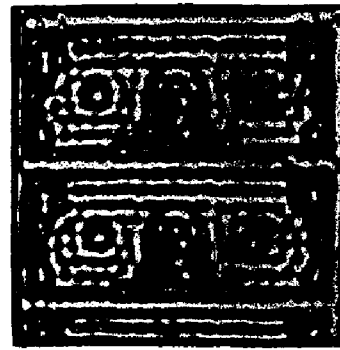
A. Unretouched Image



B. 9-Point Restoration



C. 25-Point Restoration



D. Wiener Restoration

Figure 8.21: Restoration of Extremely Defocused Image

The significant artifacts in the restored images are probably caused by errors in the system estimates—the estimates of the scene and noise power spectra and of the acquisition and display transfer functions. Fewer artifacts are present in the images restored with the small kernels. Possibly, the small kernels are more inherently robust than the Wiener filter. With fewer degrees of freedom, small restoration kernels are less responsive to variations in system estimates, so errors in these estimates have a smaller impact. This hypothesis is supported by the observation that the system of equations that defines the small kernels is better conditioned than the system of equations that defines the Wiener filter. Restoration of real images often has the feel of guessing until you get it right. In the world of real problems, robustness is a significant concern. If small kernels are indeed more robust, it would be a powerful incentive for using them. This is a question that should be answered by more research.

References

- [1] C. B. Johnson. A method for characterizing electro-optical device modulation transfer functions. *Photographic Science and Engineering*, 14(6):413-415, 1970.
- [2] Otto H. Schade, Sr. Image reproduction by a line raster process. In Lucien M. Biberman, editor, *Perception of Displayed Information*, chapter 6, pages 233-278, Plenum Press, New York, NY, 1973.
- [3] Stephen K. Park and Robert A. Schowengerdt. Image sampling, reconstruction, and the effect of sample-scene phasing. *Applied Optics*, 21(17):3142-3151, 1982.
- [4] Judith A. McCormick, Rachel Alter-Gartenberg, and Friedrich O. Huck. Image gathering and restoration: Information and visual quality. *Journal of the Optical Society of America A*, 6(7):987-1005, 1989.
- [5] Friedrich O. Huck and Judith A. McCormick. Image restoration for fidelity and visual quality. Submitted to *Computer Vision, Graphics, and Image Processing*.
- [6] Otto H. Schade. Electro-optical characteristics of television systems: electro-optical specifications for television systems. *RCA Review*, 9:245-286, 1948.

Chapter 9

Conclusions

In 1984, Hunt[1, p. 73] wrote: "If an image restoration problem can be solved, then about 75% of the time it can be treated with some of the simplest techniques, for example, inverse or Wiener filter." Though "solved" is too strong a word, the Wiener filter and other fairly simple restoration techniques are effective. The Wiener filter is often effective even if the underlying assumptions are incorrect (e.g., the noise is signal-dependent or, as in the conventional formulation, sufficient sampling is incorrectly assumed). In applications where processing speed is important, the most significant problem with the inverse filter, Wiener filter, and other Fourier-transform-based techniques is computational cost. High-speed implementation of these global algorithms requires expensive, special-purpose hardware. Restoration implemented by convolution with small kernels requires significantly less computation than frequency-domain processing. Moreover, small-kernel convolution is a local operation that can be implemented in parallel on relatively inexpensive hardware.

The technique described in this dissertation allows explicit spatial constraints on the restoration kernel to be matched to the processing implementation. Within these constraints, the small kernels are designed to yield optimal fidelity (i.e., expected mean-square restoration error is minimized). Fidelity is not a perfect measure for restoration quality, but the popularity of the Wiener filter attests to its utility. The derivation of the small kernel follows the standard frequency-domain derivation of the Wiener filter. The resulting linear system of equations defines the optimal spatial kernel. The development accounts for the significant components of digital imaging systems—the scene, acquisition blur, sampling, noise, and display reconstruction.

The optimal small kernels are effective. The simulated and unretouched restorations presented in Chapter 8 indicate that the technique is nearly

as effective as the Wiener filter. Images with high SNR (little noise) were dramatically restored with very small kernels. Small kernels were relatively less successful in restoring images with low SNR (significant noise), but the Wiener filter does not do well with low SNR either. Small kernels seem more robust than large kernels and the Wiener filter—that is, errors in the estimates of the system characterizations affected the small kernel restoration less. This observation is buttressed by the fact that the defining system of equations of small kernels is better conditioned than those for larger kernels and the Wiener filter, but any conclusions should be supported by further research.

These encouraging results should spur research into related problems:

- The results of Chapter 8 suggest that small kernels are more robust. Robustness is a significant advantage in real applications. This hypothesis should be thoroughly tested.
- Knowledge of the relationship between restoration kernel size and error would be very helpful. Undoubtedly, this relation is a complicated function of all of the components of the imaging system, but a parametric expression for predicting error would be useful in setting the kernel size.
- Cascaded convolution is a technique for efficient spatial processing. A cascade is a series of convolutions. Certain types of large kernels can be decomposed into a series of small kernels and applied much more efficiently. Cascading small spatial kernels is an area of active research[2,3,4,5,6,7,8,9,10,11].
- One might wish to impose other constraints on the restoration. For example, the kernel values might be limited to integers so the implementation can be restricted to integer arithmetic. Constraints also can be placed on the restoration result (e.g., as in the constrained-least-squares filter). These restrictions and others have received widespread consideration, but should be integrated with constraints on restoration kernel size.
- Lahart[12] noted that small kernels can be varied adaptively to deal with spatially variant PSF's, noise, and autocorrelation. In an example, he classed pixels as lying either in regions with high or low autocorrelation and applied one of two restoration kernels according to the pixels class. He noted that this binary division could be expanded to a range of classes. Because the assumption of a stationary process is seldom realistic, space-variant methods merit investigation. Because

adaptive processing requires more computation, efficient algorithms are particularly important.

- Fidelity measures are criticized for not matching our visual criteria. This being the case, how do restorations with small kernels compare subjectively to Wiener restorations? Hazra, Miller, and Park[13] have initiated studies to relate fidelity of acquisition and display to subjective standards. This work could be applied to image restoration. RMS optimal small kernel restorations should be compared to results for other methods, especially other techniques that may be used to limit processing (e.g., windowed kernels and iterative techniques).
- Small kernels can be used in problems other than image restoration. For example, computer vision requires high-speed image processing for edge detection, pattern matching, and texture analysis. Small kernel convolution could be applied to these and other problems. The approach can be applied directly to yield a mean-square optimal estimate of a characteristic function of the scene (e.g., of a band-pass filtered scene for edge detection).

The optimal small restoration kernel is based on a rigorous derivation, but restoration research should not stray far from practical solutions. The optimal small kernel restoration is a practical technique—it is simple, yet efficient and effective.

References

- [1] B. R. Hunt. Image restoration. In Michael P. Ekstrom, editor, *Digital Image Processing Techniques*, chapter 2, pages 53–76, Academic Press, Orlando, FL, 1984.
- [2] James H. McClellan. The design of two-dimensional digital filters by transformations. In *Proceedings of the Princeton Conference on Information Sciences and Systems*, pages 247–251, 1973.
- [3] Russel M. Mersereau, Wolfgang F. G. Mecklenbräuker, and Thomas F. Quatieri. McClellan transformations for two-dimensional digital filtering: I—Design. *IEEE Transactions on Circuits and Systems*, 23(7):405–414, 1976.
- [4] Wolfgang F. G. Mecklenbräuker and Russel M. Mersereau. McClellan transformations for two-dimensional digital filtering: II—Implementation. *IEEE Transactions on Circuits and Systems*, 23(7):414–422, 1976.
- [5] James H. McClellan and David S. Chan. A 2-D FIR filter structure derived from the Chebyshev recursion. *IEEE Transactions on Circuits and Systems*, 24(7):372–378, 1977.
- [6] Jean-François Abramatic and Oliver D. Faugeras. Design of two-dimensional FIR filters from ‘small’ generating kernels. In *Proceedings of the Conference on Pattern Recognition and Image Processing*, pages 116–118, IEEE, 1978.
- [7] Oliver D. Faugeras and Jean-François Abramatic. 2-D FIR filter design from independent ‘small’ generating kernels using a mean square and Tchebyshev error criterion. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pages 1–4, IEEE, 1979.
- [8] Jean-François Abramatic. Image filtering by sequential convolution of ‘small’ generating kernels. In *Proceedings of the International Symposium on Circuits and Systems*, pages 539–542, IEEE, 1979.
- [9] William K. Pratt. Intelligent image processing display terminal. In *Advances in Display Technology*, pages 189–194, SPIE, 1979.
- [10] Russel M. Mersereau. The design of arbitrary 2-D zero-phase FIR filters using transformations. *IEEE Transactions on Circuits and Systems*, 27(2):142–144, 1980.

- [11] Jean-François Abramatic and Oliver D. Faugeras. Sequential convolution techniques for image filtering. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 30(1):1-10, 1982.
- [12] Martin J. Lahart. Local image restoration by a least-squares method. *Journal of the Optical Society of America*, 69(10):1333-1339, 1979.
- [13] Rajeeb Hazra, Keith W. Miller, and Stephen K. Park. Model-based quantification of image quality. In *Proceedings of the International Workshop on Visual Information Processing for Television and Tele-robotics*, NASA, 1989.

Appendix A

Useful Functions

This appendix defines several basic functions. They are described as infinite and continuous in one-dimension and, if appropriate, extended to two-dimensions. The one-dimensional functions are illustrated in Figure A.1. Most of the digital versions can be obtained by sampling—taking the function values at uniformly spaced points. Most of the functions can be periodically extended. A more rigorous mathematical development of these and other functions is given by Bracewell[1].

The *unit-pulse* has unit-value over a unit-range centered at the origin and is zero outside that range:

$$\Pi(x) = \begin{cases} 1 & |x| < \frac{1}{2} \\ \frac{1}{2} & |x| = \frac{1}{2} \\ 0 & |x| > \frac{1}{2} \end{cases} \quad (\text{A.1})$$

The periodic extension of the pulse is called a *pulse train*. The discrete function consists of the sample points of the continuous function— $\Pi[n] = \Pi(n)$. The two-dimensional unit-pulse is a *separable* function—it can be written as the product of a function of x and a function of y :

$$\Pi(x, y) = \Pi(x) \Pi(y). \quad (\text{A.2})$$

The unit-pulse is used to calculate a local average, for nearest-neighbor interpolation, and for truncation windowing.

The *unit-step* is unit-valued at locations greater than zero and zero at locations less than zero:

$$H(x) = \begin{cases} 1 & x > 0 \\ \frac{1}{2} & x = 0 \\ 0 & x < 0 \end{cases} \quad (\text{A.3})$$

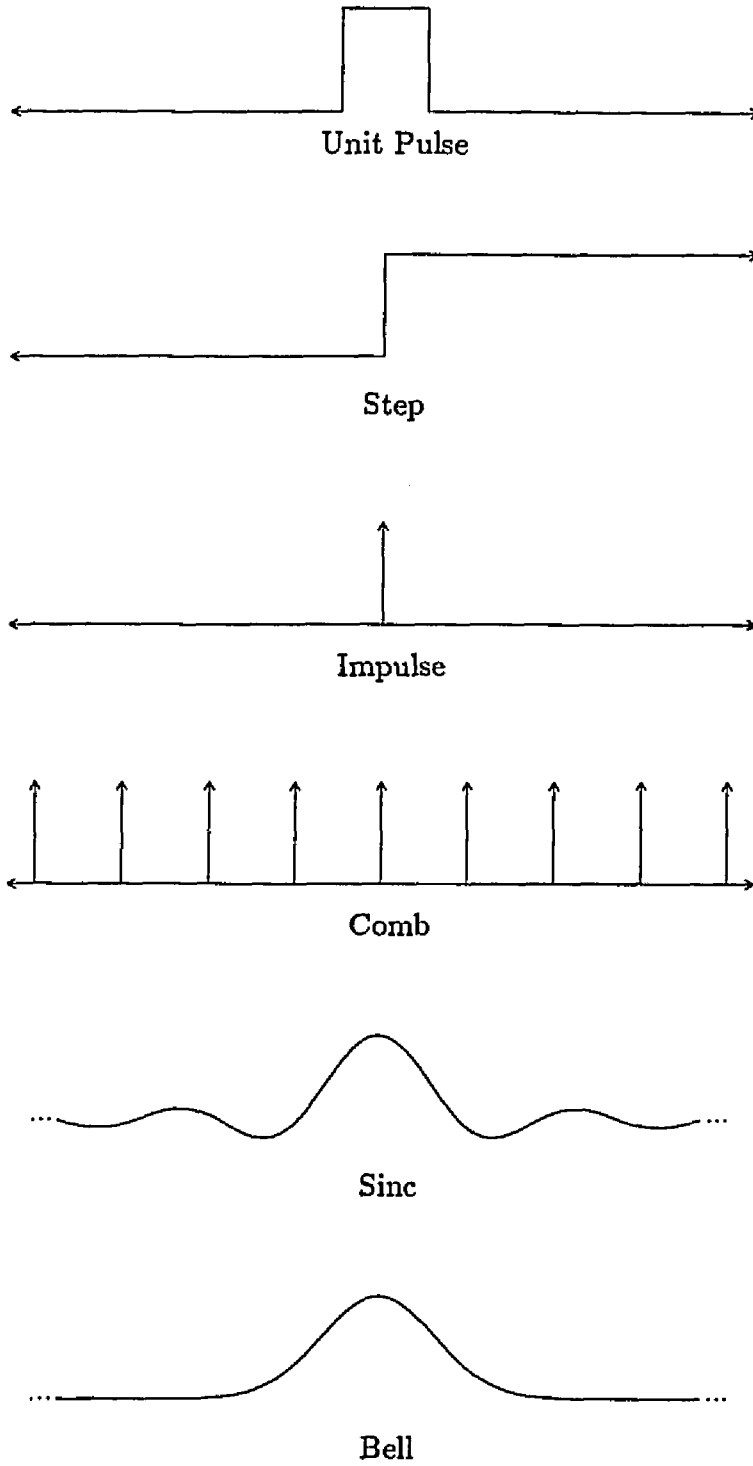


Figure A.1: Some Useful Functions

The step function cannot be periodically extended. The discrete step function consists of the sample points of the continuous function— $H[n] = H(n)$. The step function can be generalized to two-dimensions by specifying the angle θ of the step edge:

$$H_{\theta}(x, y) = \begin{cases} 1 & y > x \arctan \theta \\ \frac{1}{2} & y = x \arctan \theta \\ 0 & y < x \arctan \theta. \end{cases} \quad (\text{A.4})$$

The step function is used to represent simple discontinuities.

The *impulse* is an infinitely-strong pulse of infinitesimal extent. It is zero-valued, except at the origin, but has a unit integral:

$$\delta(n) = 0 \quad (n \neq 0) \quad (\text{A.5})$$

$$\int_{-\infty}^{\infty} \delta(n) dn = 1 \quad (\text{A.6})$$

The impulse is also called the *delta* or *Dirac delta* function. Though a delta function is not realizable, it can be defined using the limits of realizable functions, such as the unit-pulse:

$$\delta(x) = \lim_{w \rightarrow 0} w^{-1} \Pi\left(\frac{x}{w}\right). \quad (\text{A.7})$$

The impulse has an important relationship with the unit-step—the derivative of the step is the impulse:

$$\delta(x) = \frac{dH(x)}{dx} \quad (\text{A.8})$$

and the sequence of integrals of the impulse is the unit-step:

$$H(x) = \int_{-\infty}^x \delta(x') dx'. \quad (\text{A.9})$$

The digital impulse has the width of the unit-pulse and is scaled by the image size (or period):

$$\delta[n] = N \Pi(n). \quad (\text{A.10})$$

The two-dimensional impulse is the product of one-dimensional impulses:

$$\delta(x, y) = \delta(x) \delta(y). \quad (\text{A.11})$$

The impulse is used for sampling an image at a particular point.

The periodic extension of the impulse is the *comb* function. The comb function, sometimes called the *shah* function, is a train of impulses at unit intervals:

$$\text{III}(x) = \sum_{n=-\infty}^{\infty} \delta(x - n). \quad (\text{A.12})$$

The comb function is periodic, with a fundamental period 1. The two-dimensional comb function is a rectangular array of two-dimensional impulses (a bed-of-nails):

$$\text{III}(x, y) = \text{III}(x)\text{III}(y). \quad (\text{A.13})$$

The comb function is used for uniform sampling.

The *sinc* function, is a scaled sine wave:

$$\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x}. \quad (\text{A.14})$$

The sinc function and the unit pulse are a Fourier transform pair. The sinc function is infinite and cannot be periodically extended. The discrete function consists of the sample points of the continuous function— $\text{sinc}[n] = \text{sinc}(n)$. The two-dimensional sinc is the product of the one-dimensional functions:

$$\text{sinc}(x, y) = \frac{\sin(\pi x)}{\pi x} \frac{\sin(\pi y)}{\pi y}. \quad (\text{A.15})$$

The sinc function is the “ideal” interpolating function, meaning that if a continuous function is sampled sufficiently, the sinc can be used to exactly reconstruct the original function.

The *bell* function (or *Gaussian-spot* function) is

$$\exp(-x^2). \quad (\text{A.16})$$

This function is also called the *normal curve* because it is the distribution function of a normal random variable. The bell is infinite and cannot be periodically extended. The discrete function consists of the sample points of the continuous function. The two-dimensional function is separable and radially symmetric:

$$\exp(-x^2) \exp(-y^2) = \exp(-(x^2 + y^2)) \quad (\text{A.17})$$

The bell function is often used to model imaging devices (e.g., the display spot of a monitor).

Other images, such as sine and cosine waves, have important roles in image processing as well. These elementary images are basic tools of image processing.

References

- [1] Ronald N. Bracewell. *The Fourier Transform and Its Applications*. McGraw-Hill, New York, NY, revised second edition, 1986.

Appendix B

Characterizing Digital Image Acquisition Devices

B.1 Introduction

The mean-square-optimal small-kernel algorithm and most other image restoration algorithms require a characterization of the image acquisition device. Despite the popularity of digital imaging devices (e.g., charged-coupled device (CCD) array cameras) the problem of accurately characterizing such systems has been largely neglected in the literature. This appendix describes a simple method for accurately estimating the point spread function (PSF) and optical transfer function (OTF) of digital imaging devices. The method is based on the traditional knife-edge technique[1,2,3,4,5,6,7,8,9,10,11,12,13], but explicitly deals with fundamental sampled system considerations: aliasing, sample/scene phase shifts, and asymmetrical system functions.

Traditional knife-edge techniques rely on oversampling. For example, Tescher and Andrews[11] use 20-fold oversampling to estimate the spread and transfer functions from knife-edge scans. This procedure works well if the edge scans are oversampled. The knife-edge technique was originally developed for analog systems whose continuous output could be easily oversampled. For example, the photographic print from a film camera can be sampled to very high resolution with a digitizing microdensitometer.[2]

Design considerations dictate that typical digital imaging devices insufficiently sample.[14] If the transfer function response falls off at frequencies below the Nyquist frequency (half the sampling rate), the images will be blurred. If instead the transfer function response extends to frequencies beyond the Nyquist frequency, the images will be degraded by aliasing. A sharp cutoff of the transfer function at the Nyquist frequency causes ringing

(Gibbs phenomenon). For a well-designed digital imaging system, the transfer function is made to roll-off smoothly about the Nyquist frequency in an attempt to balance blurring, aliasing, and ringing. The resulting undersampled system cannot be accurately characterized by the traditional knife-edge technique.

In this appendix, the basic knife-edge technique is extended to address the problem of undersampling. Subpixel registration of sampled knife-edge scans increases the effective sampling rate, virtually eliminates aliasing, and permits estimation of the transfer function above the Nyquist frequency. Line averaging increases the SNR and effectively removes sample/scene phase effects.

The knife-edge technique yields an estimate of a one-dimensional slice through the center of the two-dimensional OTF. Even though the sampling grid of a digital acquisition device is fixed, the extended knife-edge technique can be used to estimate a slice at virtually any angle through the OTF. Multiple slices are used to assess system symmetry or separability or to construct a two-dimensional estimate of the OTF.

B.2 The Traditional Knife-Edge Technique

The knife-edge technique is popular for two reasons. First, a straight edge is much easier to accurately fabricate than an infinitesimal pulse, a sinusoid target, or a bar target (that must contain many edges at fixed intervals). Only one knife edge is needed and, with care, radiance nonuniformity and edge irregularity can be made small. Second, in some situations (e.g., remote sensing), the PSF and OTF must be estimated without the use of a specially fabricated target. Because sharp edges occur naturally in many scenes, knife-edge techniques frequently can be used even if an image of a special target is not available.

B.2.1 Theoretical Basis

If it were not for noise and the effects of sampling, estimating the PSF and OTF from knife-edge scans would be straightforward. The system's response to an edge is the *edge spread function* (ESF). The ESF for a scan line is the system's response to a step edge perpendicular to the scan. For example, the

ESF along the x -axis is a function in x of the PSF:¹

$$\begin{aligned} E_x(x, y) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} H(x') h(x - x', y - y') dx' dy' \\ &= \int_{-\infty}^x \int_{-\infty}^{\infty} h(x', y') dx' dy' \end{aligned} \quad (\text{B.1})$$

where $H(x)$ is the Heaviside unit-step function:

$$H(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x < 0. \end{cases} \quad (\text{B.2})$$

The derivative of the ESF with respect to the scan line is the *line spread function* (LSF). For the example along the x -axis,

$$\begin{aligned} L_x(x, y) &= \frac{\partial E_x(x, y)}{\partial x} \\ &= \int_{-\infty}^{\infty} h(x, y') dy'. \end{aligned} \quad (\text{B.3})$$

In the Fourier frequency domain, the transform of the ESF is a function of a slice of the OTF. Along the x -axis,

$$\hat{E}_x(u, v) = \begin{cases} \left(\frac{1}{2}\delta(u) - \frac{i}{2\pi u}\right) \hat{h}(u, v) & \text{if } v = 0 \\ 0 & \text{otherwise.} \end{cases} \quad (\text{B.4})$$

The spectrum of the LSF is a slice of the OTF. To continue with the example along the x -axis,

$$\begin{aligned} \hat{L}_x(u, v) &= (i2\pi u) \hat{E}_x(u, v) \\ &= \begin{cases} \hat{h}(u, v) & \text{if } v = 0 \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \quad (\text{B.5})$$

This is the basis of the knife-edge method—the Fourier transform of the derivative of an edge scan is a slice of the OTF.

Useful techniques must address several practical concerns: the edge scans of these equations have infinite extent, are noise-free, and have infinite resolution, but edge scans from real digital systems are finite, noisy, and sampled. The limit on the size of real edge scans is not a significant problem. Typically, the PSF is much narrower than the size of the scan, so the impact of this limitation is negligible.[9] Noise is invariably present in the edge scans and can be a source of significant error in the system function estimates. Methods for dealing with noise are described in the Section B.2.2. Sampling causes no problems if the cutoff of the system OTF is below the Nyquist limit, but digital imaging systems typically undersample. As shown in Section B.2.3, the traditional knife-edge technique gives a very poor estimate of the system functions in the face of undersampling.

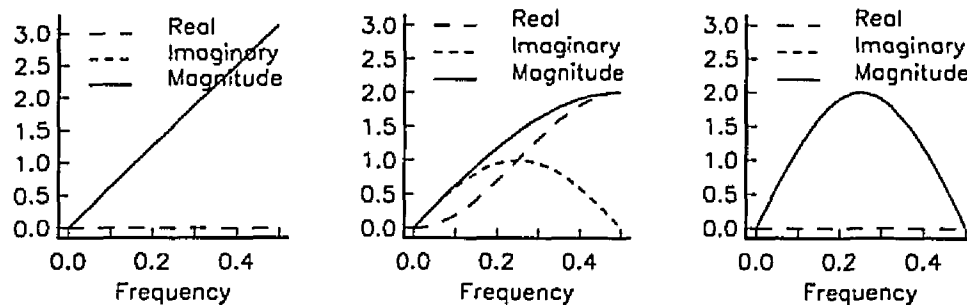
¹Cartesian coordinates, rather than radial coordinates, are used because the sampling grids are rectangular.

B.2.2 Dealing with Noise

Noise in the edge scans is magnified in the OTF estimate. Noise obscures the ESF in actual scans of the edge. Differentiation of the noisy ESF to obtain the LSF amplifies high-frequency components where the SNR is typically lowest. PSF and OTF estimation is an inverse or restoration problem—the result of the system must be analyzed to determine a component of the system. The inverse operation in the knife-edge procedure is differentiation. Differentiating the noisy ESF is identical to attempting to restore the LSF by application of the inverse filter—the derivative is the inverse of the step-edge target:

$$\frac{\partial(H(x))}{\partial x} = \delta(x) \quad (\text{B.6})$$

Differentiating, or inverse filtering, noisy data does not give satisfactory results[15]. Small changes in the edge scan (i.e., noise) can result in non-trivial changes in the estimate of the OTF. Taking the derivative of the scan with respect to x corresponds to multiplication in the Fourier frequency domain by $i2\pi u$. (This function is graphed in Figure B.1A.) The largest amplification is at the high frequencies where the SNR is typically lowest.



A. Exact Derivative B. [-1 1] Approximation C. [-1 0 1] Approximation

Figure B.1: Spatial Derivatives in the Fourier Frequency Domain

Many methods have been proposed to deal with noise, both in the literature on PSF and OTF estimation and the literature on restoration in general. Jones[6] suggests a perfect bandpass filter with a cutoff equal to the system cutoff frequency. This does nothing to noise below the cutoff frequency and requires knowledge of the system cutoff frequency. (Also, this assumes oversampling.) Blackman[7] suggests a combination of first smoothing the noisy OTF estimate by convolution with a frequency-variant filter

(the width of the Gaussian increases with frequency) and then averaging OTF estimates from several scans. He provides no theoretical basis for the specifics of his approach to smoothing the noisy OTF estimate. Tescher and Andrews[11] average several scans to reduce noise before differentiating. They also admit to resorting to repeated use of the Hanning filter to smooth the scans. The Hanning filter attenuates high-frequencies with the aim of suppressing noise. Unfortunately, high-frequencies of the OTF estimate are also suppressed. Popular restoration filters such as the Wiener (optimal least-squares) filter[16] and constrained least-squares filter[17,18] suffer from the same dilemma—signal as well as noise is suppressed. Approximating the derivative by convolving the noisy ESF with the kernel $[-1 \ 1]$ or $[-1 \ 0 \ 1]$ also suppresses high-frequency restoration. (Compare the high-frequency roll-off in the transforms of these kernels, pictured in Figures B.1B and B.1C, with the transform of the exact derivative, pictured in Figure B.1A.) Smith[12] suggests a parametric least-squares fit to a truncated series of Hermite polynomials, but in doing so forces the OTF estimate to a fixed form with as few as 4 to 6 variables.

Of these methods, only averaging deals with noise without affecting the estimate of the PSF and OTF. The other methods either suppress the estimated OTF where noise is high or presume a parametric form for the result. Line averaging is implemented by generating many scans across the edge and calculating the average along lines parallel to the edge. Assuming that the edge is perpendicular to the x -axis (a temporary assumption) and that the additive noise is white (zero-mean and spatially uncorrelated), the noise in the average of many scans is negligible:

$$\begin{aligned}
 \bar{E}_x(x, y) &= \frac{1}{N} \sum_{y=0}^{N-1} \left(\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} H(x') h(x - x', y - y') dx' dy' + e(x, y) \right) \\
 &= \int_{-\infty}^x \int_{-\infty}^{\infty} h(x', y') dx' dy' + \frac{1}{N} \sum_{y=0}^{N-1} e(x, y) \\
 &\xrightarrow{N \rightarrow \infty} \int_{-\infty}^x \int_{-\infty}^{\infty} h(x', y') dx' dy'.
 \end{aligned} \tag{B.7}$$

Under these assumptions, line averaging suppresses the noise without altering the signal.

B.2.3 Problems with Undersampling

The traditional knife-edge procedure fails to account for sampling effects and does not accurately characterize undersampled systems even in the absence of noise. Consider a noise-free system with an imaging device with a Gaussian

OTF:

$$\hat{h}(u, v) = e^{-\frac{u^2+v^2}{\rho_c^2}} \quad (\text{B.8})$$

The example system pictured in Figure B.2A has a value of 0.55 for ρ_c . This is midway between a system informationally optimized for high SNR ($\rho_c = 0.3$) and a more typical imaging system ($\rho_c = 0.8$).[19] The traditional knife-edge estimate of the system functions of this example are inaccurate. (See Figure B.2B). The OTF estimate is incorrect below the Nyquist frequency and is cut off at the Nyquist frequency. The sharp cutoff in the OTF estimate appears as ringing in the PSF estimate. The loss of high frequencies in the OTF estimate is seen in the loss of sharpness in the PSF estimate. Sample/scene phase shift causes variability in the estimate. If the knife edge in the image is shifted left or right relative to the sample points, the system estimates change.

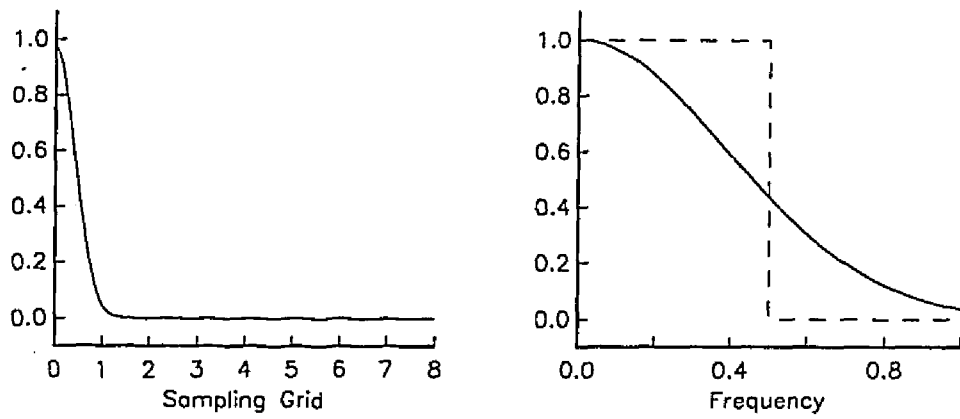
B.3 Sampled Systems

In the approach described here, the knife-edge scans are aligned to sub-pixel accuracy to achieve super-resolution (resolution greater than the sampling rate). This technique alleviates the problems of undersampling. Figure B.2C illustrates the improved system estimates obtained by this method for the example system of Section B.2.3. The OTF estimate is not cutoff at the Nyquist limit and there is very little blurring or ringing in the PSF estimate.

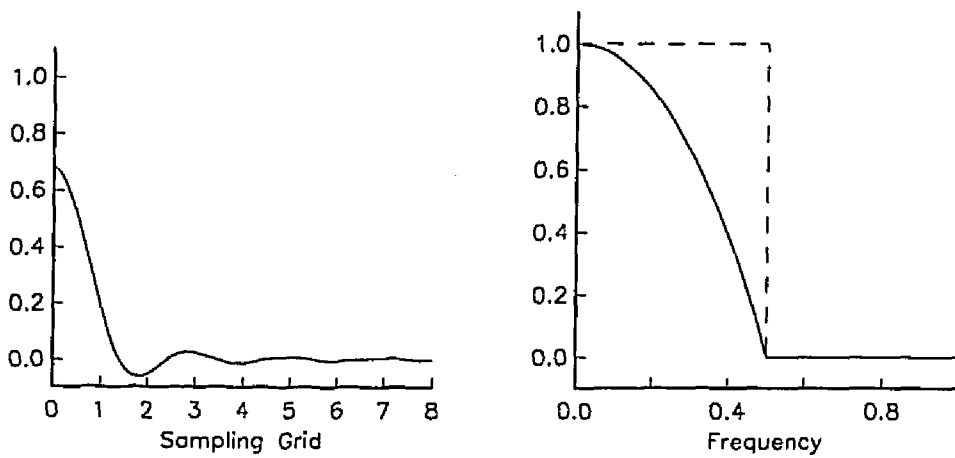
In order to achieve super-resolution, the knife edge is aligned slightly sloped relative to perpendicular (Figure B.3A). Tescher and Andrews[11] characterized this unavoidable shift as unfortunate, but it is the key to increasing the effective sampling rate and averaging sample/scene phase. Figure B.3B shows a sequence of scans, with the sample points marked. The edge shifts slightly relative to the samples from scan to scan. If the scans are first registered so that the edges align and then combined (as in Figure B.3C), the result contains many more sample points along the edge scan than any single scan line.

Precisely registering noisy scans (so the edges align) is difficult. Fortunately, the edge points in the scans do not need to be located exactly. For example, taking the system in Figure B.2A, doubling the sampling rate is sufficient to nearly eliminate aliasing. If the edge point can be located within an interval that is one-half of the sampling interval, it is possible to double the effective sampling rate.

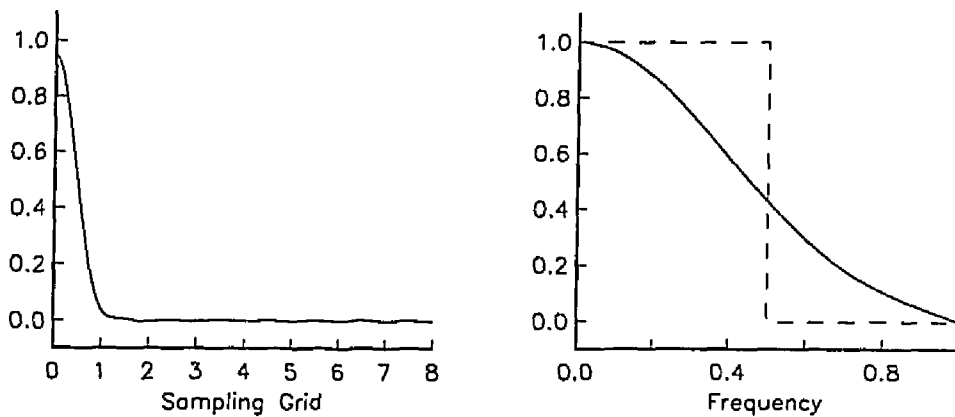
Several techniques have been suggested for locating edges to subpixel accuracy.[20,21,22,23] The approach used here is: 1) obtain an estimate for



A. Example System PSF and OTF($\rho_c = 0.55$)

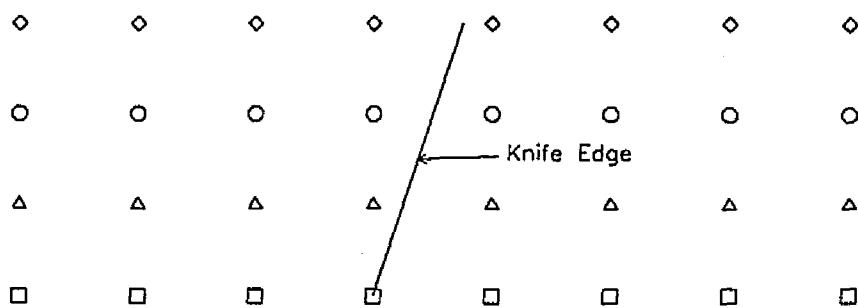


B. Traditional Knife-Edge Estimates

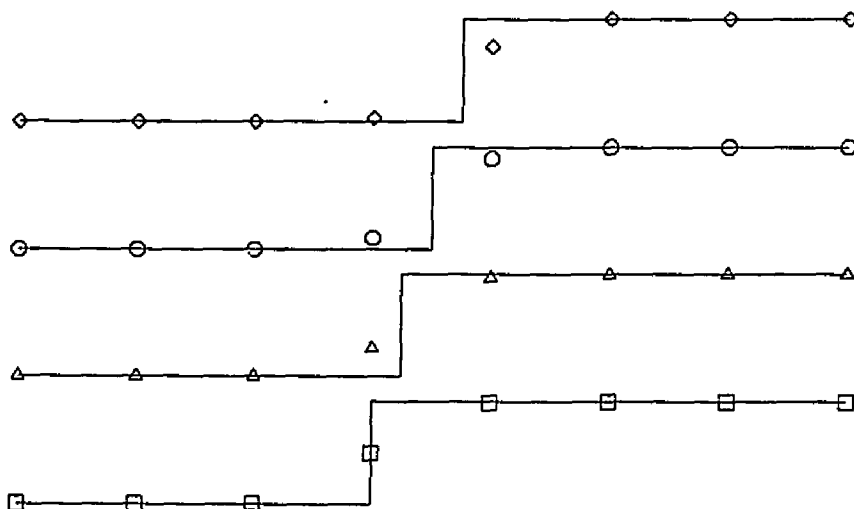


C. Improved Knife-Edge Estimates

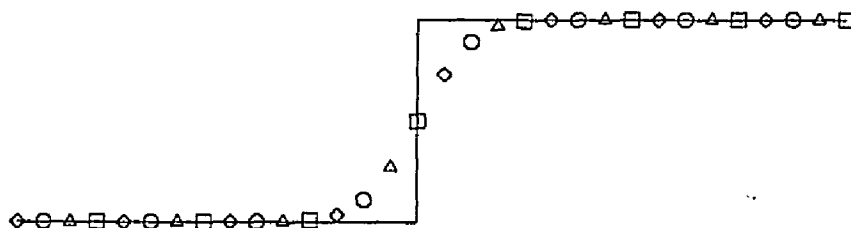
Figure B.2: Example System and Estimates



A. Sampling Grid With Knife Edge Skewed From Perpendicular



B. Knife-Edge Shift In Successive Scans



C. Combined Scan With Registered Edges

Figure B.3: Registering Scans With Shifted Knife Edge

the edge in each successive scan and 2) fit a line through the edge estimates. (Implicit in this approach is the assumption that the knife edge is precisely linear. If the edge is not linear, the step of fitting a line should be omitted.)

1. For each scan, determine an estimate of the edge location by the following steps:

- A. Obtain a first estimate of the sample interval containing the edge as the interval with the largest first-difference. This first estimate n_0 maximizes $|p[n_0 + 1] - p[n_0]|$ where p is the current edge scan. If the edge is sharp and clear, this interval is unique. The method breaks down if the PSF is so broad or the noise so great that this interval is not unique.
- B. Estimate the brightness values of the background and object by averaging the samples on the respective sides of the initial edge estimate:

$$\tilde{b} = \frac{1}{n_0 - k + 1} \sum_{n=0}^{n_0-k} p[n] \quad (\text{B.9})$$

$$\tilde{o} = \frac{1}{N - n_0 - k - 1} \sum_{n=n_0+k+1}^{N-1} p[n] \quad (\text{B.10})$$

where k is chosen larger than the effective radius of the PSF. Then estimate the brightness value at the edge point as the value midway between the background brightness and the object brightness:

$$\tilde{z} = \frac{\tilde{b} + \tilde{o}}{2}. \quad (\text{B.11})$$

- C. Update the value of n_0 as the interval in which the scan response crosses the brightness value \tilde{z} . That is where $p[n_0] \leq \tilde{z} < p[n_0 + 1]$.
- D. Calculate a *subpixel* estimate of the edge location as the point where the cubic spline fit to the scan samples crosses \tilde{z} . The edge estimate is obtained by solving for x in the cubic equation

$$\begin{aligned} \tilde{z} = & p[n_0 - 1](-(x + 1)^3 + 5(x + 1)^2 - 8(x + 1) + 4) \\ & + p[n_0](x^3 - 2x^2 + 1) \\ & + p[n_0 + 1]((1 - x)^3 - 2(1 - x)^2 + 1) \\ & + p[n_0 + 2](-(2 - x)^3 + 5(2 - x)^2 - 8(2 - x) + 4). \end{aligned} \quad (\text{B.12})$$

2. Obtain an estimate of the knife-edge location as the linear, least-squares fit to the set of edge estimates in the individual scans[24].

This technique for locating the edge has proved successful, but any other method that accurately locates step edges to subpixel resolution could be used in its place.

Next, the scans are registered according to the position of the edge in the scan. Then, the samples in the scans are located within subsample ranges. Figures B.4A and B.4B picture half-sample ranges in single scans relative to the edge. Each value in the average scan is calculated from sam-

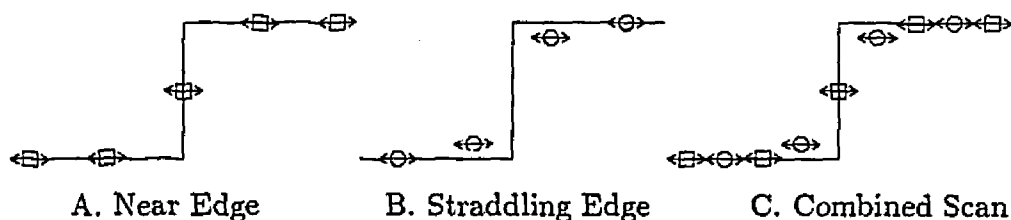


Figure B.4: Intervals for Registering Scan Lines

ple values that fall in the corresponding subsample interval. Figure B.4C illustrates how half-sample ranges double the number of sampling intervals in the original scans. The number of subsample ranges determines how much super-resolution is achieved.

It is convenient to define a new scale to match the increased resolution. In Figure B.4, the new sampling interval is half the previous sampling interval:

$$\Delta t \triangleq \frac{\Delta x}{2}. \quad (\text{B.13})$$

The values on the new scale are twice the values on the old:

$$t = 2x \quad (\text{B.14})$$

and the relative frequencies are halved:

$$\xi = \frac{u}{2}. \quad (\text{B.15})$$

Note that the super-resolution ratio need not be 2. It could just as well be any integer, but for typical systems 2 is sufficient. It is also convenient to revert to one-dimensional equations, postponing two-dimensional concerns until the next section.

The averaged edge scan, expressed in the new coordinates, is

$$\bar{p}[t] = \frac{1}{N} \sum_{n=0}^{N-1} \int_{-\infty}^{\infty} H(t' - a[n])h(t - t')dt' + e[t; n] \quad (\text{B.16})$$

where $a[n]$ is the edge shift in scan n and $e[t; n]$ is the additive noise of sample t in scan n . The edge shift is limited to one sample interval ($|a(n)| < \Delta t/2$). (Refer to Figure B.4C.) If the distribution of the edge shift is uniform over this range, then averaging a large number of scans approximates convolution with a pulse. Again, averaged white noise goes to zero:

$$\begin{aligned} \bar{p}[t] &\xrightarrow{N \rightarrow \infty} \int_{-\frac{1}{2}}^{\frac{1}{2}} \int_{-\infty}^{\infty} H(t' - a) h(t - t') dt' da \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Pi(a) H(t' - a) h(t - t') dt' da. \end{aligned} \quad (\text{B.17})$$

Subsequent computations (ideal reconstruction, the exact derivative, and compensation for the averaging pulse) are more efficient in the Fourier frequency domain. The transform equivalent of Equation B.17 is

$$\hat{\bar{p}}(\xi) = \int_{-\infty}^{\infty} \text{sinc}(\xi') \left(\frac{1}{2} \delta(\xi') - \frac{i}{2\pi\xi'} \right) \hat{h}(\xi) \text{III}(\xi - \xi') d\xi'. \quad (\text{B.18})$$

If the new sampling rate is sufficient to eliminate aliasing, then ideal reconstruction exactly recovers the function:

$$\hat{\bar{p}}(\xi) \Pi(\xi) = \text{sinc}(\xi) \left(\frac{1}{2} \delta(\xi) - \frac{i}{2\pi\xi} \right) \hat{h}(\xi). \quad (\text{B.19})$$

Just as for the traditional knife-edge approach, the spatial derivative is the inverse of the step edge:

$$\hat{\bar{p}}(\xi) \Pi(\xi) (i2\pi\xi) = \text{sinc}(\xi) \hat{h}(\xi). \quad (\text{B.20})$$

The estimation procedure must account for the averaging of sample/scene phase as well:

$$\frac{\hat{\bar{p}}(\xi) \Pi(\xi) (i2\pi\xi)}{\text{sinc}(\xi)} = \hat{h}(\xi). \quad (\text{B.21})$$

This estimate of the OTF extends beyond the Nyquist frequency of the original sampling function.

In summary, to generate the average scan:

1. Generate many scan lines by imaging a knife-edge target that is oriented nearly perpendicular to the scan lines. (A slant of at least one sample interval over the number of lines is required to produce a uniform distribution of sample/scene phase shifts.)
2. Register the edge points in the scans with subpixel accuracy.

3. Average the values from the scans that fall within subsample intervals. The size of the interval is chosen to provide the necessary super-resolution. For example, two-fold super-resolution requires subpixel intervals one-half the sampling interval. The result is a scan with increased effective sampling rate, greater SNR, and no sample/scene phase variability.

As noted, a bar target is used, so these steps are performed for both edges of the bar. The result is an averaged bar scan rather than an averaged edge scan.

Then, to estimate the OTF slice:

1. Discrete Fourier transform the average scan. (Because the following steps are performed only on the base period, truncation at the Nyquist limit is unnecessary.)
2. Calculate the exact derivative of the scan, in the frequency domain, by multiplying the transform at each frequency ξ by $i2\pi\xi$.
3. Sharpen the derivative of the scan, in the frequency domain, by dividing each frequency by $\text{sinc}(\xi)$. (This accounts for the averaging of sample/scene phase.)

In practice, the computations of the derivative and sharpening can be combined. The result is an estimate of a slice of the OTF.

B.4 Two-Dimensional Estimates

The knife-edge method yields an estimate of a one-dimensional slice through the center of the two-dimensional OTF. The knife-edge approach can be used to derive a two-dimensional estimate by generating a series of rotated slice estimates and interpolating between them. Before proceeding with this involved process, it is sensible to analyze the system for symmetry and separability.

If the system functions are radially symmetric, then a single slice of the OTF is sufficient to characterize the system; the slice of the OTF is a function in one variable—the radial distance from the origin. The radially symmetric PSF is given by the Hankel transform of the OTF slice. The symmetry of the OTF can be gauged by comparing the slice estimates along- x and along- y . If the two estimates are the same, it suggests (but does not demonstrate) that the system functions are radially symmetric.

Research has suggested that the system functions of digital imaging systems are not perfectly symmetric. For example, analog processing in the along-scan direction causes asymmetry in the system functions of digital scanners.[13] The array sensors of CCD camera systems also may cause asymmetry.[25] Large asymmetry in the system functions is undesirable, so the deviations in most systems are fairly small. If the along- x and along- y estimates are sufficiently similar, a radially symmetric estimate may be good enough. If the deviations are significant, then the system can be analyzed for separability.

The rectangular structure of CCD array cameras suggests that they be analyzed for separability. (See Figure B.5.) If the OTF is separable, then the

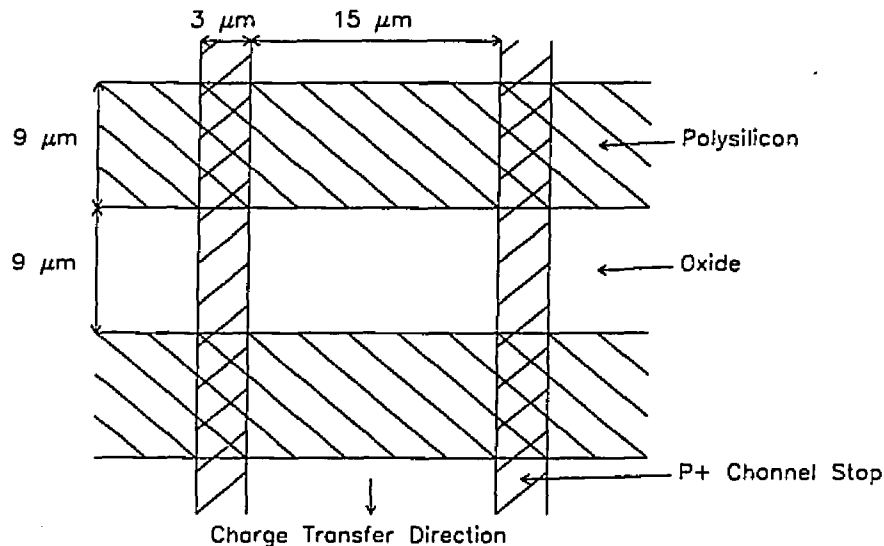


Figure B.5: CCD Detector Array

two slices along the axes are sufficient to characterize the system. An OTF is separable if there exist functions in each of the variables whose product is the OTF:

$$\hat{h}(u, v) = \hat{h}_x(u)\hat{h}_y(v). \quad (\text{B.22})$$

In this case, the OTF can be determined from slices along each dimension:

$$\hat{h}(u, v) = \frac{\hat{h}(u, 0)\hat{h}(0, v)}{\hat{h}(0, 0)}. \quad (\text{B.23})$$

If the OTF is separable, then the PSF is separable.

For example, the model horizontal and vertical response functions (linear falloff between potential wells) pictured in Figure B.6 are adapted from Talmi

and Simpson[26] and Hopwood[27]. (It is not suggested that this idealized model accurately portrays the responses of a CCD array; rather, its simplicity is well-suited for an example.) The separable product of these functions is

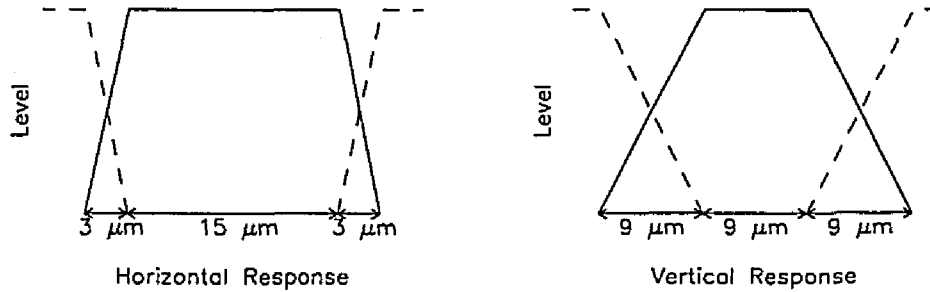


Figure B.6: Idealized Horizontal and Vertical Spatial Responses

$$h_d(x, y) = \Pi\left(\frac{6x}{5}\right) \Pi(6x) \Pi(2y) \Pi(2y) \quad (\text{B.24})$$

$$\hat{h}_d(u, v) = \text{sinc}\left(\frac{5u}{6}\right) \text{sinc}\left(\frac{u}{6}\right) \text{sinc}\left(\frac{v}{2}\right) \text{sinc}\left(\frac{v}{2}\right). \quad (\text{B.25})$$

The convolution of two separable functions is separable. Likewise, the product of two separable functions is separable. Therefore if, in addition to the detector array function, the functions of the other components of the device are separable (e.g., the OTF of the lens is Gaussian), then the system PSF and OTF are separable.

The separability of the system functions can be assessed empirically. Obtain three OTF-slice estimates from three knife-edge images—one along each of the two axes and one at 45° . The OTF slice at 45° is estimated from the edge scans such as those illustrated in Figure B.7. (It is important to note that the sampling interval of this scan is not the same as either the horizontal or vertical scans. All of the estimates should be converted to a single scale.) The product of the slices of the OTF along the axes should accurately predict the diagonal estimate by Equation B.23 if the system is separable. If Equation B.23 holds for the diagonal, it is strongly suggestive, but not definitive, evidence that the system is separable.

If the system functions are neither symmetric nor separable, a two-dimensional estimate can be derived by applying the extended knife-edge technique along angles between the horizontal, diagonal, and vertical estimates already described. Just as in estimating the diagonal slice (illustrated in Figure B.7), the knife-edge is rotated to approximately perpendicular the

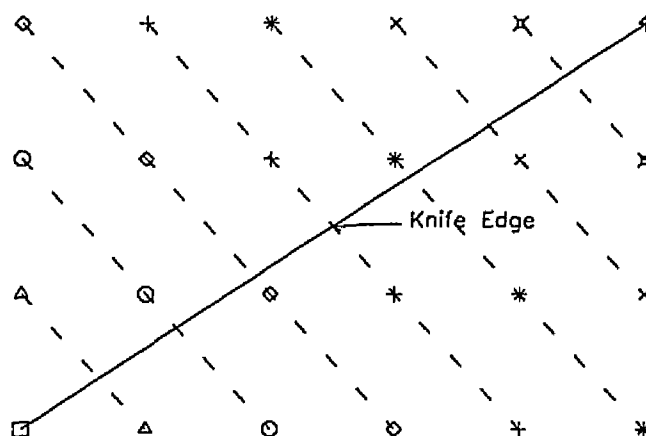


Figure B.7: Diagonal Knife-Edge Scans

desired estimate. Just as for the diagonal estimate, the slanted scans consist of the samples of many rows and columns.

B.5 Simulation Results

Simulation provides an exacting testbed, with a known system, scene, and noise process, for assessing the accuracy of this approach. The system model consists of two components: a model of the optics and a model of the CCD sensor array. The model for the optics was suggested by Johnson[28]:

$$\hat{h}_o(u, v) = \exp\left(-\left(\frac{\sqrt{u^2 + v^2}}{\rho_c}\right)^n\right). \quad (\text{B.26})$$

where ρ_c is the frequency at which the OTF is $1/e$ and n is the slope of the OTF line on a log-log plot. The idealized CCD model was introduced in the previous section. The equation for the idealized model is

$$\hat{h}_d(u, v) = \text{sinc}(W_x u) \text{sinc}(G_x u) \text{sinc}(W_y v) \text{sinc}(G_y v). \quad (\text{B.27})$$

where W_x and W_y are the dimensions of the potential wells and G_x and G_y are the distances between wells. (An example is pictured in Figures B.5 and B.6.) The OTF model is the product of the functions for the optics and the sensor array:

$$\hat{h}(u, v) = \hat{h}_o(u, v) \hat{h}_d(u, v). \quad (\text{B.28})$$

The results presented in this section were generated with the following parameters: $\rho_c = 1/\sqrt{2}$, $n = 2$ (a Gaussian, a separable function), $W_x = 5/6$, $G_x = 1/6$, $W_y = 1/2$, and $G_y = 1/2$ (as in Figures B.5 and B.6).

Three 1024×1024 digital scenes were used: one with a vertical bar, one with a horizontal bar, and one with a diagonal bar. In each scene, the bar was slightly sloped relative to the scan direction (a slope of 1 scene element per 64 scans). Each of these targets provided two knife edges (one on each side of the bar). The effect of the system on the scene is simulated by multiplying the transform of the scene by the OTF. The inverse Fourier transform of this product is the spatial image. The images of the horizontal and vertical bars were resampled in the scan direction to produce 1024, 64-element edge scans. The image of the diagonal bar was sampled both horizontally and vertically to produce 127 edge scans of varying length. (See Figure B.7.) In all three images, border effects corrupted the top and bottom scans, so only half the edge scans were retained (those in the center). White noise was then added to these edge scans, producing the type of data one expects from a real system.

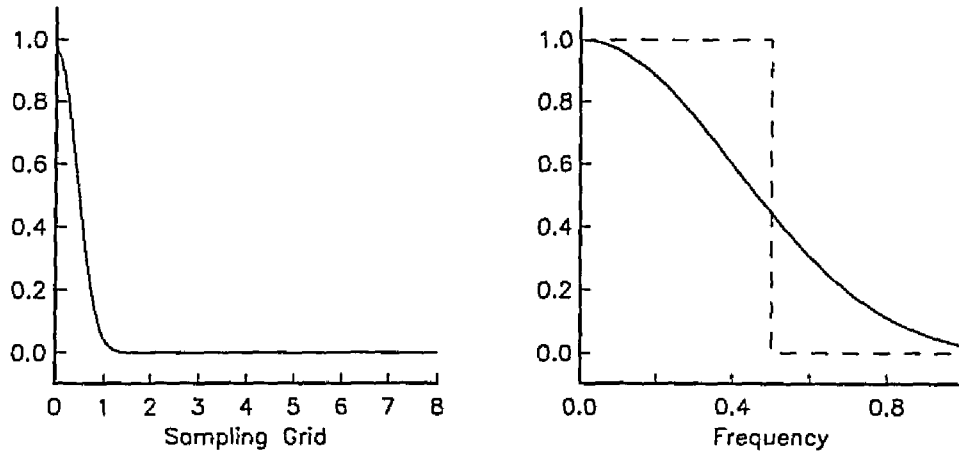
The model system function along- x is graphed in Figure B.8. Two sampled edge scans (the 0th and 256th) without noise are pictured in Figure B.9A. The same scans with noise (Image SNR = 64) are pictured in Figure B.9B. From only two scans, the estimate of the system functions is unreliable. Averaging all 512 scans, with subpixel registration, yields the scan with reduced noise and doubled effective sampling frequency pictured in Figure B.9C. This scan gives an accurate measure of the system, pictured in Figure B.8B. (A zero-phase system is assumed, so only the real part of the OTF estimate is retained. This is equivalent to averaging the left and right halves of the PSF estimate.)

Figure B.10 summarizes the accuracy of the system measure along- x . The ordinate of the graph is the mean-square-error (MSE) of the estimate relative to the energy of the OTF:

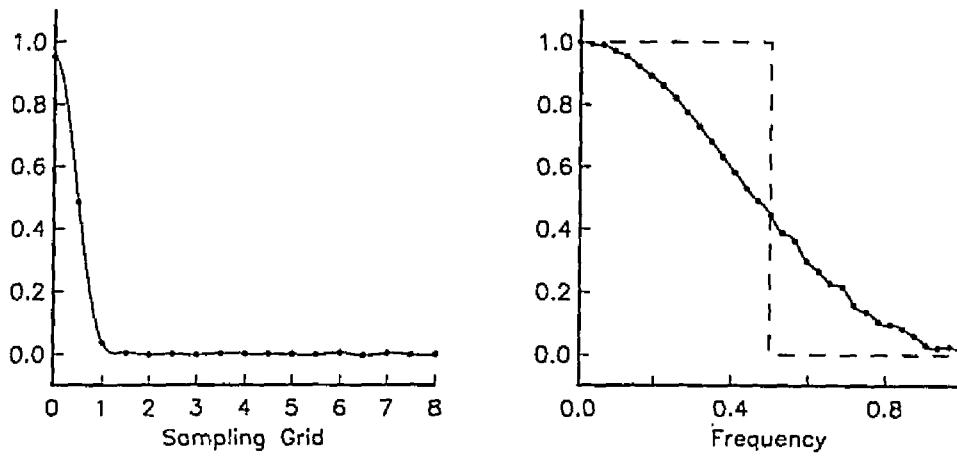
$$\text{Relative MSE} \triangleq \frac{\sum_u |\hat{h}(u) - \hat{h}_{est}(u)|^2}{\sum_u |\hat{h}(u)|^2}. \quad (\text{B.29})$$

The abscissa is the number of lines used in the line-averaging process. The different symbols indicate different levels of noise in the scans. The image signal-to-noise ratio (SNR) is the ratio of the step-edge height to the contrast (standard deviation) of the noise:

$$\text{Image SNR} \triangleq \frac{|\bar{o} - \bar{b}|}{\sigma_{Noise}}. \quad (\text{B.30})$$

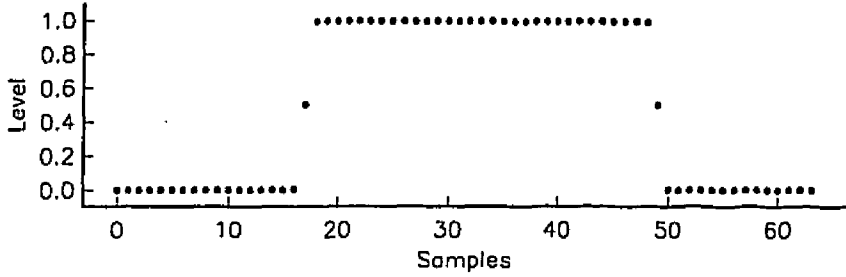
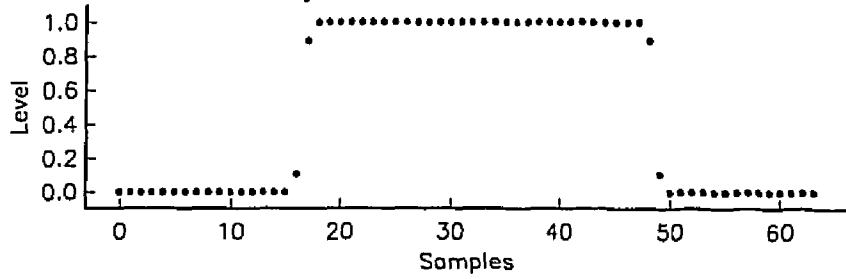


A. Simulated System Functions Along- x and Knife-Edge Estimate

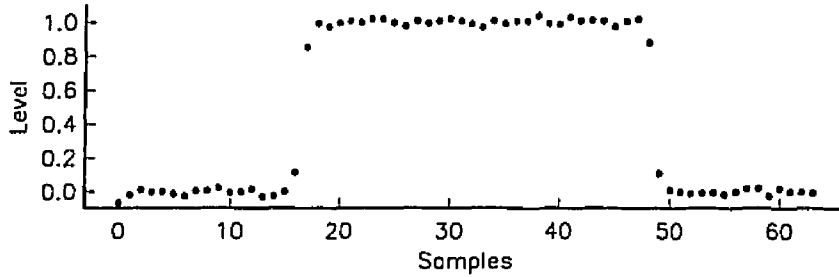


B. Estimate of System Functions

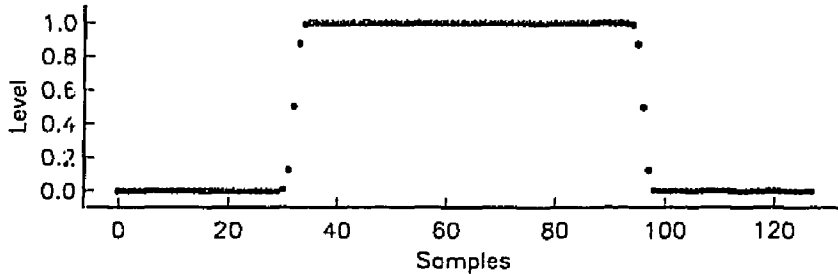
Figure B.8: Simulated System Function Along- x



A. Noise-Free Scans



B. Noisy Scans (Image SNR = 64)



C. Registered and Averaged Scan

Figure B.9: Scans

where \bar{o} is the function value of the object (i.e., the bar) and \bar{b} is the function value of the background. The plot values are averages for multiple choices of scan lines. For averaging only two lines, the pairs 0 and 256, 1 and 257, ..., 255 and 511 were used. For four lines the combinations were 0, 128, 256, and 384; ...; 127, 255, 383, and 511. The combinations for the other numbers of lines were generated in this fashion. The limiting effect of the ratio of scene-elements to scan-elements is evident in that the estimates for the noise-free image do not improve beyond using 16 lines. (The ratio of scene-elements to scan-elements is 16:1.) In fact, noise-free refers only to the additive noise as there is inevitably error introduced in digitally simulating a continuous process.

The MSE plots behave much as theory would suggest. Generally, halving image SNR (e.g., by quadrupling the energy or variance of the noise) quadruples the MSE of the estimate. Similarly, halving the number of lines averaged doubles the MSE of the estimate. Only at the bottom of the graph, where the limits of the accuracy of the simulation are reached, do these relationships break down. This graph can be used to gauge the accuracy of the estimate of a physical system or to indicate the number of lines required for desired accuracy. All that is required is an estimate of the noise to calculate the image SNR. Of course, the utility of these figures depends on how closely the real system is modeled by the simulation.

Figure B.5 illustrates the analysis of the system for symmetry and separability. Figure B.5A depicts the OTF along-x and the knife-edge estimate. Figure B.5B shows the OTF along-y and the knife-edge estimate. Figure B.5C shows the OTF along-diagonal, the knife-edge estimate, and the product of OTF estimates along-x and along-y. (Each the OTF estimates are based on 64 edge scans with SNR of 256.) The OTF estimates along-x and along-y differ slightly indicating that the system is not perfectly symmetric. However, as would be desirable in a real system, the asymmetry is small. The product of the along-x and along-y estimates is similar to the diagonal knife-edge estimate, supporting the hypothesis that the system is separable. None of the estimates is very different from the others, indicating that the system function is both nearly symmetric and separable. It is clear from these estimates that the Gaussian component of the simulated system (Equation B.28) is dominant.

B.6 Experimental Results

The extended knife-edge procedure was used to estimate the OTF of a Photometrics 183S slow-scan, three-stage, thermoelectric-cooled CCD camera

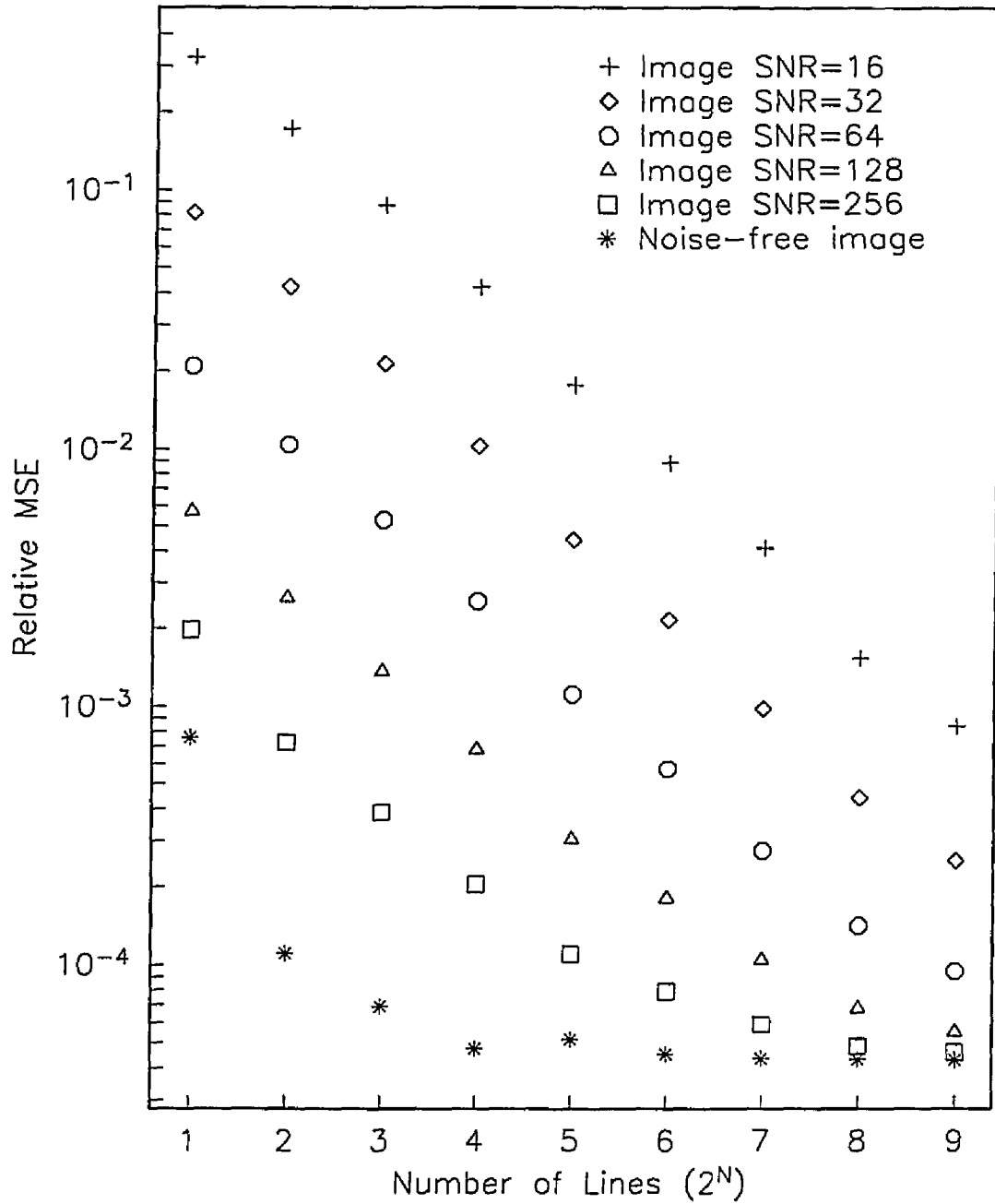
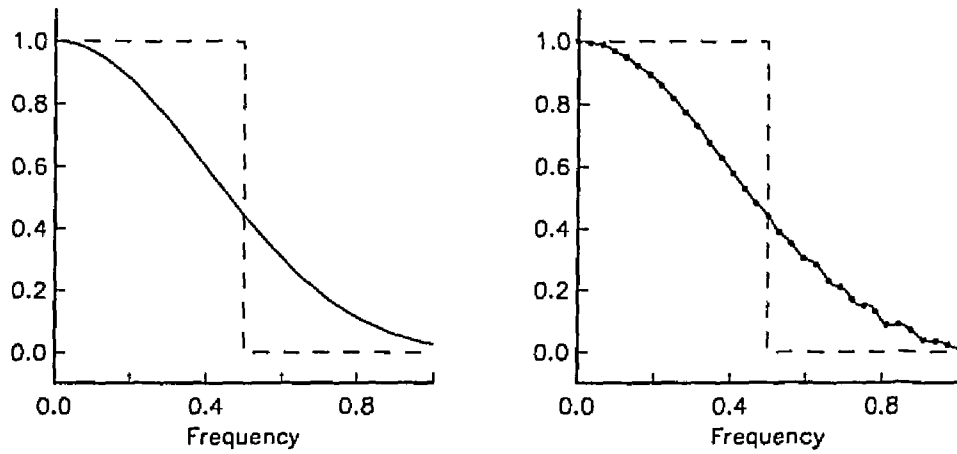
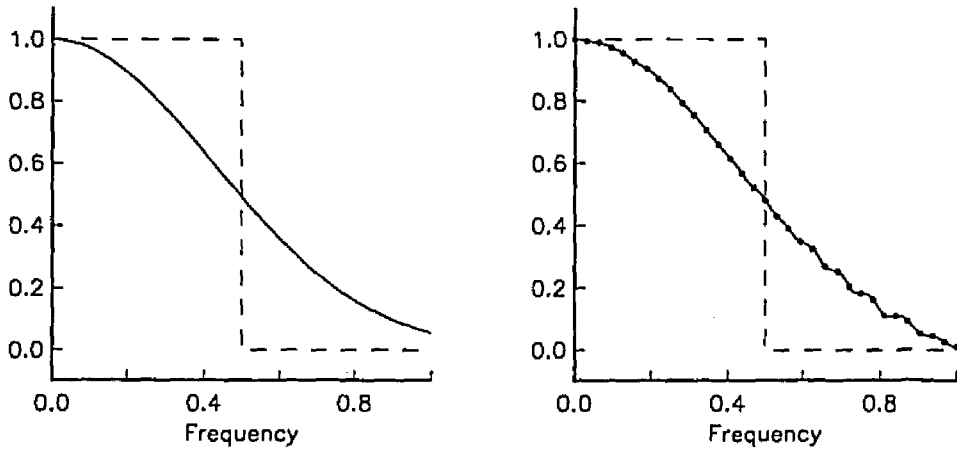


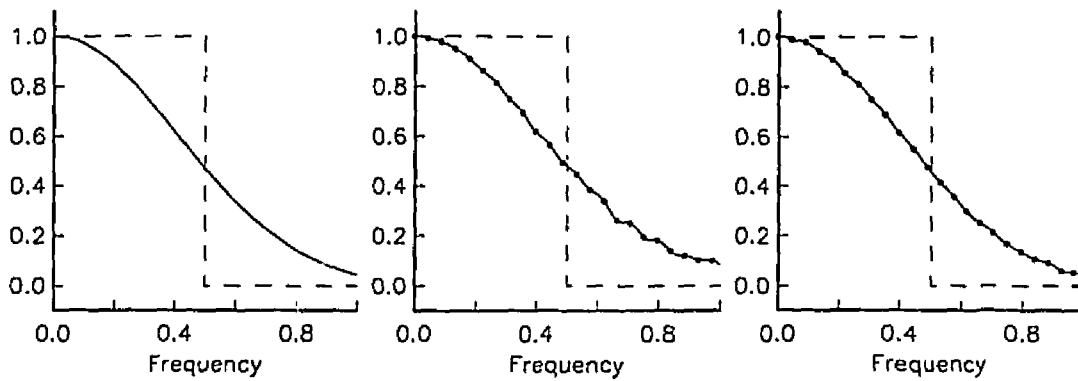
Figure B.10: Relative MSE of Estimates of System Function Along- x



A. OTF Along-x and Estimate



B. OTF Along-y and Estimate



C. OTF Along-Diagonal, Estimate, and Product of Along-x and Along-y Estimates

Figure B.11: Analysis of System Separability

system with a Texas Instruments 4849 array sensor and a standard 50mm lens. A commercial test target with a dark square printed on white mylar was placed as close to the camera as could be focused (45cm). The f -stop was set to 9.5 and the exposure time was 0.1 seconds. The resulting images were corrected for dark-field and flat-field response as

$$p'[m,n] = \frac{p[m,n] - d[m,n]}{f[m,n]/\bar{f}} \quad (\text{B.31})$$

where $d[m,n]$ is the dark-field response (the CCD output with the shutter closed), $f[m,n]$ is the flat-field response (the response to a target of uniform reflectance), and \bar{f} is the mean of the flat-field image. A 64×372 image containing a vertical bar was cut from the full 584×388 image. The image of the horizontal bar was cut to 372×64 and the image of the diagonal bar was cut to 128×128 .

Figure B.12 illustrates the estimate of the OTF to the sampling frequency. The response is typical for digital imaging systems balancing the tradeoff between blurring and aliasing—some attenuation of frequencies below the Nyquist frequency and some response above the Nyquist frequency that will result in aliasing. The magnitude of the OTF at the Nyquist limit is greater than 0.3. The estimate of the OTF begins to breakup near the sampling frequency where the OTF magnitude and the SNR are small. A traditional knife-edge estimate of the OTF of this system would be inaccurate below the Nyquist frequency and would not measure the components above it.

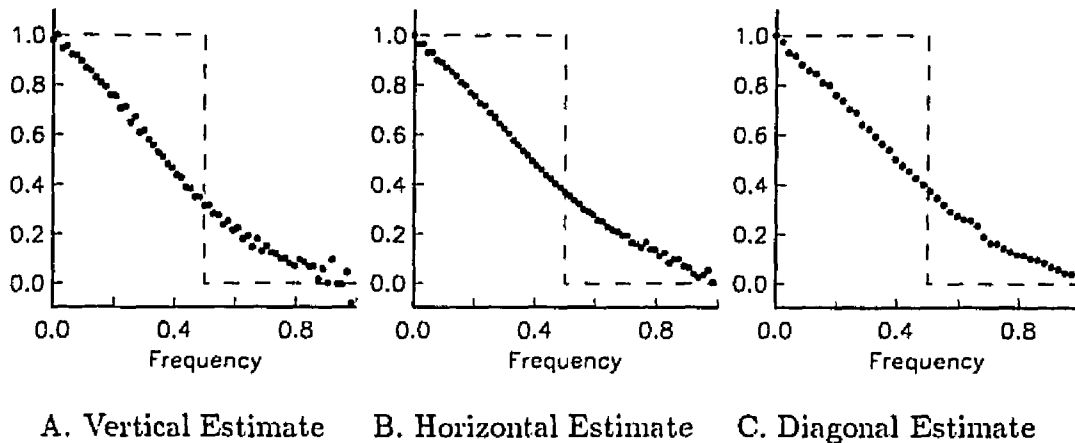


Figure B.12: Estimates of CCD System OTF

The vertical, horizontal, and diagonal responses are not very much different from one another. This is evidence that the system functions are nearly

circularly symmetric. The product of the vertical and horizontal transfer function estimates is a poor estimate of the diagonal response at the appropriate frequency. This indicates that the system function is not separable.

The OTF estimates were corroborated using an image of a bar chart that conforms to NBS 1010A and ANSI/ISO #2 standards taken under the conditions described above. The frequencies on the bar chart ranged from 1 cycle/mm to 5.6 cycles/mm. The sampling frequency in the target plane was 6 samples/mm, so the square-wave frequencies relative to the sampling frequency ranged from .167 to .933. From the OTF estimate of Figure B.12B, the modulation or contrast (i.e., the standard deviation) in the images of the square-waves is predicted by the equation

$$\hat{g}[\nu] = \frac{2a^2}{\pi^2} \sum_{k=0}^{\infty} \left| \frac{\hat{h}[\nu(2k+1)]}{2k+1} \right|^2 \quad (\text{B.32})$$

where $\hat{g}[\nu]$ is the estimated contrast for a square-wave of frequency ν , a is a scaling constant, and $\hat{h}[\nu]$ is the OTF. The normalized predicted contrast and observed contrast of the square-wave images are given in Figure B.6. The estimates accurately predict the observed square-wave contrast. The scatter of the observed contrast is due to noise and sample/scene phase shift. Only three bars were usable at all frequencies and with so few periods of the square wave, sample/scene phase is a source of significant variability.

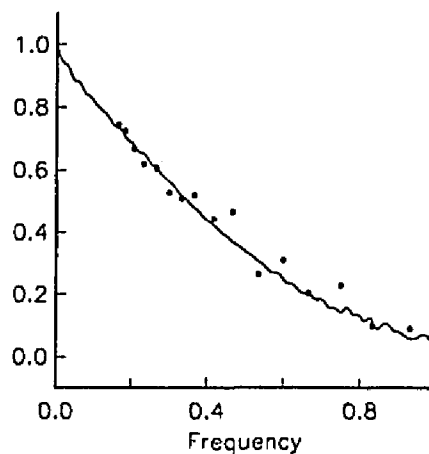


Figure B.13: Square-Wave Estimated and Observed Contrast

B.7 Conclusion

This technique provides accurate estimates of the transfer functions of digital imaging systems. The estimate is extended beyond the Nyquist limit by super-resolution achieved by subpixel registration of knife-edge scans. Line-averaging successfully eliminates sample/scene phase shift and increases SNR in the scans making the estimates less variable and more accurate. The technique can be applied at virtually any angle. Digital systems can be assessed for symmetry or separability. Two-dimensional estimates are generated by applying the technique to a series of rotated edge scans.

References

- [1] Roland V. Shack. Characteristics of an image-forming system. *Journal of Research of the National Bureau of Standards*, 56(5):245-260, 1956.
- [2] George C. Higgins. Methods for engineering photographic systems. *Applied Optics*, 3(1):1-10, 1964.
- [3] Richard Barakat and Agnes Houston. Line spread function and cumulative line spread function for systems with rotational symmetry. *Journal of the Optical Society of America*, 54(6):768-773, 1964.
- [4] Berge Tatian. Method for obtaining the transfer function from the edge response function. *Journal of the Optical Society of America*, 55(8):1014-1019, 1965.
- [5] Richard Barakat. Determination of the optical transfer function directly from the edge spread function. *Journal of the Optical Society of America*, 55(10):1217-1221, 1965.
- [6] Robert A. Jones. An automated technique for deriving MTF's from edge traces. *Photographic Science and Engineering*, 11(2):102-106, 1967.
- [7] Elliot S. Blackman. Recovery of system transfer functions from noisy photographic records. In *Seminar on Image Information Recovery*, pages 105-112, SPIE, 1969.
- [8] Robert A. Jones and Edward C. Yeadon. Determination of the spread function from noisy edge scans. *Photographic Science and Engineering*, 13(4):200-204, 1969.
- [9] M. E. Rabedeau. Effect of truncation of line-spread and edge-response functions on the computed optical transfer function. *Journal of the Optical Society of America*, 59(10):1309-1314, 1969.
- [10] Edward C. Yeadon, Robert A. Jones, and J. T. Kelly. Confidence limits for individual modulation transfer function measurements based upon the phase transfer function. *Photographic Science and Engineering*, 14(2):153-156, 1970.
- [11] A. G. Tescher and H. C. Andrews. Data compression and enhancement of sampled images. *Applied Optics*, 11(4):919-925, 1972.
- [12] P. L. Smith. New technique for estimating the MTF of an imaging system from its edge response. *Applied Optics*, 11(6):1424-1425, 1972.

- [13] R. M. Simonds. Two-dimensional modulation transfer functions of image scanning systems. *Applied Optics*, 20(4):619-622, 1981.
- [14] William F. Schreiber. *Fundamentals of Electronic Imaging Systems: Some Aspects of Image Processing*. Springer-Verlag, New York, NY, 1986.
- [15] H. C. Andrews and B. R. Hunt. *Digital Image Restoration*. Prentice-Hall, Englewood Cliffs, NJ, 1977.
- [16] C. W. Helstrom. Image restoration by the method of least squares. *Journal of the Optical Society of America*, 57(3):297-303, 1967.
- [17] David L. Phillips. A technique for the numerical solution of certain integral equations of the first kind. *Journal of the Association for Computing Machinery*, 9(1):84-97, 1962.
- [18] B. R. Hunt. The application of constrained least squares estimation to image restoration by digital computer. *IEEE Transactions on Computers*, 22(9):805-812, 1973.
- [19] Friedrich O. Huck, Carl L. Fales, Judith A. McCormick, and Stephen K. Park. Image-gathering system design for information and fidelity. *Journal of the Optical Society of America A*, 5(3):285-299, 1988.
- [20] Ali J. Tabatabai and O. Robert Mitchell. Edge location to subpixel values in digital imagery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(2):188-201, 1984.
- [21] Edward M. Mikhail, Mark L. Akey, and O. Robert Mitchell. Detection and sub-pixel location of photogrammetric targets in digital images. *Photogrammetria*, 39(3):63-83, 1984.
- [22] M. Mazumdar, B. K. Sinha, and C. C. Li. A comparison of several estimates of edge point in noisy digital data across a step edge. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 27-33, IEEE Computer Society, 1985.
- [23] Peter Seitz. Optical superresolution using solid-state cameras and digital signal processing. *Optical Engineering*, 27(7):535-540, 1988.
- [24] William H. Press, Brian P. Flannery, Saul A. Teukolsky, and William T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, New York, NY, 1988.

- [25] G. Béal, G. Boucharlat, J. Chabbal, J. P. Dupin, Fort B., and Y. Mellier. Thomson-CSF frame-transfer charge-coupled-device imagers: design and evaluation at low flux level. *Optical Engineering*, 26(9):902-910, 1987.
- [26] Yair Talmi and R. W. Simpson. Self-scanned photodiode array: A multi-channel spectrometric detector. *Applied Optics*, 19(9):1401-1414, 1980.
- [27] Ronald K. Hopwood. Design considerations for a solid-state image sensing system. In *Minicomputers and Microprocessors in Optical Systems*, pages 72-82, SPIE, 1980.
- [28] C. B. Johnson. A method for characterizing electro-optical device modulation transfer functions. *Photographic Science and Engineering*, 14(6):413-415, 1970.

Bibliography

Jean-François Abramatic. Image filtering by sequential convolution of 'small' generating kernels. In *Proceedings of the International Symposium on Circuits and Systems*, pages 539-542, IEEE, 1979.

Jean-François Abramatic and Oliver D. Faugeras. Design of two-dimensional FIR filters from 'small' generating kernels. In *Proceedings of the Conference on Pattern Recognition and Image Processing*, pages 116-118, IEEE, 1978.

Jean-François Abramatic and Oliver D. Faugeras. Sequential convolution techniques for image filtering. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 30(1):1-10, 1982.

H. C. Andrews and B. R. Hunt. *Digital Image Restoration*. Prentice-Hall, Englewood Cliffs, NJ, 1977.

Roger J. Arguello, Harvey R. Sellner, and John A. Stuller. Transfer function compensation of sampled imagery. *IEEE Transactions on Computers*, 21(7):812-818, 1972.

G. Backus and F. Gilbert. Uniqueness in the inversion of inaccurate gross earth data. *Philosophical Transactions of the Royal Society of London A*, 266:123-192, 1970.

Dana H. Ballard and Christopher M. Brown. *Computer Vision*. Prentice-Hall, Englewood Cliffs, NJ, 1982.

Richard Barakat. Determination of the optical transfer function directly from the edge spread function. *Journal of the Optical Society of America*, 55(10):1217-1221, 1965.

Richard Barakat and Agnes Houston. Line spread function and cumulative line spread function for systems with rotational symmetry. *Journal of the Optical Society of America*, 54(6):768-773, 1964.

- G. Béal, G. Boucharlat, J. Chabbal, J. P. Dupin, Fort B., and Y. Melier. Thomson-CSF frame-transfer charge-coupled-device imagers: design and evaluation at low flux level. *Optical Engineering*, 26(9):902-910, 1987.
- Elliot S. Blackman. Recovery of system transfer functions from noisy photographic records. In *Seminar on Image Information Recovery*, pages 105-112, SPIE, 1969.
- Ronald N. Bracewell. *The Fourier Transform and Its Applications*. McGraw-Hill, New York, NY, revised second edition, 1986.
- Kenneth R. Castleman. *Digital Image Processing*. Prentice-Hall, Englewood Cliffs, NJ, 1979.
- Nim-Yau Chu and Clare D. McGillem. Image restoration filters based on a 1-0 weighting over the domain of support of the PSF. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 27(5):457-464, 1979.
- Carl L. Fales, Friedrich O. Huck, Judith A. McCormick, and Stephen K. Park. Wiener restoration of sampled image data: end-to-end analysis. *Journal of the Optical Society of America A*, 5(3):300-314, 1988.
- Oliver D. Faugeras and Jean-François Abramatic. 2-D FIR filter design from independent 'small' generating kernels using a mean square and Tchebyshev error criterion. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pages 1-4, IEEE, 1979.
- B. Roy Frieden. Image restoration by discrete convolution of minimal length. *Journal of the Optical Society of America*, 64(5):682-686, 1974.
- Jack D. Gaskill. *Linear Systems, Fourier Transforms, and Optics*. John Wiley and Sons, New York, NY, 1978.
- Robert A. Gonsalves, Thomas A. Lianza, and Andrew Masia. Generation of random scenes with controlled statistics. In *Image Understanding Systems II*, pages 146-152, SPIE, 1979.
- Rafael C. Gonzalez and Paul Wintz. *Digital Image Processing*. Addison-Wesley, Reading, MA, second edition, 1987.
- William B. Green. *Digital Image Processing: A Systems Approach*. Van Nostrand Reinhold, New York, NY, 1983.
- R. W. Hamming. *Digital Filters*. Prentice-Hall, Englewood Cliffs, NJ, second edition, 1983.

- Robert M. Haralick. Computer vision theory: The lack thereof. *Computer Vision, Graphics, and Image Processing*, 36:372-386, 1986.
- James L. Harris, Sr. Image evaluation and restoration. *Journal of the Optical Society of America*, 56(5):569-574, 1966.
- Rajeeb Hazra, Keith W. Miller, and Stephen K. Park. Model-based quantification of image quality. In *Proceedings of the International Workshop on Visual Information Processing for Television and Telrobotics*, NASA, 1989.
- C. W. Helstrom. Image restoration by the method of least squares. *Journal of the Optical Society of America*, 57(3):297-303, 1967.
- George C. Higgins. Methods for engineering photographic systems. *Applied Optics*, 3(1):1-10, 1964.
- Ronald K. Hopwood. Design considerations for a solid-state image sensing system. In *Minicomputers and Microprocessors in Optical Systems*, pages 72-82, SPIE, 1980.
- Berthold Klaus Paul Horn. *Robot Vision*. MIT Press, Cambridge, MA, 1986.
- Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, New York, NY, 1985.
- Friedrich O. Huck, Carl L. Fales, Nesim Haylo, Richard W. Samms, and Kathryn Stacy. Image gathering and processing: Information and fidelity. *Journal of the Optical Society of America A*, 2(10):1644-1666, 1985.
- Friedrich O. Huck, Carl L. Fales, Judith A. McCormick, and Stephen K. Park. Image-gathering system design for information and fidelity. *Journal of the Optical Society of America A*, 5(3):285-299, 1988.
- Friedrich O. Huck and Judith A. McCormick. Image restoration for fidelity and visual quality. Submitted to *Computer Vision, Graphics, and Image Processing*.
- B. P. Hunt. The application of constrained least squares estimation to image restoration by digital computer. *IEEE Transactions on Computers*, 22(9):805-812, 1973.
- B. R. Hunt. Image restoration. In Michael P. Ekstrom, editor, *Digital Image Processing Techniques*, chapter 2, pages 53-76, Academic Press, Orlando, FL, 1984.

- C. B. Johnson. A method for characterizing electro-optical device modulation transfer functions. *Photographic Science and Engineering*, 14(6):413-415, 1970.
- Robert A. Jones. An automated technique for deriving MTF's from edge traces. *Photographic Science and Engineering*, 11(2):102-106, 1967.
- Robert A. Jones and Edward C. Yeadon. Determination of the spread function from noisy edge scans. *Photographic Science and Engineering*, 13(4):200-204, 1969.
- Donald E. Knuth. *The Art of Computer Programming: Seminumerical Algorithms*. Volume 2, Addison-Wesley, Reading, MA, second edition, 1981.
- Martin J. Lahart. Local image restoration by a least-squares method. *Journal of the Optical Society of America*, 69(10):1333-1339, 1979.
- M. Mazumdar, B. K. Sinha, and C. C. Li. A comparison of several estimates of edge point in noisy digital data across a step edge. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 27-33, IEEE Computer Society, 1985.
- James H. McClellan. The design of two-dimensional digital filters by transformations. In *Proceedings of the Princeton Conference on Information Sciences and Systems*, pages 247-251, 1973.
- James H. McClellan and David S. Chan. A 2-D FIR filter structure derived from the Chebyshev recursion. *IEEE Transactions on Circuits and Systems*, 24(7):372-378, 1977.
- Judith A. McCormick, Rachel Alter-Gartenberg, and Friedrich O. Huck. Image gathering and restoration: Information and visual quality. *Journal of the Optical Society of America A*, 6(7):987-1005, 1989.
- Wolfgang F. G. Mecklenbräuker and Russel M. Mersereau. McClellan transformations for two-dimensional digital filtering: II—Implementation. *IEEE Transactions on Circuits and Systems*, 23(7):414-422, 1976.
- Russel M. Mersereau. The design of arbitrary 2-D zero-phase FIR filters using transformations. *IEEE Transactions on Circuits and Systems*, 27(2):142-144, 1980.
- Russel M. Mersereau, Wolfgang F. G. Mecklenbräuker, and Thomas F. Quatieri. McClellan transformations for two-dimensional digital filtering: I—Design. *IEEE Transactions on Circuits and Systems*, 23(7):405-414, 1976.

- Edward M. Mikhail, Mark L. Akey, and O. Robert Mitchell. Detection and sub-pixel location of photogrammetric targets in digital images. *Photogrammetria*, 39(3):63-83, 1984.
- Don P. Mitchell and Arun N. Netravali. Reconstruction filters in computer graphics. *Computer Graphics*, 22(4):221-228, 1988.
- J. W. Modestino and R. W. Fries. Edge detection in noisy images using recursive digital filtering. *Computer Graphics and Image Processing*, 6:409-433, 1977.
- R. Nathan. *Digital Video Handling*. Technical Report 32-877, NASA, 1966.
- Athansios Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, New York, NY, 1984.
- Stephen K. Park. *Discrete Linear Systems*. 1988. Unpublished Lecture Notes (Computer Science Department, College of William and Mary).
- Stephen K. Park. *Introduction to Digital Image Processing*. 1987. Unpublished Draft (Computer Science Department, College of William and Mary).
- Stephen K. Park and Keith W. Miller. Random number generators: Good ones are hard to find. *Communications of the ACM*, 31(10), 1988.
- Stephen K. Park and Robert A. Schowengerdt. Image sampling, reconstruction, and the effect of sample-scene phasing. *Applied Optics*, 21(17):3142-3151, 1982.
- Stephen K. Park, Robert A. Schowengerdt, and Mary-Anne Kaczynski. Modulation-transfer-function analysis for sampled image systems. *Applied Optics*, 23(15):2572-2582, 1984.
- David L. Phillips. A technique for the numerical solution of certain integral equations of the first kind. *Journal of the Association for Computing Machinery*, 9(1):84-97, 1962.
- William K. Pratt. *Digital Image Processing*. John Wiley and Sons, New York, NY, 1978.
- William K. Pratt. Intelligent image processing display terminal. In *Advances in Display Technology*, pages 189-194, SPIE, 1979.
- William H. Press, Brian P. Flannery, Saul A. Teukolsky, and William T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, New York, NY, 1988.

- Keith Price. Anything you can do, I can do better (no you can't)... *Computer Vision, Graphics, and Image Processing*, 36:387-390, 1986.
- M. E. Rabedeau. Effect of truncation of line-spread and edge-response functions on the computed optical transfer function. *Journal of the Optical Society of America*, 59(10):1309-1314, 1969.
- Stephen E. Reichenbach, John C. Burton, and Keith W. Miller. A comparison of algorithms for computing the two-dimensional discrete Hartley transform. *Journal of the Optical Society of America A*, 6(6):818-823, 1989.
- Stephen E. Reichenbach and Stephen K. Park. Computer generated scenes and simulated imaging. In *Technical Digest of the Optical Society of America Annual Meeting*, page 170, 1988.
- Terry E. Riemer and Clare D. McGillem. Constrained optimization of image restoration filters. *Applied Optics*, 12(9):2027-2029, 1973.
- Terry E. Riemer and Clare D. McGillem. Optimum constrained image restoration filters. *IEEE Transactions on Aerospace and Electronic Systems*, 13(2):136-146, 1977.
- Richard A. Roberts and Clifford T. Mullis. *Digital Signal Processing*. Addison-Wesley, Reading, MA, 1987.
- Azriel Rosenfeld and Avinash C. Kak. *Digital Picture Processing*. Academic Press, Orlando, FL, second edition, 1982.
- Bahaa E. A. Saleh. Trade off between resolution and noise in restoration by superposition of images. *Applied Optics*, 13(8):1833-1838, 1974.
- Otto H. Schade. Electro-optical characteristics of television systems: characteristics of vision and visual systems. *RCA Review*, 9:5-37, 1948.
- Otto H. Schade. Electro-optical characteristics of television systems: electro-optical specifications for television systems. *RCA Review*, 9:245-286, 1948.
- Otto H. Schade, Sr. Image reproduction by a line raster process. In Lucien M. Biberman, editor, *Perception of Displayed Information*, chapter 6, pages 233-278, Plenum Press, New York, NY, 1973.
- Ray Schmidt. *The USC-Image Processing Institute Data Base*. Technical Report 780, University of Southern California Image Processing Institute, October 1977.
- William F. Schreiber. *Fundamentals of Electronic Imaging Systems: Some Aspects of Image Processing*. Springer-Verlag, New York, NY, 1986.

- R. W. Schutten and G. F. Vermeij. The approximation of image blur restoration filters by finite impulse responses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(2):176-180, 1980.
- Peter Seitz. Optical superresolution using solid-state cameras and digital signal processing. *Optical Engineering*, 27(7):535-540, 1988.
- Roland V. Shack. Characteristics of an image-forming system. *Journal of Research of the National Bureau of Standards*, 56(5):245-260, 1956.
- R. M. Simonds. Two-dimensional modulation transfer functions of image scanning systems. *Applied Optics*, 20(4):619-622, 1981.
- D. Slepian. Linear least-squares filtering of distorted images. *Journal of the Optical Society of America*, 57(7):918-922, 1967.
- Harvey A. Smith. Improvement of the resolution of a linear scanning device. *SIAM Journal of Applied Mathematics*, 14(1):23-41, 1966.
- P. L. Smith. New technique for estimating the MTF of an imaging system from its edge response. *Applied Optics*, 11(6):1424-1425, 1972.
- John A. Stuller. An algebraic approach to image restoration filter design. *Computer Graphics and Image Processing*, 1:107-122, 1972.
- Ali J. Tabatabai and O. Robert Mitchell. Edge location to subpixel values in digital imagery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(2):188-201, 1984.
- Yair Talmi and R. W. Simpson. Self-scanned photodiode array: A multi-channel spectrometric detector. *Applied Optics*, 19(9):1401-1414, 1980.
- Berge Tatian. Method for obtaining the transfer function from the edge response function. *Journal of the Optical Society of America*, 55(8):1014-1019, 1965.
- A. G. Tescher and H. C. Andrews. Data compression and enhancement of sampled images. *Applied Optics*, 11(4):919-925, 1972.
- S. Twomey. On the numerical solution of fredholm integral equations of the first kind by inversion of the linear system produced by quadrature. *Journal of the Association for Computing Machinery*, 10(1):97-101, 1963.
- N. Wiener. *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*. MIT Press, Cambridge, MA, 1949.

Edward C. Yeadon, Robert A. Jones, and J. T. Kelly. Confidence limits for individual modulation transfer function measurements based upon the phase transfer function. *Photographic Science and Engineering*, 14(2):153-156, 1970.

Stephen Edward Reichenbach

Stephen E. Reichenbach was born on August 15, 1954, in Lincoln, Nebraska. He attended Lincoln public schools and graduated from Lincoln Southeast High School in 1972. He was awarded a Regents' Scholarship from the University of Nebraska in Lincoln where he studied from 1972 to 1976 and earned a bachelor's degree in English in 1976. He taught English and mathematics at Simms High School, Simms, Montana (1976-77), George C. Marshall High School, Ankara, Turkey (1977-79), and Castle Rock Junior High School, Castle Rock, Colorado (1979-83). He continued his studies at the University of Northern Colorado (1980) and the University of Colorado (1981-82). From 1983 to 1985, he attended Washington University in Saint Louis where he was a graduate assistant (1983-85) and earned a master's degree in computer science in 1984. He was an instructor in the Computer Science Department at Southern Illinois University in Edwardsville during the 1985-86 academic year. From 1986 to 1989, he attended the College of William and Mary. While at William and Mary, he was a graduate assistant at the Virginia Institute of Marine Science (1986-87) and a recipient of a NASA Graduate Researcher Fellowship (1987-89). In 1989, he completed his dissertation, *Small-Kernel Image Restoration*, and earned a doctoral degree in computer science. He joined the faculty of the Computer Science and Engineering Department at the University of Nebraska in Lincoln in the fall of 1989 and was granted a one-year leave-of-absence to accept an award from the National Research Council for post-doctoral research at the NASA Langley Research Center in Hampton, Virginia.