

2001

Sub-pixel techniques to improve spatial resolution

Mark Robert Idema

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

Idema, Mark Robert, "Sub-pixel techniques to improve spatial resolution" (2001). *Dissertations, Theses, and Masters Projects*. Paper 1539623390.

<https://dx.doi.org/doi:10.21220/s2-xp6v-k986>

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.

SUB-PIXEL TECHNIQUES TO IMPROVE SPATIAL
RESOLUTION

A Dissertation

Presented to

The Faculty of the Department of Computer Science

The College of William & Mary in Virginia

In Partial Fulfillment

Of the Requirements for the Degree of

Doctor of Philosophy

by

Mark R. Idema

2001

APPROVAL SHEET

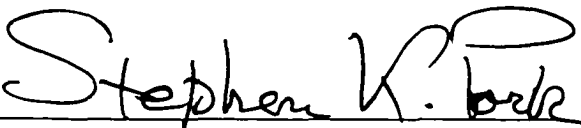
This dissertation is submitted in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy

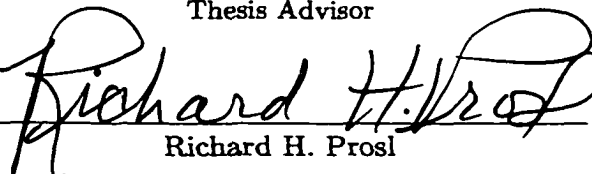


Mark R. Idema

Approved, December 2001




Stephen K. Park
Thesis Advisor (P.H. Prosl)



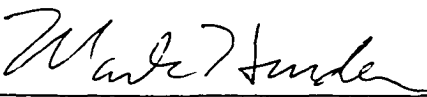
Richard H. Prosl



Weizhen Mao



Zia-ur Rahman
Co-Advisor



Mark K. Hinders

for my wife,

whose infinite patience and continuous support were invaluable in completing this work

Table of Contents

Acknowledgments	ix
List of Figures	xiv
Abstract	xv
1 Introduction	2
1.1 Background	2
1.2 Recent Research	5
1.3 Notation	9
1.4 Overview	11
2 C/D/C Model Description	12
2.1 Coordinate System	14
2.2 Use of Fourier domain	14
2.3 C/D/C Model	15
2.3.1 Scene Representation	15
2.3.2 Image Formation	17

2.3.3	Sampling	19
2.3.4	Noise	20
2.3.5	Image Restoration	23
2.3.6	Image Reconstruction	24
2.4	Output Image Components	26
2.5	End-to-end Metric	28
3	Constrained Least Squares Filter	30
3.1	Filter Derivation	31
3.2	Importance of a Properly Defined System Model	35
3.2.1	C/C Model Results	35
3.2.2	C/D/C Model Results	40
3.2.3	CLS Suppression of Noise Enhancement	41
3.3	A Priori Knowledge of CLS Filter Components	43
4	Pre-Sample Filter Estimation	44
4.1	Filter Estimation Technique	45
4.2	Estimating \hat{H} From p_c	49
4.2.1	Scene Model	49
4.2.2	\hat{H} Shape and Subsampling	51
4.2.3	\hat{H} Estimate ($\alpha = 0.2$)	52
4.2.4	\hat{H} Estimate ($\alpha = 0.4$)	53
4.2.5	\hat{H} Estimate ($\alpha = 0.6$)	54
4.3	Bins Per Pixel (Noise vs. Nyquist Frequency)	54

4.4	Data Density Along Frequency (ω) Axis	56
4.5	Algorithm	59
4.6	Alternative Approach to OTF Estimation	60
4.7	Fitting a Model Function to the Estimate	60
4.8	\hat{H} Is 2-dimensional	62
4.9	Importance of Edge Location Estimation	64
5	Edge Location Algorithms	66
5.1	System Model	67
5.1.1	Scene Model	67
5.1.2	PSF/ESF Model	68
5.1.3	Noise Model	70
5.1.4	System Simulation	71
5.1.5	Ensemble Simulation	72
5.2	Half-Pixel PSF Algorithm Bias	73
5.3	Edge Location Algorithms	75
5.3.1	Global PSF Algorithms	76
5.3.1.1	Approximation Using Discrete Differentiation (GPSF1) . .	76
5.3.1.2	Approximation Using Discrete Differentiation with Means (GPSF2)	79
5.3.2	Local PSF Algorithms	81
5.3.2.1	Quadratic Interpolation (LPSF1)	81
5.3.2.2	Weighted Mean (Center of Mass) (LPSF2)	82

5.3.2.3	Weighted Mean (Center of Mass) Revisited (LPSF3)	85
5.3.3	Global ESF Algorithms	86
5.3.3.1	Moment Matching (GESF1)	87
5.3.4	Local ESF Algorithms	89
5.3.4.1	Linear Interpolation (LESF1)	89
5.3.4.2	Cubic Convolution (LESF2)	92
5.3.4.3	Hermite Spline (LESF3)	94
5.4	Conclusion	96
6	Microscanning	99
6.1	System Model with Microscanning	99
6.2	CLS Filter with Microscanning	102
6.3	The Nyquist Frequency and Aliasing	103
6.3.1	Nyquist Frequency relative to the OTF	103
6.3.2	Effects of Aliasing	105
6.4	1-Dimensional Microscan Example	109
6.5	Complete vs. Partial Reconstitution of Input Scene	111
6.6	Real Digital Image Processing	118
7	Real Data Application and Conclusions	119
7.1	OTF Estimation	120
7.2	Real Data System Model	124
7.3	Conclusions	129

ACKNOWLEDGMENTS

I would like to express my appreciation to the various individuals who made this work possible. First, I would like to thank my committee for their time and effort in reviewing this document. In particular, I owe recognition to Dr. Zia-ur Rahman for his guidance and assistance toward completion of this work.

In addition, I am indebted to James Howe and the Night Vision Laboratory at Fort Belvoir, Virginia, as well as Dr. Mark Hinders and Gregg and Kia Rippel for their assistance in generating digital images used throughout this research endeavor.

Last, but certainly not least, I owe a great deal of gratitude to Dr. Stephen K. Park, whose constant, thought-provoking guidance and endless attention to detail compelled me to produce the best that I could. Although Dr. Park was not present for the completion of this work, I hope that this dissertation constitutes “something [he] and I can be proud of.”

List of Figures

2.1	System Model	12
2.2	Coordinate System	14
2.3	Representation Passband, Ω_s	16
2.4	Acquisition Passband, Ω_h (relative to Ω_s - shaded region)	19
2.5	Sampling Passband, Ω_p , solid square (relative to Ω_h - solid circle, and Ω_s - dotted square)	22
2.6	Components of p	26
2.7	Processed Components of p	27
3.1	Generic High-Boost Filter	31
3.2	Definition of p'	32
3.3	C/C System Model	35
3.4	scene (s)	36
3.5	OTF (\hat{H})	37
3.6	OTF blurred ($g = s \otimes h$)	38
3.7	Inverse filter, no noise	38
3.8	Inverse filter, with quantization noise only	39

3.9	C/D/C System Model	40
3.10	CLS filter, no noise	41
3.11	CLS filter, with noise	42
3.12	Noise Suppression through High-Boost Filter	42
4.1	Knife Edge and Edge Profile	45
4.2	Edge Location Estimation	46
4.3	Binning Edge Profiles to form Composite Edge Profile	47
4.4	Binned data in 16 bins across 4 pixels	48
4.5	Averaged bin values in 16 bins across 4 pixels	48
4.6	Super-Resolution Edge Profile	49
4.7	Ideal edge function, input scene	50
4.8	Various h graphs relative to α	52
4.9	\hat{H} Estimate, $\alpha = 0.2$	53
4.10	\hat{H} Estimate, $\alpha = 0.4$	53
4.11	\hat{H} Estimate, $\alpha = 0.6$	54
4.12	Binned data in 8 bins across 4 pixels	55
4.13	Averaged bin values in 8 bins across 4 pixels	55
4.14	Super-Resolution Edge Profile	56
4.15	Super-Resolution Edge Profile with Padded Data	57
4.16	\hat{H} Estimate with Padded Data	57
4.17	Acquired Square Image and Estimated \hat{H} in x_2 direction	61
4.18	Acquired Square Image and Estimated \hat{H} in x_1 direction	63

4.19	Acquired Square Image and Estimated \hat{H} in diagonal direction	64
5.1	Ideal Edge Profile	67
5.2	PSF and ESF graphs	69
5.3	Sampled Edge Profile Data	73
5.4	Discretely Differentiated Edge Profile Data	73
5.5	Half-pixel Bias Depiction	74
5.6	Discrete Differentiation Algorithm RMSE Results	78
5.7	Representation of K in Edge Profile Data	80
5.8	Discrete Differentiation with Means Algorithm RMSE Results	80
5.9	Quadratic Interpolation	82
5.10	Quadratic Interpolation Algorithm RMSE Results	83
5.11	Center of Mass Calculation	83
5.12	Center of Mass Calculation, smallest q value lower-bound	84
5.13	Center of Mass Algorithm RMSE Results	85
5.14	Center of Mass Calculation, zero lower-bound	85
5.15	Center of Mass Revisited Algorithm RMSE Results	86
5.16	Moment Matching Algorithm RMSE Results	89
5.17	Linear Interpolation Calculation	90
5.18	Linear Interpolation Algorithm RMSE Results	91
5.19	Cubic Convolution Calculation	92
5.20	Cubic Convolution Algorithm RMSE Results	94
5.21	Hermite Spline Calculation	95

5.22	Hermite Spline Algorithm RMSE Results	96
5.23	Algorithms Producing Lowest RMSE	97
6.1	Acquisition Model	99
6.2	Microscan Acquisition and Composition	100
6.3	Various OTFs and Nyquist Frequency	104
6.4	$s(x_1, x_2)$	105
6.5	Single Scan Nyquist Frequency and 2×2 Microscan Nyquist Frequency	106
6.6	$p[n_1, n_2]$, 128×128 Sample	107
6.7	$r(x_1, x_2)$, $M = 1$	108
6.8	$r(x_1, x_2)$, $M = 2$	108
6.9	Sample Input Signal	109
6.10	Single Scan Sample and Reconstruction	110
6.11	$2 \times$ Microscan Samples and Reconstruction	110
6.12	$2 \times$ Microscan Samples, Restoration and Reconstruction	111
6.13	$s(x_1, x_2)$	112
6.14	$g(x_1, x_2)$	112
6.15	Frequency response $ \hat{S} $ of Aerial image with $\omega_1 = \omega_2$	113
6.16	$p[n_1, n_2]$, $N = 128$	113
6.17	$r(x_1, x_2)$, No Restoration, No Microscanning	114
6.18	$r(x_1, x_2)$, CLS Restoration, No Microscanning	115
6.19	$p[n_1, n_2]$, 2×2 Microscan	115
6.20	$r(x_1, x_2)$, No Restoration, 2×2 Microscanning	116

6.21	$\tau(x_1, x_2)$, CLS Restoration, 2×2 Microscanning	117
7.1	Knife-Edge Image	121
7.2	Super-Resolution LSF	122
7.3	Super-Resolution PSF	122
7.4	OTF Estimates, x_1 (\circ) and x_2 (\bullet) directions	123
7.5	OTF Estimates, x_1 and x_2 directions with \hat{H} model function	124
7.6	System Model	124
7.7	Real System Model	124
7.8	$p_{0,0}$ - First Scan	125
7.9	No Restoration, Single Scan	126
7.10	CLS Restoration, Single Scan	126
7.11	No Restoration, Microscan	128
7.12	CLS Restoration, Microscan	128

ABSTRACT

Image acquisition using a scene sampling device generally results in a loss of fidelity in the acquired image, particularly if the scene contains high frequency features. Acquired images are also degraded by the blurring effects of acquisition filtering, image reconstruction, and additive noise effects. To compensate for these degradations, a digital restoration filter that attempts to partially eliminate the blurring while avoiding amplification of the noise effects is needed. In addition, to compensate for undersampling, a subpixel technique known as microscanning is required. This dissertation provides research into the spatial resolution enhancement of digital images based on subpixel techniques that will help to minimize the impact of these degradations. Subpixel techniques investigated include microscanning and estimation of the function that measures the amount of blurring incurred during acquisition. These techniques will be used in conjunction with a constrained least squares restoration filter to achieve the best possible representation of the original scene.

**SUB-PIXEL TECHNIQUES TO IMPROVE SPATIAL
RESOLUTION**

Chapter 1

Introduction

1.1 Background

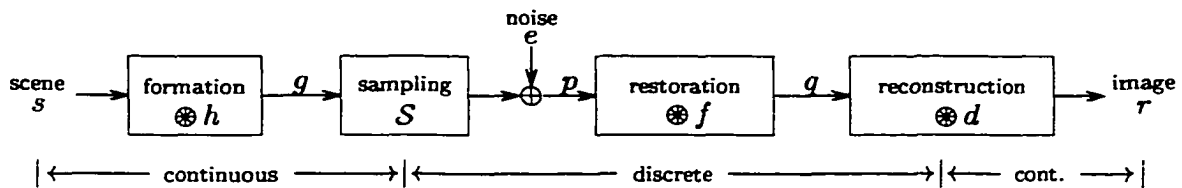
Image acquisition using a device that samples a scene (i.e., a digital imaging system) can potentially cause a lack of image fidelity because the device has a finite number of detectors. For example, scenes which contain high frequency features (or fine detail) may not be accurately acquired because the relative position of the detector sampling grid with respect to the scene could cause pixel-scale features to fall between adjacent detectors. The physical limitation imposed on the acquisition device by the sampling density of the detectors creates a problem known as *undersampling*.

Undersampling causes a noise-like effect known as aliasing. Aliasing occurs when frequencies in the scene exceed the Nyquist frequency of the acquisition device and are “folded” back onto lower frequencies.¹ Aliasing can be suppressed by *microscanning*. Microscanning

¹The Nyquist frequency is defined as the reciprocal of twice the distance between two adjacent sampling detectors.

is a process in which multiple digital images of the same scene are acquired, where each digital image differs from the other images in a prescribed manner. This process of acquisition will suppress aliasing at the expense of increased acquisition time. The difference among these multiple digital images is subpixel shifts of the image acquisition device prior to image acquisition. The subpixel shifts may be horizontal, vertical, or diagonal. When the multiple acquired images are digitally merged to form a composite image, the sampling density of the composite image increases relative to the sampling density of the individual images. Accordingly, it may be possible to resolve the pixel-scale features that are aliased in the individual images. The composite image can be restored using conventional digital image restoration methods, providing better spatial resolution than the individual images.

Using the Continuous/Discrete/Continuous system model (shown below and defined in Chapter 2)



it can be stated that the purpose of a restoration filter f is to account for and minimize the impact of (a) the blurring caused by the acquisition filter h and the reconstruction filter d , and (b) the noise effects caused by aliasing due to sampling S and the thermal and electronic noise in the device.

As will be discussed in Chapter 4, independent of the location of the scene relative to the sampling grid, blurring will occur during acquisition. The amount of blur is characterized by the optical transfer function (OTF) of the detectors and the optics. The acquisition OTF is the Fourier transform of the point spread function (PSF) associated with image

acquisition. When a point source of light is imaged, a certain amount of energy spreads to adjacent locations. This energy dispersion causes blurring. The amount of dispersion (blurring) is measured by the PSF[65].

To design a restoration filter that compensates for the blurring of the acquisition device an estimate of the OTF must be computed. A subpixel technique to estimate the OTF of an acquisition device has been proposed by Reichenbach et al[51]. This technique uses a composite “super-resolution” edge image to determine the line spread function of the acquisition device in either the vertical or the horizontal direction. Differentiation of the line spread function, followed by a Fourier transform, produces an estimate of the OTF for the chosen direction. For a complete two-dimensional representation of the OTF, an estimate of the OTF in multiple directions must be performed. The specifics of this technique will be presented in Chapter 4.

A blurring similar to that which occurs at image acquisition occurs during image reconstruction. Reconstruction involves the transformation of data from a digital image to a continuous output scene. Interpolation of the digital data on the display causes blurring. For the purposes of this dissertation, the associated reconstruction transfer function (RTF) will be modeled using a parametric cubic convolution function[45].

Image degradation can occur not only from the inherent characteristics of the acquisition and display devices but also from random additive noise effects during acquisition. “Inherent characteristics” of the acquisition device include all of the physical objects (lenses, detectors, etc.) which blur the image prior to sampling. Typically electronic noise effects are modeled as random and additive. However, in reality this may not be true. Noise effects that are represented in the c/d/c model (described in Section 2.4) include the quantization effects

which are introduced when analog data is converted to a discrete digital image. Non-uniformity of the response of the detectors is a source of noise not represented in the $c/d/c$ model. It is assumed that variations in the response of the detectors and (at the extreme) detector failure have been calibrated out of the digital image before image formation.

The restoration filter f is the only component of the system model that digitally processes the image. In Chapter 3, we introduce a specific restoration filter. The constrained least squares (CLS) restoration filter defined in Chapter 3 is constructed using knowledge of the OTF of the acquisition device and knowledge of the RTF of the reconstruction filter. The effectiveness of the restoration filter is investigated for a variety of signal to noise ratios.

1.2 Recent Research

Considerable research has been performed with respect to subpixel techniques and general image restoration[11, 12, 20, 21, 22, 23, 28, 29, 47, 57]. In a paper by Gillette, Stadtmiller, and Hardie[16] the authors propose the use of microscanning to reduce the aliased noise which is present when undersampling occurs. Both a “controlled” and an “uncontrolled” version of microscanning are presented. In the description of controlled microscanning, the authors refer to the use of a mirror or beam steerer to obtain digital images which are at a known subpixel displacement from one another. Composition of the resulting microscanned digital images into a single image can be accomplished either in the Fourier domain or in the spatial domain. Their version of uncontrolled microscanning, where the subpixel shifts between images are unknown, entails a complex reconstruction method that is not considered in this research. Although their theory is sound, they do not incorporate a

restoration filter into their model. The use of an inverse filter is mentioned as a means of accounting for “the effects of the presampling filter”, however, it is not employed in their model or their results. By using microscanning only, without restoration, they are effectively reconstructing the blurred, “pre-sample” image rather than the input scene. Further research was performed by Hardie[17], however, the system model presented still fails to incorporate restoration and reconstruction filters.

A failure to appreciate the need for a restoration filter is prevalent in the literature. For example, Feidenberg[13] also extols the benefits of microscanning. The use of “rectangular bidirectional” microscanning versus “diagonal” microscanning is discussed. A “reciprocal lattice” is defined and used to explain the effects of microscanning with respect to the Nyquist frequency. Derivations of a temporal modulation transfer function (or MTF, defined as the magnitude of the OTF) and an interpolation MTF (referred to in this research as RTF) are discussed. This discussion, however, is limited to the effects of microscanning with respect to the Nyquist frequency. The removal of the blurring associated with the PSF of the acquisition device is not considered.

Jacquemod et al[24] also use microscanning as a means of “oversampling” an input scene. The authors recognize the need for a restoration filter to eliminate the blurring with respect to the PSF of the acquisition device. However, their “deconvolution” filter is based solely on the impulse response of the detectors. They fail to account for the blurring induced by the optics of the acquisition device and the blurring which occurs during reconstruction. In addition, they assume a high signal-to-noise ratio and thus use an inverse filter as the restoration filter. In practice this type of restoration filter will boost the noise present in the image, perhaps dramatically.

Savakis and Trussell[55] investigate estimation of the PSF as a step in defining a restoration filter; however, their simulation model is flawed because they do not address the issue of aliasing. They claim that “given the sampling [density] there is nothing that can be done to overcome the limitations of undersampling.” They assume that the input scene is band-limited to the Nyquist frequency of the scene sampling device. Their assumption is inaccurate because most input scenes contain high-frequency features well beyond the Nyquist frequency. The authors fail to recognize that microscanning can facilitate restoration to well beyond the Nyquist frequency of the acquisition device. Thus, by performing the sampling operation as a convolution of the PSF filtered image (as their model states) rather than an actual sampling of the image data they artificially guarantee that aliasing will not occur.

Tzannes and Mooney[62] present a multi-step technique to estimate the OTF of an acquisition device. The first few steps of their technique are the same as the technique presented by Reichenbach et al[51], but in the final steps Tzannes and Mooney estimate the PSF as a function that is the summation of three Fermi functions. This estimated function requires the computation of 10 free parameters, however, they do not provide an explanation of how to compute these parameters. Thus, it is difficult to reproduce their results. In addition, the authors focus on an estimation of the OTF without explaining the purpose of the estimation. That is, there is no indication that the information obtained by estimating the OTF of an acquisition device can be used to restore the acquired image.

One approach to restoration is to use the Wiener filter, which is the optimal restoration filter based on the mean-square difference between the input scene and the output image. The Wiener filter, however, can only be implemented using theoretical models because it

relies upon information that might not be known, the frequency spectrum of the scene, for instance, to achieve optimal results. Uncertainty about the accuracy of the model has caused research into more practical restoration filters. A constrained least squares (CLS) filter is presented in this dissertation as a more appropriate restoration filter. With the exception of one real-valued user-specified smoothing parameter, all parameters for the CLS filter are known or can be estimated a priori.

1.3 Notation

For reference, all of the important notation and symbols used in this dissertation are summarized below.

s	→	spatial domain representation of the input scene.
\hat{S}	→	Fourier domain representation of the input scene.
σ_s	→	Standard deviation of s .
h	→	spatial domain representation of the pre-sample acquisition filter (PSF).
\hat{H}	→	Fourier domain representation of the pre-sample acquisition filter (OTF).
g	→	spatial domain representation of the pre-sample blurred scene.
\hat{G}	→	Fourier domain representation of the pre-sample blurred scene.
e	→	spatial domain representation of the Gaussian noise matrix.
\hat{e}	→	Fourier domain representation of the Gaussian noise matrix.
σ_e	→	Standard deviation of e .
$S()$	→	Sampling function.
p	→	spatial domain representation of the sampled image matrix.
\hat{p}	→	Fourier domain representation of the sampled image matrix.
f	→	spatial domain representation of the CLS restoration filter.
\hat{f}	→	Fourier domain representation of the CLS restoration filter.
\hat{C}	→	Fourier domain representation of the CLS restoration filter smoothing function.
α	→	CLS restoration filter parameter.
q	→	spatial domain representation of the filtered image matrix.
\hat{q}	→	Fourier domain representation of the filtered image matrix.
d	→	spatial domain representation of the reconstruction kernel.
\hat{D}	→	Fourier domain representation of the reconstruction kernel (RTF).
r	→	spatial domain representation of the reconstructed output image.
\hat{R}	→	Fourier domain representation of the reconstructed output image.
$()^*$	→	Complex conjugate.
\otimes	→	Circular Convolution operator (continuous or discrete).
Ω_s	→	Representation Passband: set of frequencies used to define \hat{S} .
Ω_h	→	Acquisition Passband: set of frequencies associated with the OTF values not significantly different from zero.
Ω_d	→	Reconstruction Passband: set of frequencies associated with the RTF values not significantly different from zero.
Ω_p	→	Sampling Passband: set of frequencies associated with the sampled image.

- $\tau_1, \tau_2 \rightarrow$ constants, generally a power of two, indicating the maximum frequency indexes used in determining Ω_s .
- $N_1, N_2 \rightarrow$ number of samples in the x_1 and x_2 directions, respectively.
- $\xi_1, \xi_2 \rightarrow$ intersample distance in the x_1 and x_2 directions, respectively.
- $P_1, P_2 \rightarrow$ period in the x_1 and x_2 directions, respectively. Defined to be $P_i = N_i \xi_i$.
- $\nu_1, \nu_2 \rightarrow$ frequency domain coordinates (discrete).
- $\omega_1, \omega_2 \rightarrow$ frequency domain coordinates (continuous) $\omega_i = \nu_i / P_i$.
- $x_1, x_2 \rightarrow$ spatial domain coordinates (continuous).
- $n_1, n_2 \rightarrow$ spatial domain coordinates (discrete).
- $\rho \rightarrow$ bins per pixel parameter.

1.4 Overview

The remainder of this dissertation is organized as follows. In Chapter 2, a detailed description of the *continuous-discrete-continuous* c/d/c system model is presented. The model described will be used for all results presented in this dissertation. A critical component of the c/d/c model is the digital restoration filter. Thus in Chapter 3, a derivation of a specific restoration filter, the CLS restoration filter, will be presented. The CLS restoration filter requires a priori knowledge of certain components of the system model including the frequency response of the acquisition device. In Chapter 4, a technique to estimate the frequency response of an acquisition device is presented. This technique is employed on an actual acquisition device and the frequency response (or optical transfer function (OTF)) is estimated. One step in the technique presented in Chapter 4 involves estimating the edge of a noisy, undersampled edge profile. Chapter 5 presents various algorithms for estimating the location of an edge based on an edge profile. The algorithms presented are compared based on accuracy, simplicity, and processing time. Chapter 6 introduces microscanning. Microscanning is presented as a means of suppressing aliased noise within an acquired digital image. In addition, the chapter discusses the effects of including microscanning in the c/d/c model. Chapter 7 presents results using real data from a digital JVC camcorder along with conclusions.

Chapter 2

C/D/C Model Description

The *continuous* input / *discrete* processing / *continuous* output (c/d/c) digital image processing system model indicated in Figure 2.1 is the basis for all the results in this dissertation.[42]

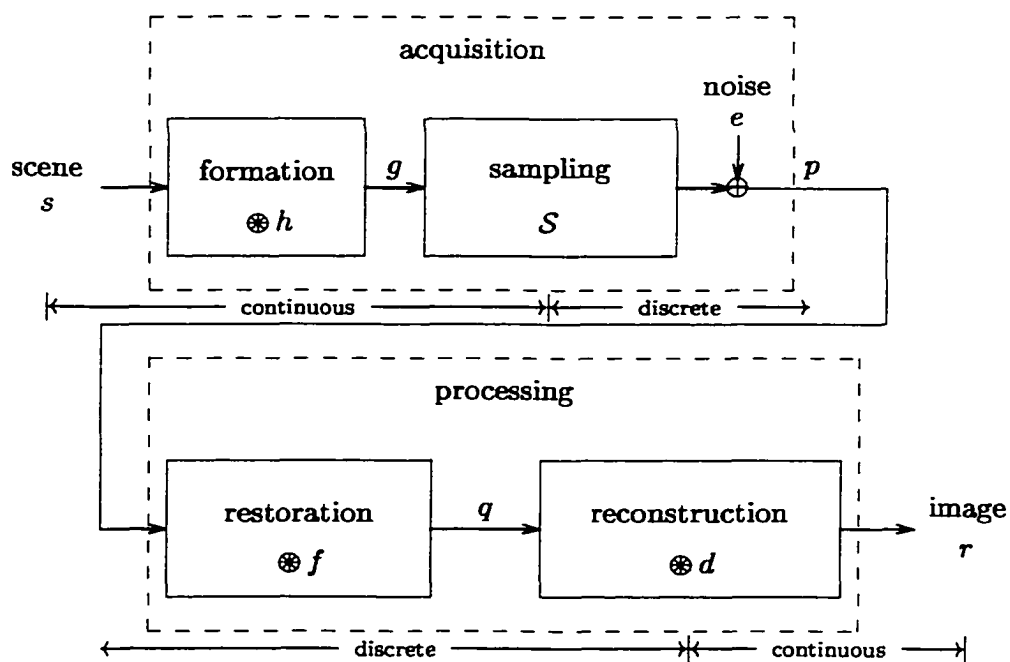


Figure 2.1: System Model

The acquired *scene* s is the 2-dimensional projection of the 3-dimensional field of view onto the acquisition device. The acquisition device (or image formation system)[42] blurs the acquired scene by convolution (shown mathematically in this research as \otimes) with the acquisition device point spread function (PSF) h . The resulting pre-sample image $g = s \otimes h$ is then sampled onto an $N_1 \times N_2$ sampling grid. N_1 and N_2 correspond to the number of detectors per row and column respectively on the acquisition device. ξ_1 and ξ_2 represent the distances between adjacent samples (or “intersample” distance) in the x_1 and x_2 directions respectively. Thus, $N_1 \times N_2$ is the number of detectors on the array. This sampled image, corrupted by additive random noise e , forms the digital image $p = S(g) + e$. The additive random noise e represents various effects of the acquisition process, including but not limited to quantization error.

The restoration filter f is the only component in the c/d/c model that is used to perform digital processing. The other filters in the model, h and d , are generally associated with hardware devices and cannot be altered except by changing the hardware configuration. The restoration filter f is a discrete processing filter defined by its software implementation. The restored digital image $q = p \otimes f$ is then convolved with the reconstruction filter d . The result of this convolution is the output image $r = q \otimes d$.

Sampling transforms the data from a continuous representation to a discrete representation. The image formation and reconstruction filters blur the image before and after sampling respectively. The restoration filter provides the digital filtering required to improve spatial resolution in the image by attempting to correct for this blurring.

2.1 Coordinate System

As with Park and Rahman[46], all system model components are measured relative to a common spatial (x_1, x_2) Cartesian coordinate system. By tradition, the coordinate system is assumed to be an orthogonal (x_1, x_2) system in which the intersample distances ξ_1, ξ_2 are constant along both axes, as illustrated in Figure 2.2. The physical center of the detectors falls along the sampling grid at pixel coordinates $[n_1, n_2]$ as indicated by the ‘•’.

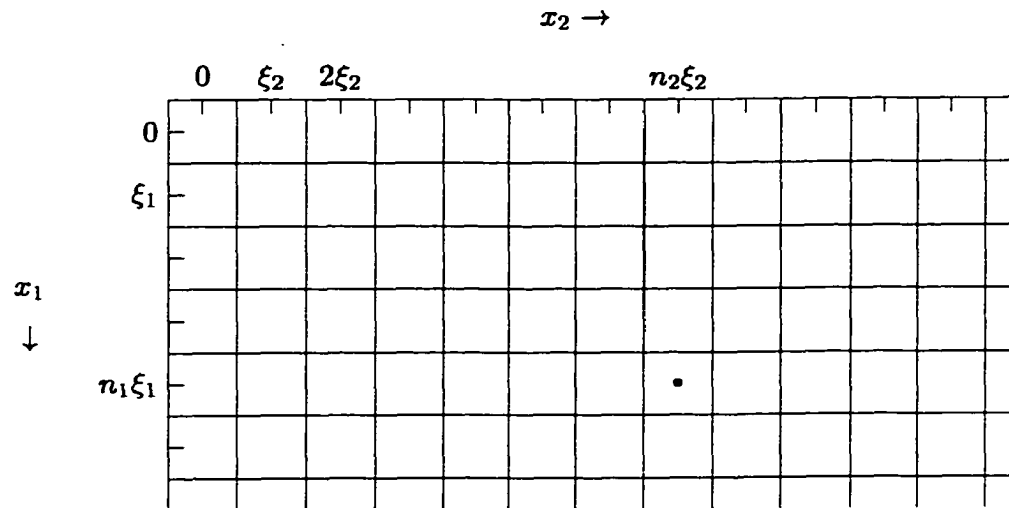


Figure 2.2: Coordinate System

With respect to the 2-dimensional projection, only the field-of-view (FOV) within the $N_1 \times N_2$ detector array is of interest.

2.2 Use of Fourier domain

In the Fourier (or frequency) domain convolution is performed as a frequency by frequency multiplication. Therefore, in a simulation environment it is preferable to perform these convolutions in the frequency domain.

2.3 C/D/C Model

2.3.1 Scene Representation

By assuming that the acquired scene within the field-of-view is periodic with period $P_1 \times P_2$, where $P_1 = N_1 \xi_1$ and $P_2 = N_2 \xi_2$, the input scene s can be defined in terms of its Fourier series coefficients¹

$$\hat{S}[\nu_1, \nu_2] = \frac{1}{P_1 P_2} \int_0^{P_1} \int_0^{P_2} s(x_1, x_2) \exp(-i2\pi x_1 \nu_1 / P_1) \exp(-i2\pi x_2 \nu_2 / P_2) dx_1 dx_2. \quad (2.1)$$

That is, because this assumption causes no significant error (except perhaps at the boundaries of the FOV), the input scene can be represented as the Fourier series [46]

$$s(x_1, x_2) = \sum_{|\nu_1| \leq \infty} \sum_{|\nu_2| \leq \infty} \hat{S}[\nu_1, \nu_2] \exp(i2\pi x_1 \nu_1 / P_1) \exp(i2\pi x_2 \nu_2 / P_2) \quad \forall (x_1, x_2). \quad (2.2)$$

Only a finite number of terms in the Fourier series $\hat{S}[\nu_1, \nu_2]$ are significantly different from zero. Thus, by restricting the summations to frequencies within the scene *representation passband* [46]

$$\Omega_s = \{(\omega_1, \omega_2) : |\omega_1| \leq \tau_1 / P_1, |\omega_2| \leq \tau_2 / P_2\}, \quad (2.3)$$

the scene is assumed to be band-limited to Ω_s , where τ_1 and τ_2 are implementation defined cut-off frequency constants. That is,

$$s(x_1, x_2) = \sum_{|\nu_1| \leq \tau_1} \sum_{|\nu_2| \leq \tau_2} \hat{S}[\nu_1, \nu_2] \exp(i2\pi x_1 \nu_1 / P_1) \exp(i2\pi x_2 \nu_2 / P_2) \quad \forall (x_1, x_2). \quad (2.4)$$

¹A distinction is made between *functions*, say $s(x_1, x_2)$, defined for a continuum of (real) values $-\infty < x_1, x_2 < \infty$ and *matrices*, say $\hat{S}[\nu_1, \nu_2]$, defined for the discrete (integer) values $\nu_1, \nu_2 = 0, \pm 1, \pm 2, \dots$. This “(·) for continuous” versus “[·] for discrete” notation is used throughout.

Real scenes are generally not band-limited, but by choosing τ_1/P_1 and τ_2/P_2 to be several times greater than the Nyquist frequency of the acquisition device, the above definition does not significantly affect the accuracy of the model. The frequencies within the representation passband are shown by the shaded area in Figure 2.3.

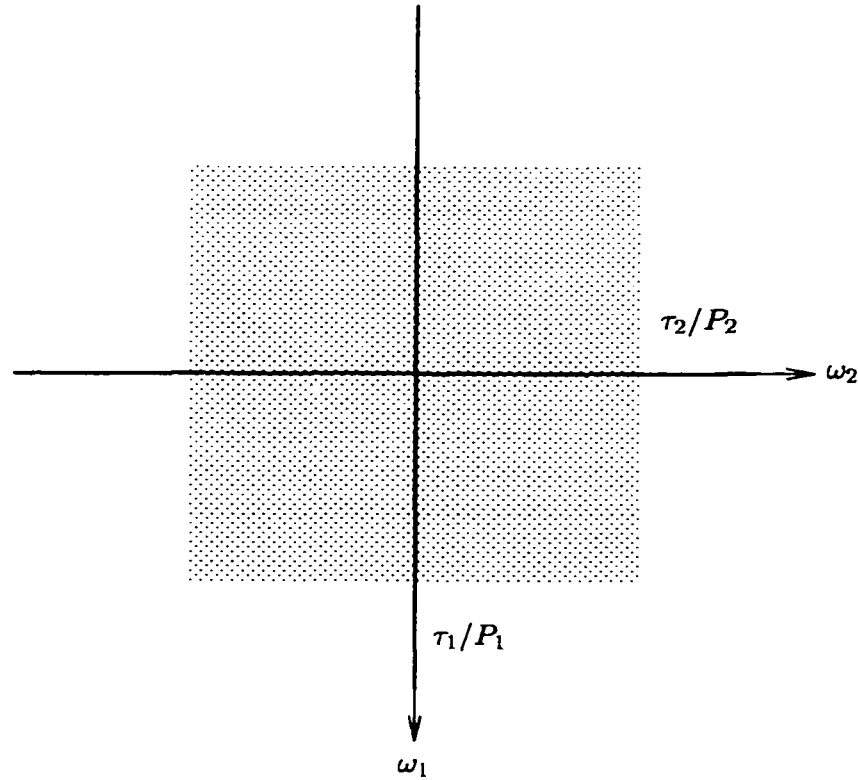


Figure 2.3: Representation Passband, Ω_s

As stated previously, the parameters N_1 and N_2 represent the number of detector elements within the FOV. Thus, if a particular acquisition device is to be simulated, N_1 and N_2 are known. In a simulation environment, however, N_1 and N_2 are generally set equal to each other and to a power of two. Although there is no fundamental reason for this power-of-two rule, it does facilitate the use of a Fast Fourier Transform (FFT) for conversions between the frequency domain and the spatial domain. For similar reasons, it is also

preferable to define τ_1 and τ_2 to be powers of two. [46]

Using the defined values of N_1 , N_2 , τ_1 , and τ_2 , the input scene can be specified by a $(2\tau_1 + 1) \times (2\tau_2 + 1)$ complex-valued Fourier coefficient array $\hat{S}[\nu_1, \nu_2]$. Because the input scene is real-valued, the Fourier coefficient array is subject to the complex-conjugate constraint $\hat{S}[-\nu_1, -\nu_2] = \hat{S}^*[\nu_1, \nu_2]$. [46]

2.3.2 Image Formation

The pre-sample image formation system response is completely characterized by a non-negative[59] *point spread function* (PSF) h normalized so that

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x_1, x_2) dx_1 dx_2 = 1. \quad (2.5)$$

Given an input *scene function* s , the (pre-sample) blurred image of this input scene is the *image function* $g = s \otimes h$ defined by convolution as

$$g(x_1, x_2) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} s(x_1 - x'_1, x_2 - x'_2) h(x'_1, x'_2) dx'_1 dx'_2 \quad \forall (x_1, x_2). \quad (2.6)$$

Because s is periodic, the convolved image function g is also periodic with period $P_1 \times P_2$.

The Fourier domain representation of Equation 2.6 is

$$\hat{G}[\nu_1, \nu_2] = \hat{S}[\nu_1, \nu_2] \hat{H}(\nu_1/P_1, \nu_2/P_2) \quad (\nu_1/P_1, \nu_2/P_2) \in \Omega_s, \quad (2.7)$$

$\nu_1, \nu_2 = 0, \pm 1, \pm 2, \dots$, where $\hat{H}(\omega_1, \omega_2)$ is the Fourier transform of $h(x_1, x_2)$ defined as

$$\hat{H}(\omega_1, \omega_2) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x_1, x_2) \exp(-i2\pi\omega_1 x_1) \exp(-i2\pi\omega_2 x_2) dx_1 dx_2 \quad \forall (\omega_1, \omega_2). \quad (2.8)$$

The convolution $s \otimes h$ simulates the blurring associated with all of the objects (lenses, detectors, etc.) which filter the image *prior* to sampling. $\hat{H}(\omega_1, \omega_2)$ is the cascaded product of each of the frequency responses associated with these devices. [46]

The *acquisition passband* is defined as the set of frequencies for which $\hat{H}(\omega_1, \omega_2)$ is significantly different from zero. This definition can also be represented as

$$\Omega_h = \{(\omega_1, \omega_2) : |\hat{H}(\omega_1, \omega_2)| \geq \epsilon\} \quad (2.9)$$

where ϵ is a small (application dependent) non-negative real-valued number. Figure 2.4 shows the relationship between the acquisition passband Ω_h and the representation passband Ω_s . The acquisition passband is depicted by the frequencies which fall within the circle².

²In practice this passband does not necessarily take a circular shape.

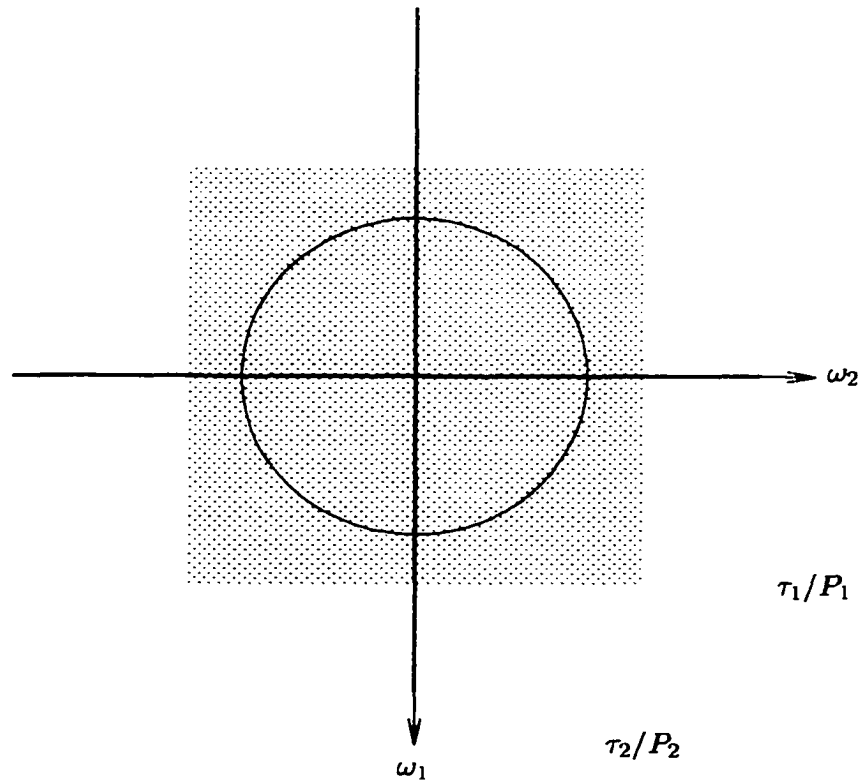


Figure 2.4: Acquisition Passband, Ω_h (relative to Ω_s - shaded region)

If τ_1 and τ_2 are chosen properly, the acquisition passband will fall within the representation passband so that $\Omega_h \subset \Omega_s$, as indicated. Once the scene has been acquired, all image energy at the frequencies which fall outside of the acquisition passband is lost. This loss of energy causes a loss of small scale features in the restored digital image. The energy at frequencies inside the acquisition passband is modulated subject to the OTF of the acquisition device. This modulation causes blurring.

2.3.3 Sampling

The image function g is sampled at integer multiples of the intersample distances ξ_1 and ξ_2 . Note that these intersample distances are particularly important parameters; $\xi_1 \times \xi_2$

represents the effective size of a pixel, or equivalently $1/2\xi_1$ represents the system's Nyquist frequency along the ω_1 axis and $1/2\xi_2$ is the Nyquist frequency along the ω_2 axis. These two quantities do not have to be equal, but for all the simulation results presented in this dissertation they are. Some acquisition devices contain square detectors, which makes this a valid assumption. If, however, the detectors are not square, the research presented generalizes to the use of the actual values of ξ_1 and ξ_2 .

2.3.4 Noise

Additive random system noise, which simulates error from various sources including quantization error, is modeled as a real-valued additive *noise array* e . The $N_1 \times N_2$ output *image array* $p = S(g) + e$ is defined by [46]

$$p[n_1, n_2] = g(n_1\xi_1, n_2\xi_2) + e[n_1, n_2] \quad \forall [n_1, n_2]. \quad (2.10)$$

Consistent with the periodicity of s and g , the digital image p is periodic with period $N_1 \times N_2$. The corresponding frequency domain representation of p is

$$\hat{p}[\nu_1, \nu_2] = \hat{G}[\nu_1, \nu_2] + \hat{a}[\nu_1, \nu_2] + \hat{e}[\nu_1, \nu_2] \quad (2.11)$$

for all $[\nu_1, \nu_2]$ where

$$\hat{p}[\nu_1, \nu_2] = \frac{1}{N_1 N_2} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} p[n_1, n_2] \exp(-i2\pi\nu_1 n_1/N_1) \exp(-i2\pi\nu_2 n_2/N_2) \quad (2.12)$$

and \hat{e} is the Discrete Fourier Transform (DFT) of the noise array e defined as

$$\hat{e}[\nu_1, \nu_2] = \frac{1}{N_1 N_2} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} e[n_1, n_2] \exp(-i2\pi\nu_1 n_1/N_1) \exp(-i2\pi\nu_2 n_2/N_2). \quad (2.13)$$

The *aliased noise* array \hat{a} is a noise component which exists due to undersampling and is defined by

$$\hat{a}[\nu_1, \nu_2] = \sum_{[k_1, k_2] \neq [0, 0]} \hat{G}[\nu_1 - k_1 N_1, \nu_2 - k_2 N_2]$$

for all $[\nu_1, \nu_2]$. This additive noise component accounts for energy which falls outside the sampling passband and is “folded” back onto frequencies inside the sampling passband. The *sampling passband*[46] is defined as

$$\Omega_p = \{(\omega_1, \omega_2) : |\omega_1| \leq 1/2\xi_1, |\omega_2| \leq 1/2\xi_2\}$$

where $1/2\xi_1$ and $1/2\xi_2$ are the Nyquist frequencies of the acquisition device. If the sampling density is *sufficient*, meaning $\Omega_h \subset \Omega_p$, aliasing will not occur in the reconstructed image. However, if $\Omega_h \not\subset \Omega_p$, the sampling density is *insufficient* and aliasing will occur. In Figure 2.5 the dashed square is the representation passband Ω_s and the circle is the acquisition passband Ω_h . The sampling passband is represented by the solid square. Of importance is the relationship between the sampling passband and the acquisition passband.

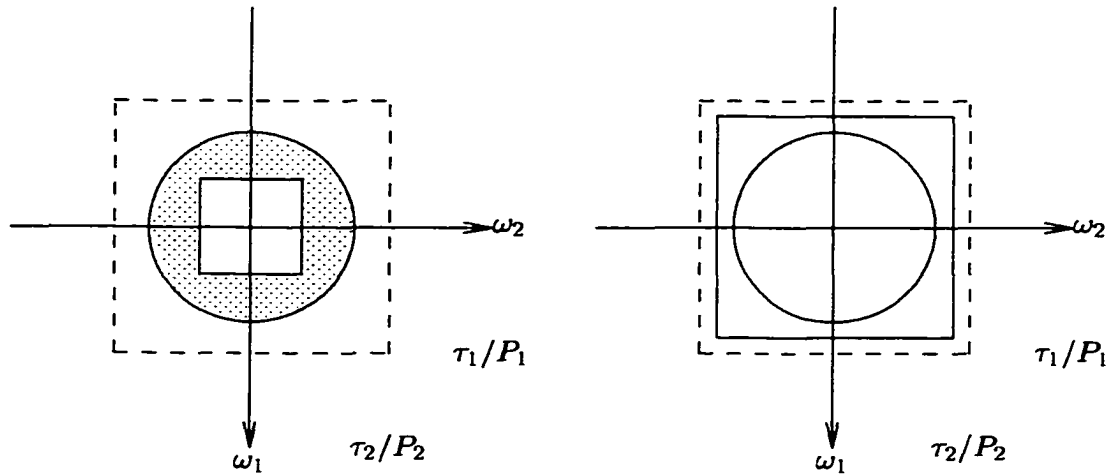


Figure 2.5: Sampling Passband, Ω_p , solid square (relative to Ω_h - solid circle, and Ω_s - dotted square)

The example on the left, where the sampling passband is a subset of the acquisition passband, depicts an instance where aliasing will be present in the sampled image. Specifically, the shaded region of the figure indicates the frequencies from which energy will be folded back into the sampling passband. The example on the right shows a case where sampling is sufficient and aliasing will not occur. This is an unusual case in image acquisition. Generally the sampling density is insufficient. Natural scenes generally contain high-frequency, sub-pixel-scale features.[46] The effects of aliasing will be investigated here and a corrective measure will be discussed.

In addition to aliasing, quantization error and random electronic noise associated with the image acquisition device both limit the ability of the restoration filter to accurately restore the input scene. The pre-sampled image g consists of real-valued data. The digital image p is an integer-valued array. Thus, a quantization of the sampled values

$$p[n_1, n_2] = \lfloor g(n_1\xi_1, n_2\xi_2) + 0.5 \rfloor$$

causes an additional noise component in the digital image ($\lfloor \cdot \rfloor$ is the floor (truncation) function). The difference

$$g(n_1\xi_1, n_2\xi_2) - p[n_1, n_2]$$

is termed *quantization noise* and can be characterized by a signal-to-noise ratio (SNR).[46]

The SNR quantifies the level of additive random noise relative to the input scene. SNR is defined to be the ratio of the standard deviations of s and e [46], or σ_s/σ_e where

$$\sigma_s^2 = \left(\sum_{|\nu_1| \leq \tau_1} \sum_{|\nu_2| \leq \tau_2} |\hat{S}[\nu_1, \nu_2]|^2 \right) - |\hat{S}[0, 0]|^2$$

and

$$\sigma_e^2 = \left(\sum_{\nu_1=0}^{N_1-1} \sum_{\nu_2=0}^{N_2-1} |\hat{e}[\nu_1, \nu_2]|^2 \right) - |\hat{e}[0, 0]|^2.$$

Note that the SNR quantifies only the relative strength of the *random* noise in the image with respect to the strength of the signal. The amount of *aliased* noise is determined in part by the relationship (depicted in Figure 2.5) between the acquisition passband Ω_h and the sampling passband Ω_p . The *aliased* noise could be more or less prominent than the *random* noise.

2.3.5 Image Restoration

The $N_1 \times N_2$ digital image p is filtered by convolution with a *restoration filter* f to produce the *filtered image array* $q = p \otimes f$ defined by[46]

$$q[n_1, n_2] = \sum_{n'_2=-\infty}^{\infty} \sum_{n'_1=-\infty}^{\infty} f[n'_1, n'_2] p[n_1 - n'_1, n_2 - n'_2]. \quad (2.14)$$

Traditionally image restoration filters have been defined and implemented in the frequency domain. Thus, image restoration performed in the frequency domain is based on the equation

$$\hat{q}[\nu_1, \nu_2] = \hat{p}[\nu_1, \nu_2] \hat{f}[\nu_1, \nu_2] \quad (2.15)$$

where

$$\hat{f}[\nu_1, \nu_2] = \sum_{n_1=-\infty}^{\infty} \sum_{n_2=-\infty}^{\infty} f[n_1, n_2] \exp(-i2\pi\nu_1 n_1/N_1) \exp(-i2\pi\nu_2 n_2/N_2) \quad \forall(\nu_1, \nu_2). \quad (2.16)$$

Note that the function \hat{f} is periodic with 2-D period 1×1 and that the frequencies $[\nu_1, \nu_2] = (\nu_1/N_1, \nu_2/N_2) = (\xi_1\omega_1, \xi_2\omega_2)$ are measured in units of cycles *per sample*. [46] Since p is periodic with period $N_1 \times N_2$, the filtered digital image q is also periodic with the same period.

2.3.6 Image Reconstruction

The *output image* r is reconstructed by convolving the *filtered image array* q with an aperiodic *reconstruction kernel* d . Image reconstruction can either mean displaying the restored image to a display device or interpolating the image onto a different coordinate grid. In either case, the image reconstruction system models the discrete-to-continuous conversion. [46] For the simulation model presented here, the reconstruction kernel is an interpolation process defined as

$$r(x_1, x_2) = \sum_{n_1=-\infty}^{\infty} \sum_{n_2=-\infty}^{\infty} q[n_1, n_2] d(x_1 - n_1\xi_1, x_2 - n_2\xi_2) \quad (2.17)$$

for all (x_1, x_2) . The image reconstruction filter d is characterized by a parametric cubic convolution function[45] normalized so that

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} d(x_1, x_2) dx_1 dx_2 = \xi_1 \xi_2. \quad (2.18)$$

Because r is periodic with period $P_1 \times P_2$ it can be represented by its Fourier coefficients as

$$r(x_1, x_2) = \sum_{|\nu_1| \leq \tau_1} \sum_{|\nu_2| \leq \tau_2} \hat{R}[\nu_1, \nu_2] \exp(i2\pi x_1 \nu_1 / P_1) \exp(i2\pi x_2 \nu_2 / P_2) \quad (2.19)$$

for all (x_1, x_2) where

$$\hat{R}[\nu_1, \nu_2] = \left(\frac{1}{\xi_1 \xi_2} \right) \hat{q}[\nu_1, \nu_2] \hat{D}(\nu_1 / P_1, \nu_2 / P_2) \quad (\nu_1 / P_1, \nu_2 / P_2) \in \Omega_s. \quad (2.20)$$

Note that the Fourier coefficients used in Equation 2.20 are based on frequency indexes which correspond to frequencies within the representation passband Ω_s . This is valid provided τ_1, τ_2 are sufficiently large relative to N_1, N_2 . [46] The image reconstruction filter is similar to the image formation filter in that they are both low-pass filters. Thus, if the *reconstruction passband* is defined to be

$$\Omega_d = \{(\omega_1, \omega_2) : |\hat{D}(\omega_1, \omega_2)| \geq \epsilon\}, \quad (2.21)$$

then the software defined constants τ_1 and τ_2 must be chosen so that $\Omega_h \subset \Omega_s$ and $\Omega_d \subset \Omega_s$.

The reconstruction transfer function (RTF) \hat{D} is defined to be [46]

$$\hat{D}(\omega_1, \omega_2) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} d(x_1, x_2) \exp(-i2\pi\omega_1 x_1) \exp(-i2\pi\omega_2 x_2) dx_1 dx_2 \quad \forall (\omega_1, \omega_2). \quad (2.22)$$

For purposes of this dissertation, reconstruction of a filtered digital image will mean interpolating the sampled data onto a sampling grid which has a higher density than $N_1 \times N_2$ within the same $P_1 \times P_2$ period. This interpolation will constitute simulating the transformation from a $N_1 \times N_2$ discrete array to a continuous scene.

2.4 Output Image Components

As defined in Equation 2.11, \hat{p} is the summation of three components. This relationship is shown in Figure 2.6.

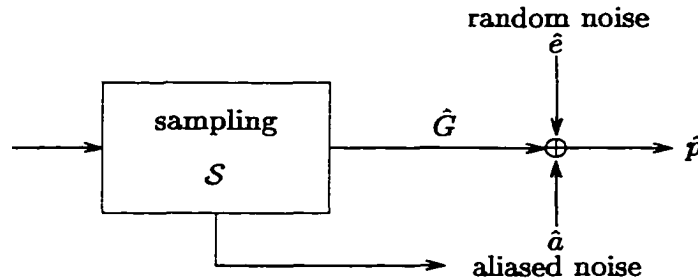


Figure 2.6: Components of \hat{p}

In a simulation environment, each of these components can be independently processed through a restoration filter and a reconstruction filter.

We define \hat{p}_a to be the component of \hat{p} that accounts for the aliased noise (previously defined as \hat{a}). Also \hat{p}_e is defined to be the component of \hat{p} that accounts for the quantization error and random electronic noise (previously defined as \hat{e}). And, \hat{p}_c is defined to be the

component of \hat{p} that accounts for the signal degraded by the OTF (previously defined as \hat{G}). Thus,

$$\hat{p} = \hat{p}_c + \hat{p}_e + \hat{p}_a.$$

The components \hat{p}_c , \hat{p}_e , and \hat{p}_a are stored as $N_1 \times N_2$ arrays³. Therefore, due to the linearity of f and d each component can be independently processed through a restoration filter and a reconstruction filter.

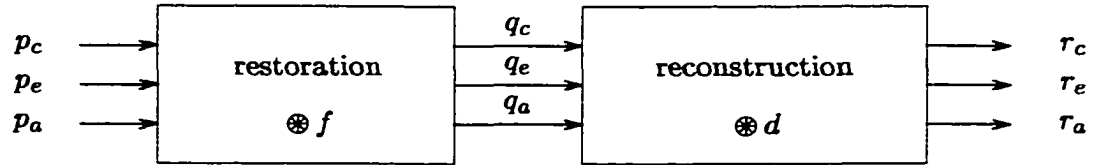


Figure 2.7: Processed Components of p

That is,

$$\hat{q}_c[\nu_1, \nu_2] = \hat{p}_c[\nu_1, \nu_2] \hat{f}[\nu_1, \nu_2],$$

$$\hat{q}_e[\nu_1, \nu_2] = \hat{p}_e[\nu_1, \nu_2] \hat{f}[\nu_1, \nu_2],$$

$$\hat{q}_a[\nu_1, \nu_2] = \hat{p}_a[\nu_1, \nu_2] \hat{f}[\nu_1, \nu_2]$$

where \hat{f} is defined by Equation 2.16 and

$$\hat{R}_c[\nu_1, \nu_2] = \left(\frac{1}{\xi_1 \xi_2} \right) \hat{q}_c[\nu_1, \nu_2] \hat{D}(\nu_1/P_1, \nu_2/P_2),$$

$$\hat{R}_e[\nu_1, \nu_2] = \left(\frac{1}{\xi_1 \xi_2} \right) \hat{q}_e[\nu_1, \nu_2] \hat{D}(\nu_1/P_1, \nu_2/P_2),$$

³Although the \hat{p}_a is not periodic, the sum $\langle \hat{G} \rangle[\nu_1, \nu_2] = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} \hat{G}[\nu_1 - k_1 N_1, \nu_2 - k_2 N_2]$ is periodic with period $N_1 \times N_2$. Therefore, it is sufficient to store $\langle \hat{G} \rangle$ and compute \hat{p}_a as needed.[46]

$$\hat{R}_a[\nu_1, \nu_2] = \left(\frac{1}{\xi_1 \xi_2} \right) \hat{q}_a[\nu_1, \nu_2] \hat{D}(\nu_1/P_1, \nu_2/P_2)$$

where \hat{D} is defined by Equation 2.22. Therefore, the output image components are defined to be r_c , r_e , and r_a [46] where

$$\tau = r_c + r_e + r_a$$

or equivalently in the Fourier domain

$$\hat{R} = \hat{R}_c + \hat{R}_e + \hat{R}_a.$$

Defining the output image in terms of its components not only presents a means of visually representing the impact of the components on the output image but also provides an alternate definition of the signal-to-noise ratio (SNR). Specifically \hat{R}_c/\hat{R}_e defines the signal-to-random-noise ratio, \hat{R}_c/\hat{R}_a defines the signal-to-aliased-noise ratio, and $\hat{R}_c/(\hat{R}_e + \hat{R}_a)$ defines the signal-to-system-noise ratio.

2.5 End-to-end Metric

Defining \hat{R} based on the frequencies within the representation passband, as discussed in Section 2.3.6, also facilitates a one-to-one comparison of the Fourier coefficients of the input scene to the Fourier coefficients of the output image. This comparison can then be used as the basis for an end-to-end metric that quantifies the effectiveness of the subpixel techniques presented in this dissertation.

If the mean square error,

$$\|s - r\|^2 = \frac{1}{P_1 P_2} \int_0^{P_1} \int_0^{P_2} |s(x_1, x_2) - r(x_1, x_2)|^2 dx_1 dx_2,$$

between the input scene and the output image is used, then an equivalent expression can be computed using the Fourier coefficients.

As stated previously, the system model computations for the simulations and results presented are done in the Fourier domain. Therefore, computation of the metric could also be done in the Fourier domain. The root mean square error (RMSE) metric is used to determine the effectiveness of the restoration methods. Using Parseval's equation and the fact that s and r are assumed to be band-limited to Ω_s , the RMSE is defined as

$$\|s - r\| = \sqrt{\|s - r\|^2} = \sqrt{\sum_{|\nu_1| \leq \tau} \sum_{|\nu_2| \leq \tau} |\hat{S}[\nu_1, \nu_2] - \hat{R}[\nu_1, \nu_2]|^2}. \quad (2.23)$$

All quantized error values described in this research are computed based upon this RMSE metric.

In the next chapter we define the “non-unity intersample distance” Constrained Least Squares restoration filter. The CLS restoration filter is a robust filter which avoids amplification of noise in the restored image (a feature inherent in an inverse filter) without a reliance on theoretical values (a requirement of the Wiener restoration filter)[14, 39].

Chapter 3

Constrained Least Squares Filter

The restoration filter f is the only component of the $c/d/c$ model that is used to digitally process the image. All other $c/d/c$ filters are typically related to hardware devices. Snyder et al[58] state that “restoration is normally performed on the precorrected image to recover the object’s intensity distribution by compensating for blurring and aberrations that are caused by the optical elements.” Successful restoration of a degraded image relies upon a properly designed and implemented restoration filter. Blurring, which may occur during acquisition and reconstruction, must be removed if a visually acceptable reconstruction of the input scene is to be achieved. The restoration filter proposed here is the constrained least-squares filter. A derivation of this filter that assumes unity intersample distances can be found in [18]. A similar derivation which accounts for non-unity intersample distances ξ_1, ξ_2 follows.

3.1 Filter Derivation

As in [18], the CLS restoration filter is based on the metric,

$$J^2 = \alpha S^2 + F^2.$$

This metric consists of a *smoothness* term

$$S^2 = \|c \otimes r\|^2 = \frac{1}{P_1 P_2} \int_0^{P_1} \int_0^{P_2} |c(x_1, x_2) \otimes r(x_1, x_2)|^2 dx_1 dx_2,$$

a *fidelity* term

$$F^2 = \|p - p'\|^2 = \frac{1}{N_1 N_2} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} |p[n_1, n_2] - p'[n_1, n_2]|^2,$$

and a positive, real-valued, user defined parameter α that controls the amount of smoothness versus fidelity. c is a high-boost filter such as depicted in Figure 3.1

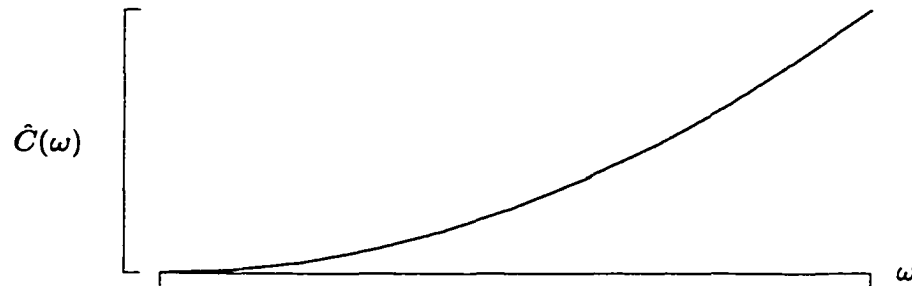


Figure 3.1: Generic High-Boost Filter

where \hat{C} represents the Fourier transform of c . \hat{C} is used to boost high frequency energy and suppress ringing in the restored image.¹ For the results presented in this research

$$\hat{C}(\omega_1, \omega_2) = \omega_1^2 + \omega_2^2.$$

¹The importance of the high-boost filter will be shown later in the chapter.

There is no fundamental reason for this choice of \hat{C} except that it is well behaved and mathematically simple. The digital image p' is produced by using the output image r as the input scene to the c/d/c model. That is,

$$\hat{p}'[\nu_1, \nu_2] = \langle \hat{R}[\nu_1, \nu_2] \hat{H}(\nu_1/P_1, \nu_2/P_2) \rangle \quad (3.1)$$

where the $\langle \cdot \rangle$ operation accounts for the frequencies which fall outside the sampling passband and are folded back inside the sampling passband, or equivalently,

$$\hat{p}'[\nu_1, \nu_2] = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} \hat{R}[\nu_1 - k_1 P_1, \nu_2 - k_2 P_2] \hat{H}(\nu_1/P_1 - k_1/\xi_1, \nu_2/P_2 - k_2/\xi_2). \quad (3.2)$$

A graphical depiction of p' is shown in Figure 3.2.

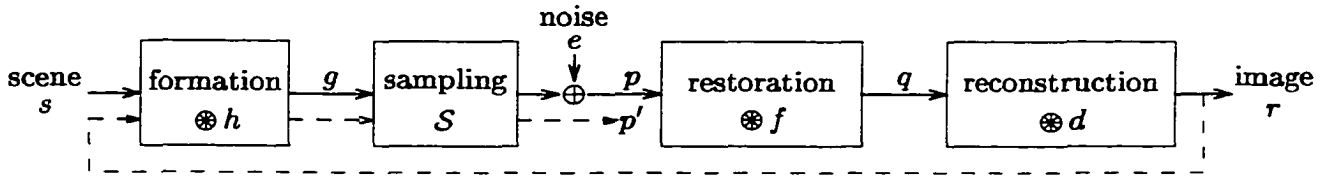


Figure 3.2: Definition of p'

Returning to the metric, the CLS restoration filter represents an attempt to achieve a compromise between acquired image sharpening F^2 and random noise enhancement S^2 [18].

This compromise is accomplished by minimization of

$$J^2 = \alpha \sum_{|\nu_1| \leq \tau_1} \sum_{|\nu_2| \leq \tau_2} \left| \hat{C}(\nu_1/P_1, \nu_2/P_2) \hat{R}[\nu_1, \nu_2] \right|^2 + \sum_{\nu_1=0}^{N_1-1} \sum_{\nu_2=0}^{N_2-1} |\hat{p}[\nu_1, \nu_2] - \hat{p}'[\nu_1, \nu_2]|^2$$

where \hat{C} is defined analogously to \hat{H} in Equation 2.8. The implications of this compromise will be shown in Section 3.2.3 (following the complete derivation of the CLS restoration filter).

The smoothness and fidelity terms can now be defined with respect to the unknown restoration filter coefficients $\hat{f}[\nu_1, \nu_2]$. Writing the smoothness component of the metric in terms of the restoration filter coefficients \hat{f} yields

$$S^2 = \sum_{\nu_1=0}^{N_1-1} \sum_{\nu_2=0}^{N_2-1} |\hat{p}[\nu_1, \nu_2]|^2 \left| \frac{1}{\xi_1 \xi_2} \hat{f}[\nu_1, \nu_2] \right|^2 \left\langle \hat{C}(\nu_1/P_1, \nu_2/P_2) \hat{D}(\nu_1/P_1, \nu_2/P_2) \right\rangle. \quad (3.3)$$

Using Equation 3.1, the following is derived.

$$\begin{aligned} \hat{p}'[\nu_1, \nu_2] &= \left\langle \hat{R}[\nu_1, \nu_2] \hat{H}(\nu_1/P_1, \nu_2/P_2) \right\rangle \\ &= \left\langle \frac{1}{\xi_1 \xi_2} \hat{q}[\nu_1, \nu_2] \hat{D}(\nu_1/P_1, \nu_2/P_2) \hat{H}(\nu_1/P_1, \nu_2/P_2) \right\rangle \quad \text{def. of } \hat{R} \\ &= \left\langle \frac{1}{\xi_1 \xi_2} \hat{p}[\nu_1, \nu_2] \hat{f}[\nu_1, \nu_2] \hat{D}(\nu_1/P_1, \nu_2/P_2) \hat{H}(\nu_1/P_1, \nu_2/P_2) \right\rangle \quad \text{def. of } \hat{q} \\ &= \frac{1}{\xi_1 \xi_2} \hat{p}[\nu_1, \nu_2] \hat{f}[\nu_1, \nu_2] \left\langle \hat{D}(\nu_1/P_1, \nu_2/P_2) \hat{H}(\nu_1/P_1, \nu_2/P_2) \right\rangle \end{aligned}$$

Similarly defining the fidelity component and using the above derivation of \hat{p}' yields

$$F^2 = \sum_{\nu_1=0}^{N_1-1} \sum_{\nu_2=0}^{N_2-1} |\hat{p}[\nu_1, \nu_2]|^2 \left| 1 - \frac{1}{\xi_1 \xi_2} \hat{f}[\nu_1, \nu_2] \left\langle \hat{H}(\nu_1/P_1, \nu_2/P_2) \hat{D}(\nu_1/P_1, \nu_2/P_2) \right\rangle \right|^2. \quad (3.4)$$

Combining Equations 3.3 and 3.4, J^2 can now be defined as the quadratic equation

$$\begin{aligned} J^2 &= \sum_{\nu_1=0}^{N_1-1} \sum_{\nu_2=0}^{N_2-1} |\hat{p}[\nu_1, \nu_2]|^2 \\ &\quad \left(1 - \hat{f}^*[\nu_1, \nu_2] \hat{B}(\nu_1/P_1, \nu_2/P_2) - \hat{f}[\nu_1, \nu_2] \hat{B}^*(\nu_1/P_1, \nu_2/P_2) + \right. \\ &\quad \left. |\hat{f}[\nu_1, \nu_2]|^2 \left| \hat{A}(\nu_1/P_1, \nu_2/P_2) \right|^2 \right) \end{aligned} \quad (3.5)$$

where

$$\hat{A}(\omega_1, \omega_2) = \frac{\alpha}{\xi_1^2 \xi_2^2} \left\langle \left| \hat{C}(\omega_1, \omega_2) \hat{D}(\omega_1, \omega_2) \right|^2 \right\rangle + \frac{1}{\xi_1^2 \xi_2^2} \left| \left\langle \hat{H}(\omega_1, \omega_2) \hat{D}(\omega_1, \omega_2) \right\rangle \right|^2$$

and

$$\hat{B}(\omega_1, \omega_2) = \frac{1}{\xi_1 \xi_2} \langle \hat{H}(\omega_1, \omega_2) \hat{D}(\omega_1, \omega_2) \rangle.$$

Minimizing the error associated with this metric is accomplished by completing the square

as

$$J^2 = J_{min}^2 + \sum_{\nu_1=0}^{N_1-1} \sum_{\nu_2=0}^{N_2-1} \hat{A}(\nu_1/P_1, \nu_2/P_2) |\hat{p}[\nu_1, \nu_2]|^2 |\hat{f}[\nu_1, \nu_2] - \hat{f}_c[\nu_1, \nu_2]|^2$$

where

$$J_{min}^2 = \sum_{\nu_1=0}^{N_1-1} \sum_{\nu_2=0}^{N_2-1} |\hat{p}[\nu_1, \nu_2]|^2 \left[1 - |\hat{f}_c[\nu_1, \nu_2]|^2 \hat{A}(\nu_1/P_1, \nu_2/P_2) \right]$$

The solution to this minimization can be shown to be

$$\hat{f}[\nu_1, \nu_2] = \frac{\hat{B}(\nu_1/N_1 \xi_1, \nu_2/N_2 \xi_2)}{\hat{A}(\nu_1/N_1 \xi_1, \nu_2/N_2 \xi_2)}$$

or, equivalently

$$\hat{f}[\nu_1, \nu_2] = \frac{\xi_1 \xi_2 \langle \hat{H}^* \hat{D}^* \rangle(\omega_1, \omega_2)}{\left| \langle \hat{H} \hat{D} \rangle(\omega_1, \omega_2) \right|^2 + \alpha \left| \langle \hat{C} \hat{D} \rangle(\omega_1, \omega_2) \right|^2} \quad (3.6)$$

where

$$\langle \hat{H}^* \hat{D}^* \rangle(\omega_1, \omega_2) = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} \hat{H}^*(\omega_1 - k_1/\xi_1, \omega_2 - k_2/\xi_2) \hat{D}^*(\omega_1 - k_1/\xi_1, \omega_2 - k_2/\xi_2)$$

$$\left| \langle \hat{C} \hat{D} \rangle(\omega_1, \omega_2) \right|^2 = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} \left| \hat{C}(\omega_1 - k_1/\xi_1, \omega_2 - k_2/\xi_2) \hat{D}(\omega_1 - k_1/\xi_1, \omega_2 - k_2/\xi_2) \right|^2$$

and $(\cdot)^*$ denotes the complex conjugate. Equation 3.6 represents the definition of the

Constrained Least Squares Restoration filter.

3.2 Importance of a Properly Defined System Model

A common problem in simulation-based restoration studies is the use of incomplete system models. Examples of this problem include a failure to account for the sampling that is inherent in the acquisition process and an assumption of a noise-free environment[32, 36, 53]. The following sections show why these incomplete models and assumptions can yield misleading or incorrect results.

3.2.1 C/C Model Results

A commonly used *continuous-input / continuous-output* (c/c) system model (no sampling component) is depicted in Figure 3.3.

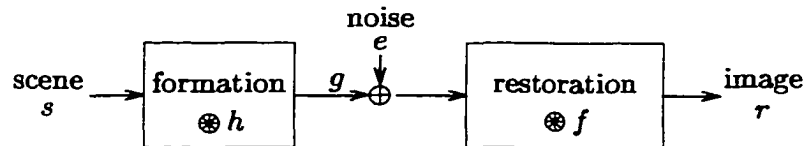


Figure 3.3: C/C System Model

From Equation 2.7, the Fourier series coefficients of the blurred image g are computed by multiplying the Fourier series coefficients of the input scene s with the Optical Transfer Function of the acquisition device. That is,

$$\hat{G}[\nu_1, \nu_2] = \hat{S}[\nu_1, \nu_2] \hat{H}(\nu_1/N_1, \nu_2/N_2) \quad (3.7)$$

where \hat{G} is the Fourier Series representation of the blurred image g and \hat{S} is the Fourier Series representation of the input scene s .

For the simulated results presented in this chapter the following aerial image of the Statue of Liberty was used as the input scene. This input scene (Figure 3.4) is a 512×512 8-bit grayscale image.



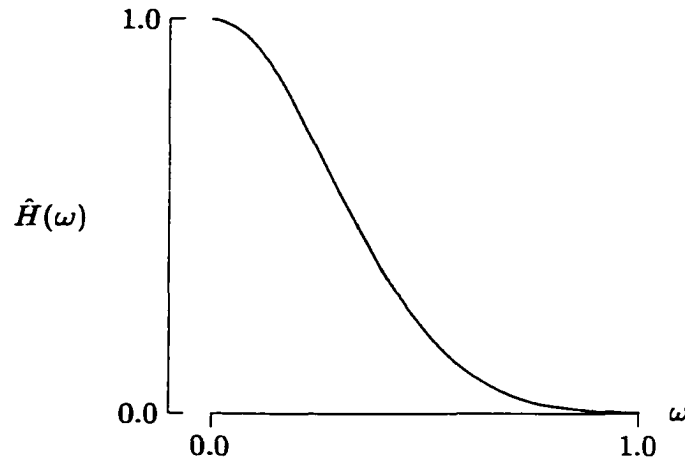
Figure 3.4: scene (s)

The OTF used to simulate the acquisition filter, depicted in Figure 3.5, is

$$\hat{H}(\omega_1, \omega_2) = \exp(-(\omega_1/\delta_1)^2) \exp(-(\omega_2/\delta_2)^2) \quad (3.8)$$

where $\delta_1 = \delta_2 = 0.4$.

Figure 3.6 is the parametric cubic convolution (PCC) reconstruction of the blurred image g prior to image restoration. In a noise-free system, an inverse restoration filter would be

Figure 3.5: OTF (\hat{H})

sufficient to restore the image. That is,

$$\hat{F}(\nu_1/N_1, \nu_2/N_2) = \frac{1}{\hat{H}(\nu_1/P_1, \nu_2/P_2)}. \quad (3.9)$$

Restoration would be achieved by multiplying the Fourier coefficients of the blurred image with the Fourier coefficients of the restoration filter or, equivalently, dividing the blurred image by the OTF.

$$\hat{R}[\nu_1, \nu_2] = \hat{G}[\nu_1, \nu_2] \hat{F}(\nu_1/N_1, \nu_2/N_2) = \frac{\hat{G}[\nu_1, \nu_2]}{\hat{H}(\nu_1/P_1, \nu_2/P_2)} \quad (3.10)$$

It is easily seen from Equations 3.7 and 3.10 that \hat{R} is equivalent to \hat{S} and thus the inverse restoration filter restores the blurred image to the input scene (Figure 3.7).

Imaging systems, however, are not noise-free. The discussion following shows that in practice simply dividing the acquired blurred image by the OTF will boost the noise that is present in the acquisition process. The resulting output image can be significantly degraded by the amplified noise, as shown in Figure 3.8.



Figure 3.6: OTF blurred ($g = s \otimes h$)

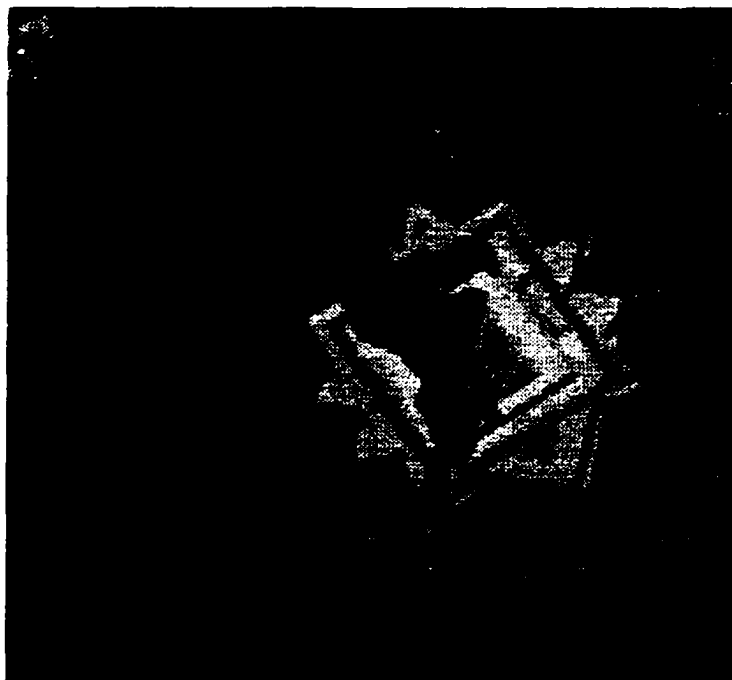


Figure 3.7: Inverse filter, no noise

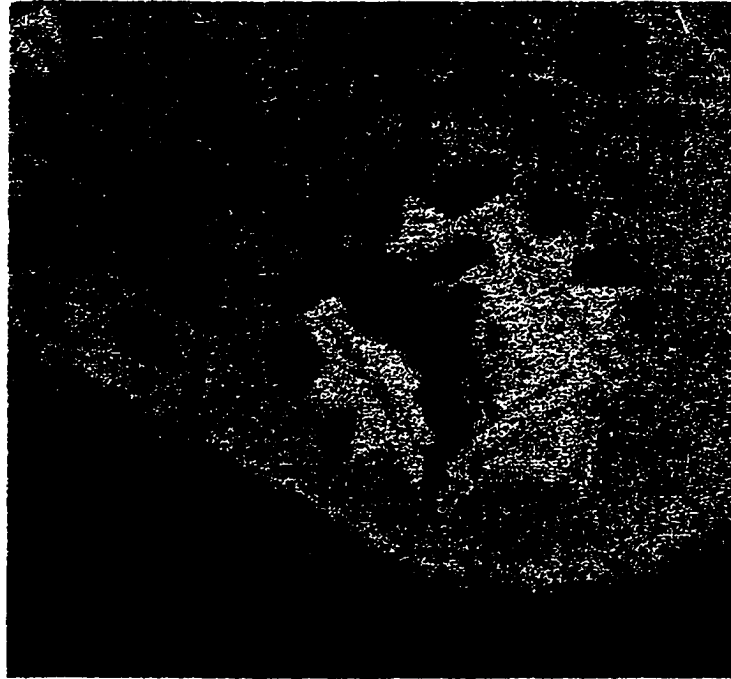


Figure 3.8: Inverse filter, with quantization noise only

The amplification of the noise can be seen through a close look at the equations that describe the acquisition and restoration process. The blurred image g is formed not only by convolution with the OTF but also with the additive random noise present in the acquisition process. That is, in a C/C system model

$$\hat{G}[\nu_1, \nu_2] = \hat{S}[\nu_1, \nu_2] \hat{H}(\nu_1/N_1, \nu_2/N_2) + \hat{e}[\nu_1, \nu_2] \quad (3.11)$$

where \hat{e} is the Fourier transform of the random additive noise array. Using the restored image obtained by the inverse filter for this model, gives

$$\hat{R}[\nu_1, \nu_2] = \frac{\hat{S}[\nu_1, \nu_2] \hat{H}(\nu_1/N_1, \nu_2/N_2) + \hat{e}[\nu_1, \nu_2]}{\hat{H}(\nu_1/N_1, \nu_2/N_2)} \quad (3.12)$$

which results in

$$\hat{R}[\nu_1, \nu_2] = \hat{S}[\nu_1, \nu_2] + \frac{\hat{e}[\nu_1, \nu_2]}{\hat{H}(\nu_1/N_1, \nu_2/N_2)}. \quad (3.13)$$

The energy present in \hat{S} at higher frequencies is negligible; however, the energy present in \hat{e} is constant at all frequencies for our assumed model of white noise. Because \hat{H} is small at high frequencies, the noise is amplified causing the restored image to have significant noise artifacts.

3.2.2 C/D/C Model Results

A more accurate model to use for simulation-based restoration studies is the *continuous input / discrete processing / continuous output* model (Figure 3.9). Acquisition of an input scene consists of sampling the scene onto the detectors of the acquisition device. The sampled digital image is discretely defined. Therefore, restoration is a discrete process.

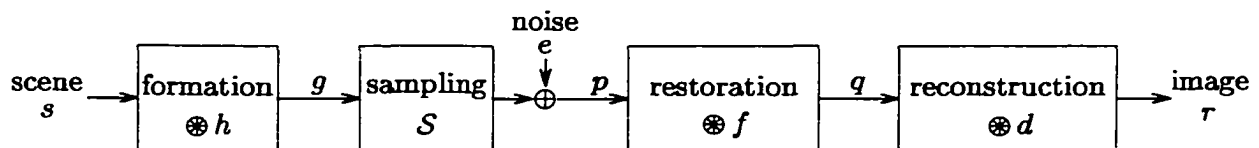


Figure 3.9: C/D/C System Model

The Constrained Least Squares filter defined in Section 3.1 is a discrete restoration filter defined in the Fourier domain and based on the acquisition filter, the reconstruction filter, a high boost filter component, and the acquisition sample size $N_1 \times N_2$.

By its definition, the CLS filter restores degraded images subject to minimization of a mean-square error metric relative to the entire input scene. The focus is not on specific types of scenes such as “star fields”[25] or individual components of the scene such as edge preservation[3, 38, 41]. By its design, the CLS filter restores degraded images while

suppressing noise amplification.² First, as a basis for comparison, in a noise free system, the CLS filter restores the image as well as the inverse filter (Figure 3.10).

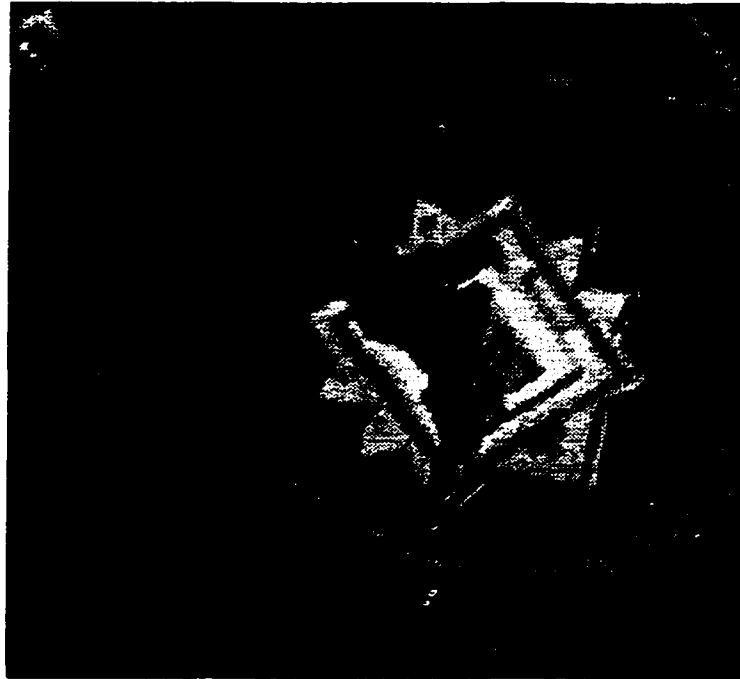


Figure 3.10: CLS filter, no noise

When noise is present, however, the CLS filter restores the image (Figure 3.11) without significantly amplifying the noise at higher frequencies.

3.2.3 CLS Suppression of Noise Enhancement

The suppression of noise enhancement is accomplished in the CLS restoration filter through the use of a high-boost filter. The CLS restoration filter,

$$\hat{f}[\nu_1, \nu_2] = \frac{\xi_1 \xi_2 \langle \hat{H}^* \hat{D}^* \rangle(\omega_1, \omega_2)}{|\langle \hat{H} \hat{D} \rangle(\omega_1, \omega_2)|^2 + \alpha \langle |\hat{C} \hat{D}|^2 \rangle(\omega_1, \omega_2)} \quad (3.14)$$

²Details of this noise suppression will be presented later in the chapter.



Figure 3.11: CLS filter, with noise

reduces to the inverse filter when $\alpha = 0$. As α is increased, energy at high frequencies (which constitutes mostly noise) is suppressed. This can be seen through the 1-D representation of the CLS filter in Figure 3.12. The dashed line represents the CLS filter with $\alpha = 0$ (or equivalently, the inverse filter). The solid line is the CLS filter with $\alpha = 0.2$.

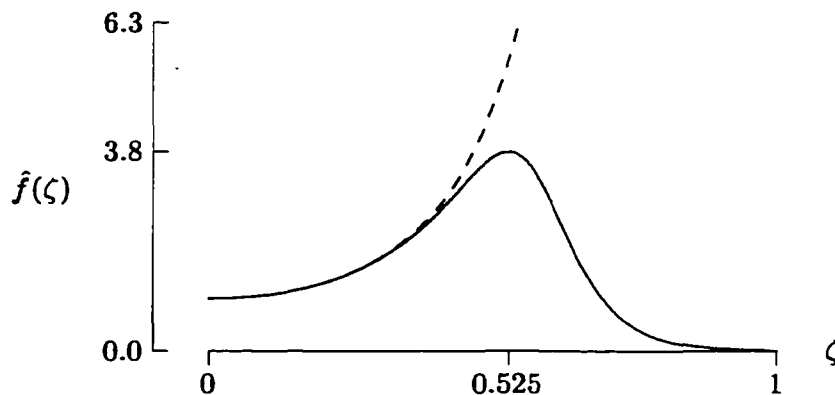


Figure 3.12: Noise Suppression through High-Boost Filter

At low frequencies, where the signal energy is more prominent than the noise energy, the

signal is restored by means of an inverse filter. At higher frequencies, however, where random noise can be more prominent than signal, the energy is suppressed. Other research[4, 7, 15] has been performed to investigate a means of effectively choosing a value for α . That investigation is not explored in this research.

3.3 A Priori Knowledge of CLS Filter Components

In this chapter we have derived the CLS Restoration filter and shown how it can be more effective than the inverse restoration filter. The equation for the CLS Restoration filter, however, relies upon knowledge of components inherent in acquisition and display. That is, in order to compute the CLS Restoration filter \hat{f} one must know the Optical Transfer Function (OTF), or \hat{H} , of the acquisition device and the Reconstruction Transfer Function (RTF), or \hat{D} , of the display device. In the next chapter we discuss a possible means by which the Optical Transfer Function of a particular acquisition device may be estimated.

Chapter 4

Pre-Sample Filter Estimation

As defined in Chapter 3, the CLS restoration filter requires a priori knowledge of the frequency response of the pre-sample acquisition filter \hat{H} and the reconstruction filter \hat{D} . In a simulation-based environment, these filters would be specified as part of the simulation model and used to process the image and to construct the CLS restoration filter. The end-to-end fidelity metric defined in Section 2.5 could then be used to determine the effectiveness of the restoration filter. To show the effectiveness of the CLS filter using actual, acquired digital data, however, requires an estimation of the \hat{H} and \hat{D} associated with an actual acquisition and reconstruction device. For the purposes of this dissertation, we will focus on the estimation of \hat{H} . Note that the estimation of \hat{H} need only be performed once per acquisition device configuration.

The Optical Transfer Function (OTF) $\hat{H}(\omega_1, \omega_2)$ of a digital acquisition device, such as a staring-array system, is the Fourier transform of the cascaded convolution of the point spread functions (PSF) of the optics, detectors, and any other physical features of the acquisition process, such as turbulence[9, 67], that blur the input scene prior to sampling.

Note that convolution is a shift invariant operation and this shift invariance is possible only if all the staring-array detectors have the same point spread function. Shift invariance may not necessarily be true in practice[33, 35, 37, 64], but, without this assumption, the OTF is not well-defined and so cannot be estimated unambiguously. The remainder of this chapter is a comprehensive investigation of OTF estimation for a particular acquisition device. In particular, use of the subpixel technique described in [43, 51] along with discussion about tradeoffs involved in estimating the OTF will be presented.

4.1 Filter Estimation Technique

The first step in estimating the OTF of an acquisition device is to acquire an image of a “knife edge” slightly inclined relative to the sampling grid such that the incline spans less than five pixel widths. The slight incline of the knife edge will produce many sampled edge profiles; ideally in each edge profile the edge location will occur at a different subpixel location relative to the sampling grid. A knife edge image, along with a typical horizontal (row) edge profile, are shown in Figure 4.1.

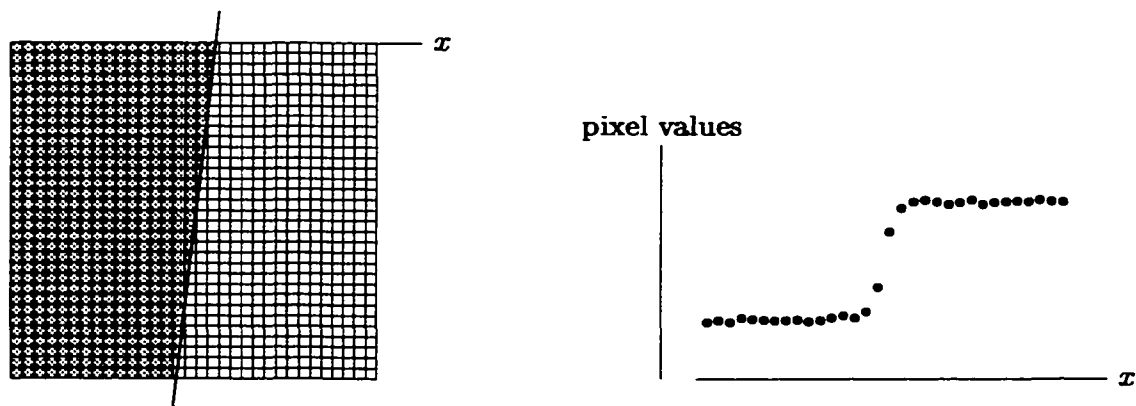


Figure 4.1: Knife Edge and Edge Profile

Once the knife edge image is acquired, an edge location algorithm is used to determine the subpixel edge location for each edge profile. That is, given a sampled edge profile as input, the algorithm calculates a subpixel edge location as output. As will be discussed in Chapter 5, there are a variety of edge location algorithms, some of which use the entire sampled edge profile, while others use a subset of the values. Regardless of the details of the algorithm used, the input is a sampled edge profile $p[n]$, $n = 0, 1, 2, \dots, N - 1$ and the output is a real-valued edge location estimate $\hat{\gamma}$, $0 < \hat{\gamma} < P$ ($P = N\xi$), as illustrated in Figure 4.2.

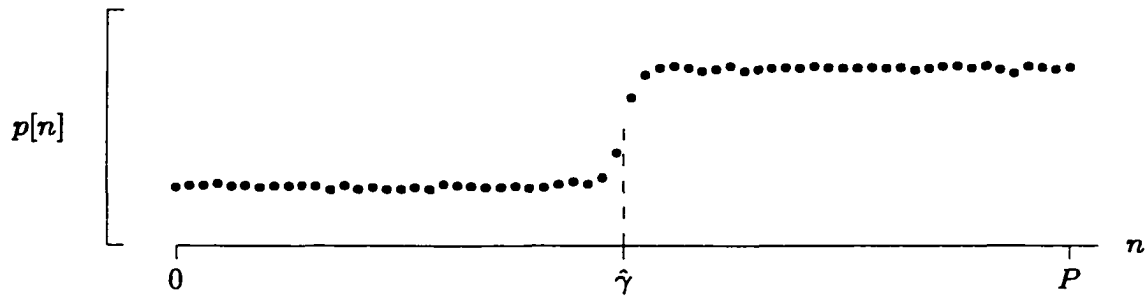


Figure 4.2: Edge Location Estimation

Once an edge location has been determined for each sampled edge profile, the profiles are aligned based on their edge locations. That is, as illustrated in Figure 4.3, a common reference point is chosen and a *composite* edge profile is constructed by positioning each edge profile with its $\hat{\gamma}$ estimate on the common reference point. For the example following, the common reference point was chosen to be $P/2$. Although there is no fundamental reason for this choice, positioning the common reference point at $P/2$ allows for the maximum possible span of edge profile values which fall in between the high (bright) intensity values and low (dark) intensity values.

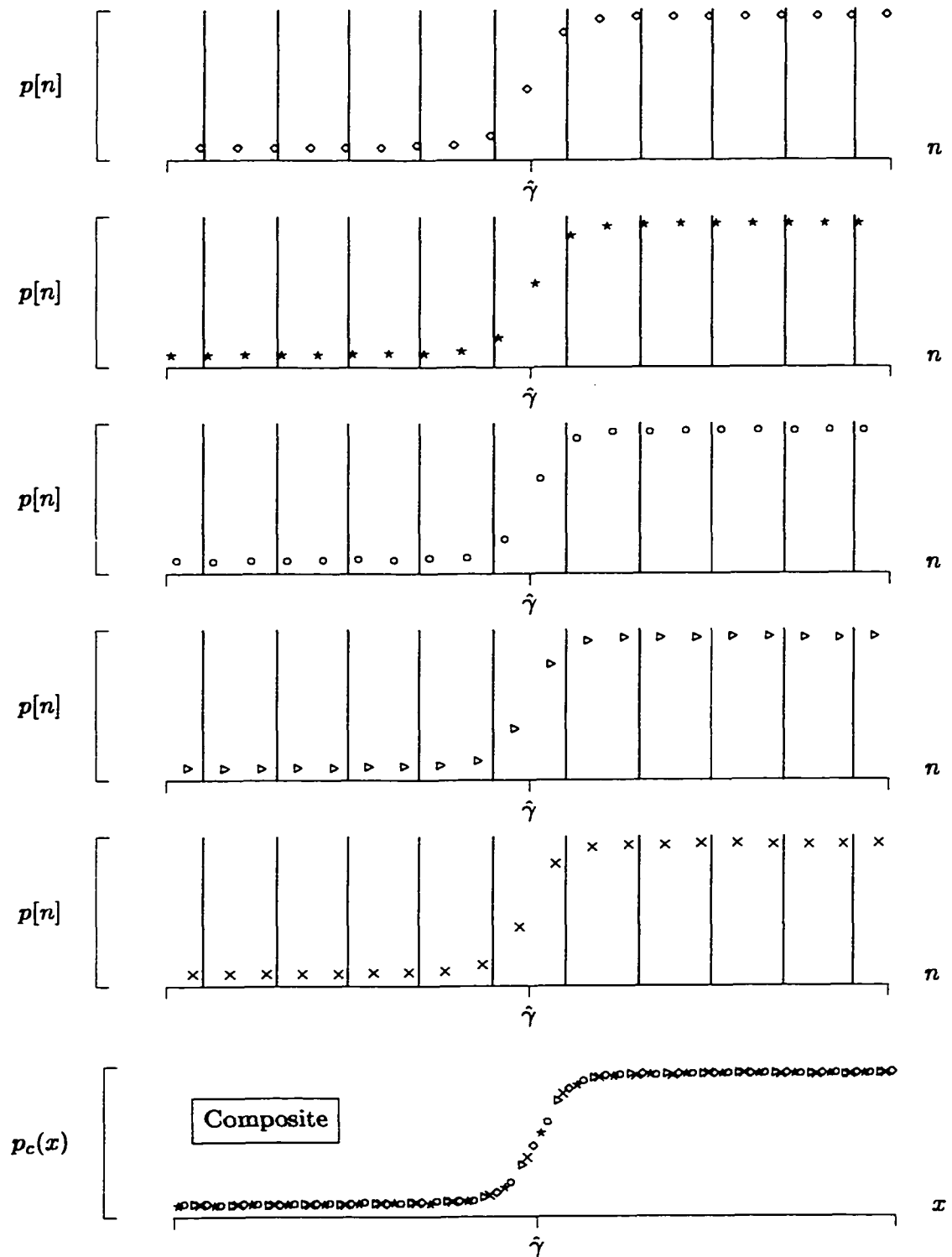


Figure 4.3: Binning Edge Profiles to form Composite Edge Profile

After the composite edge profile is created through alignment of the edge profiles based on their estimated edge locations, the sampled data from the composite edge profile is binned based on an integer-valued bins per pixel parameter ρ . For the example shown in Figure 4.4 the four center pixel widths from 50 sampled edge profiles have been aligned and binned based on $\rho = 4$ (or 4 bins per pixel width ξ).

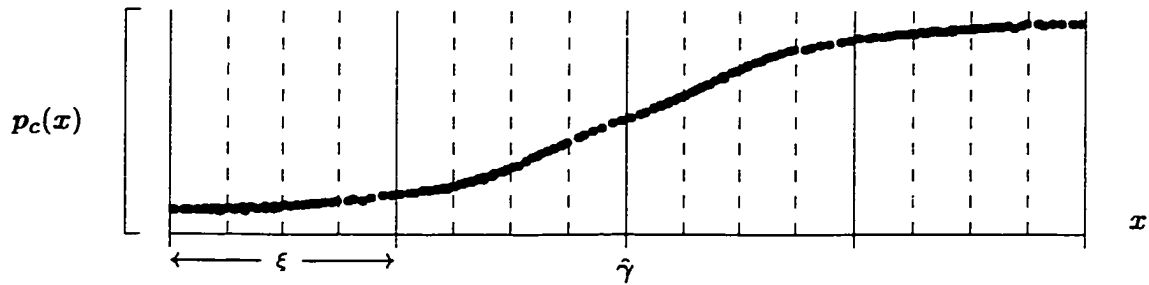


Figure 4.4: Binned data in 16 bins across 4 pixels

The sampled data in each bin is then averaged to produce a “super-resolution” edge profile, as illustrated in Figure 4.5. Although this super-resolution edge profile is not noise-free, the process of averaging the data from the initial edge profiles reduces the effects of the noise present.

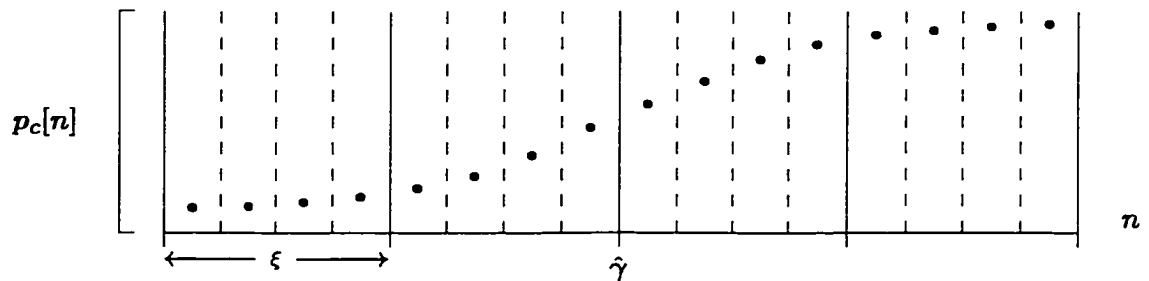


Figure 4.5: Averaged bin values in 16 bins across 4 pixels

The above graph represents the averaged data for the 4 pixels (16 bins) that are closest to the center of the “super-resolution” edge profile. The entire super-resolution edge profile

$p_c[n]$ with $n = 0, 1, 2, \dots, \rho N - 1$ (shown in Figure 4.6) represents the estimated line spread function (LSF) of the acquisition device. This estimate represents a sampled array with a higher sampling density (provided $\rho > 1$) than the individual edge profiles.

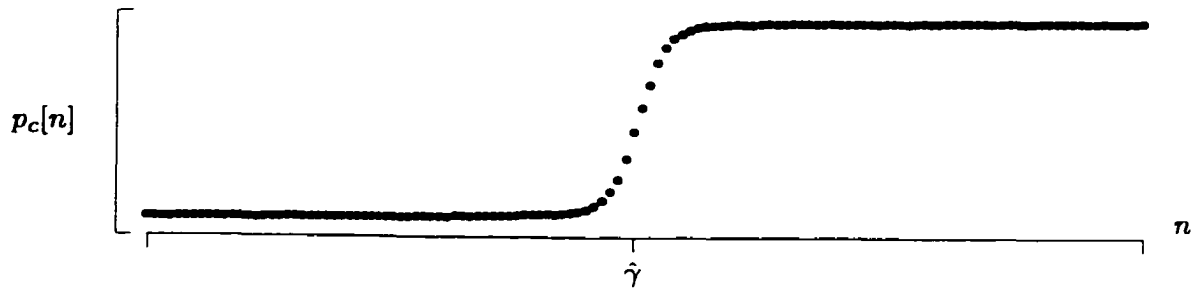


Figure 4.6: Super-Resolution Edge Profile

4.2 Estimating \hat{H} From p_c

In the previous section we defined the means by which a “super resolution” edge p_c is computed. In this section we will use this sampled edge profile to estimate the OTF \hat{H} .

4.2.1 Scene Model

Using the c/d/c system model described in Chapter 2, an input scene model is defined. An ideal unit-step edge function u is defined as

$$u(x) = \begin{cases} 1, & \text{if } x \geq 0 \\ 0, & \text{if } x < 0. \end{cases}$$

The *three-parameter* ideal edge-function

$$s(x) = \delta + (\beta - \delta)u(x - \gamma) = \begin{cases} \beta, & \text{if } x \geq \gamma \\ \delta, & \text{if } x < \gamma \end{cases}$$

defines the input scene. The parameters δ , β , γ represent the *dark* background value, the *bright* foreground value, and the *edge location* respectively, as indicated in Figure 4.7.

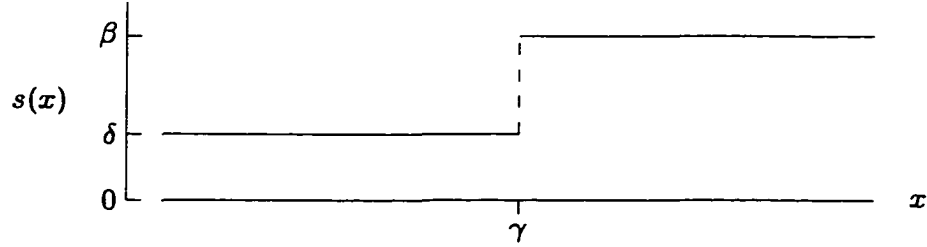


Figure 4.7: Ideal edge function, input scene

The input scene s can be defined in terms of its Fourier coefficients

$$\begin{aligned} \hat{S}[\nu] &= \frac{1}{2\pi\nu} ((\delta - \beta) \sin(2\pi\nu\gamma/P) + \beta \sin(2\pi\nu)) \\ &+ \frac{i}{2\pi\nu} ((\delta - \beta) \cos(2\pi\nu\gamma/P) + \beta \cos(2\pi\nu) - \delta) \end{aligned} \quad (4.1)$$

where

$$\hat{S}[\nu] = \frac{1}{P} \int_0^P s(x) \exp(-i2\pi x\nu/P) dx \quad (4.2)$$

for $\nu = 0, \pm 1, \pm 2, \dots$

From the 1-dimensional equivalent of Equation 2.11

$$\hat{p}[\nu] = \hat{G}[\nu] + \hat{a}[\nu] + \hat{e}[\nu].$$

If the additive random noise effects are ignored (for now), then

$$\hat{p}[\nu] = \hat{S}[\nu] \hat{H}(\nu/P) + \hat{a}[\nu].$$

Thus,

$$\hat{H}(\nu/P) = \frac{\hat{p}[\nu] - \hat{a}[\nu]}{\hat{S}[\nu]} \quad (4.3)$$

provided

$$\hat{S}[\nu] \neq 0.$$

$\hat{S}[\nu]$ is known from Equation 4.1. Substituting $\hat{p}_c[\nu]$ for $\hat{p}[\nu]$, we estimate \hat{H} to be

$$\hat{H}(\nu/P) = \frac{\hat{p}_c[\nu]}{\hat{S}[\nu]} \quad (4.4)$$

and, thus, only the effects of aliasing will degrade the estimate. Unfortunately, the effects of aliasing tend to degrade the estimation significantly. An alternative to this estimation is to calculate the Fourier Transform of the super-resolution PSF (calculated from the super-resolution LSF as $q[n] = p[n] - p[n - 1]$). For the results presented here and in subsequent chapters, the estimation of the OTF is calculated as the Fourier Transform of the super-resolution PSF.

4.2.2 \hat{H} Shape and Subsampling

For all simulation results presented in this chapter we chose to use the following OTF Model

$$\hat{H}(\omega) = \exp(-(\omega/\alpha)^2) \quad (4.5)$$

where α is a real-valued, free parameter that defines the effective width of the PSF h . For example, Figure 4.8 depicts the relationship between α and ξ (the distance between adjacent detectors, or ‘intersample’ distance).

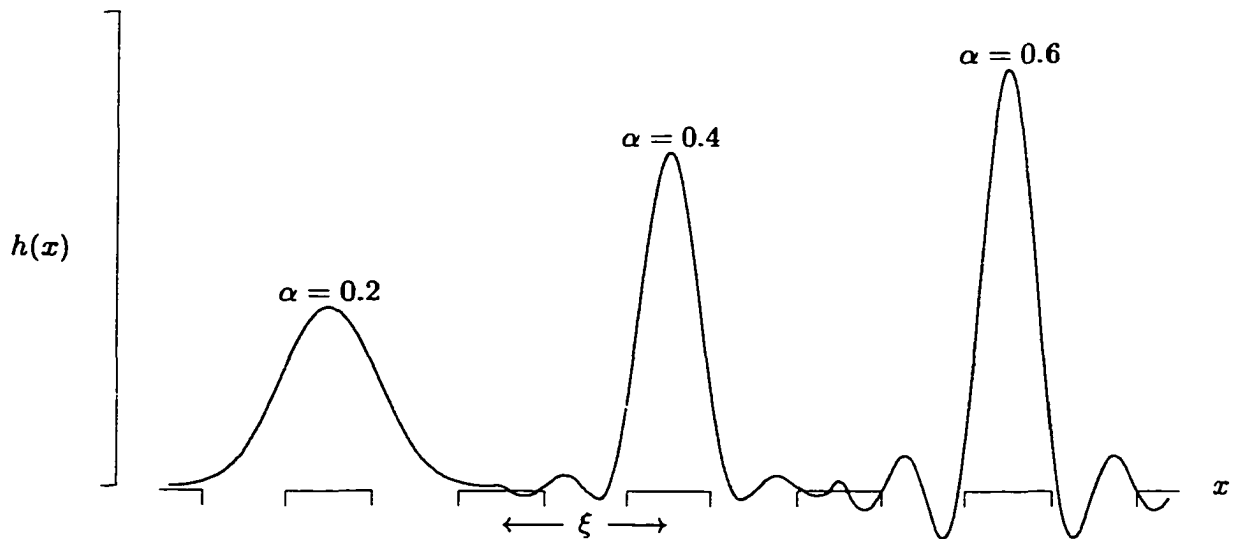
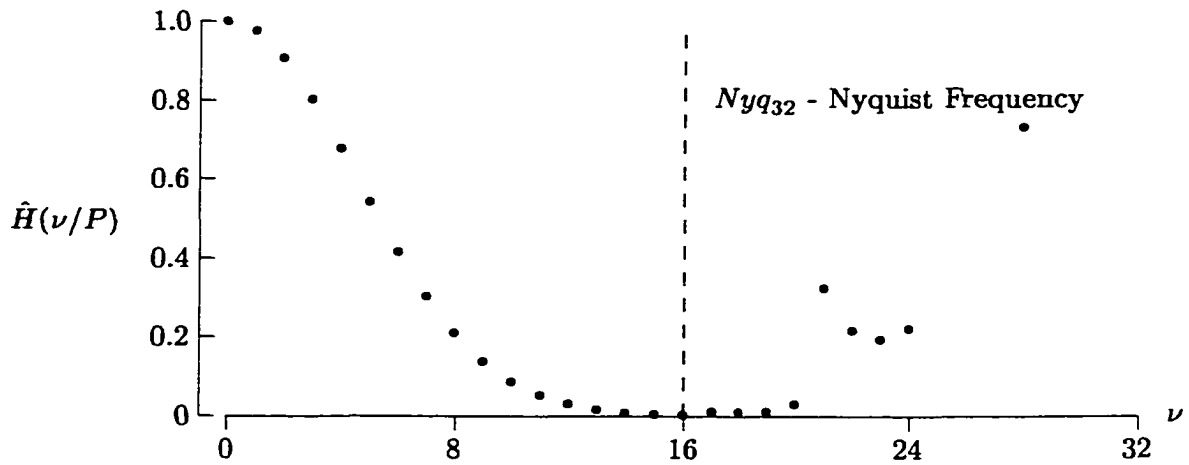


Figure 4.8: Various h graphs relative to α

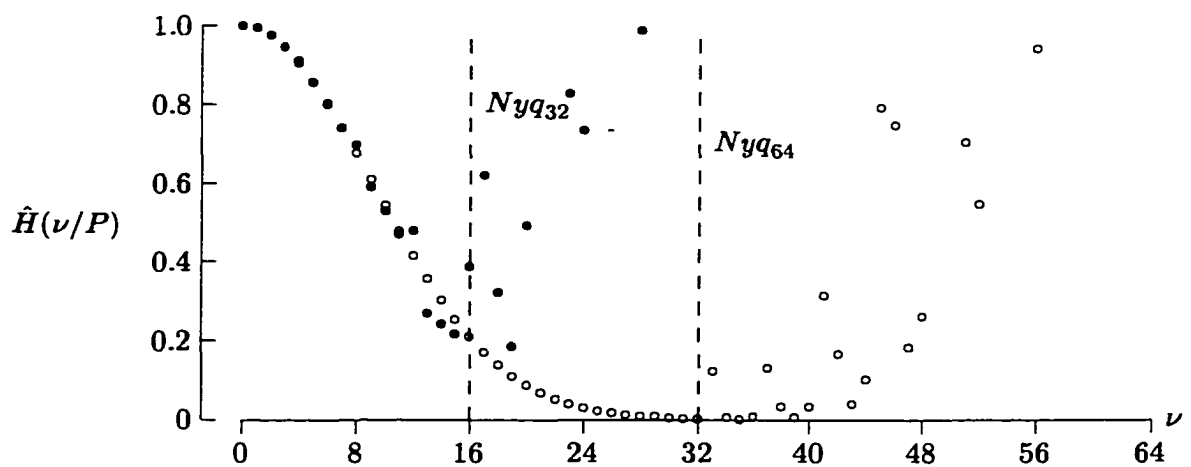
4.2.3 \hat{H} Estimate ($\alpha = 0.2$)

Figure 4.9 shows the estimate of \hat{H} with $\alpha = 0.2$ and a sampling density of $N = 32$. That is, the input scene \hat{S} was filtered by the acquisition filter \hat{H} ($\alpha = 0.2$) and sampled 32 times within the period P . The resulting sampled array \hat{p}_c was then divided by the input scene to produce the OTF estimate. At higher frequencies ($\nu > 20$) the aliased noise becomes more prominent than the signal energy and, thus, the estimate breaks down.

Figure 4.9: \hat{H} Estimate, $\alpha = 0.2$

4.2.4 \hat{H} Estimate ($\alpha = 0.4$)

Figure 4.10 shows the estimate of \hat{H} with $\alpha = 0.4$ and sampling densities of $N = 32$ ('•') and $N = 64$ ('◦'). As α increases (or equivalently the width of h decreases) the effects of the aliased noise degrade the estimate at lower frequencies. Thus, by subsampling ($N = 64$) aliasing effects can be suppressed out to higher frequencies.

Figure 4.10: \hat{H} Estimate, $\alpha = 0.4$

4.2.5 \hat{H} Estimate ($\alpha = 0.6$)

Figure 4.11 shows the estimate of \hat{H} with $\alpha = 0.6$ and sampling densities of $N = 32$ ('•'), $N = 64$ ('◦') and $N = 96$ ('△'). Once again, increasing α causes the aliased noise to degrade the estimate at lower frequencies. Increasing the sampling density suppresses the aliased noise out to higher frequencies.

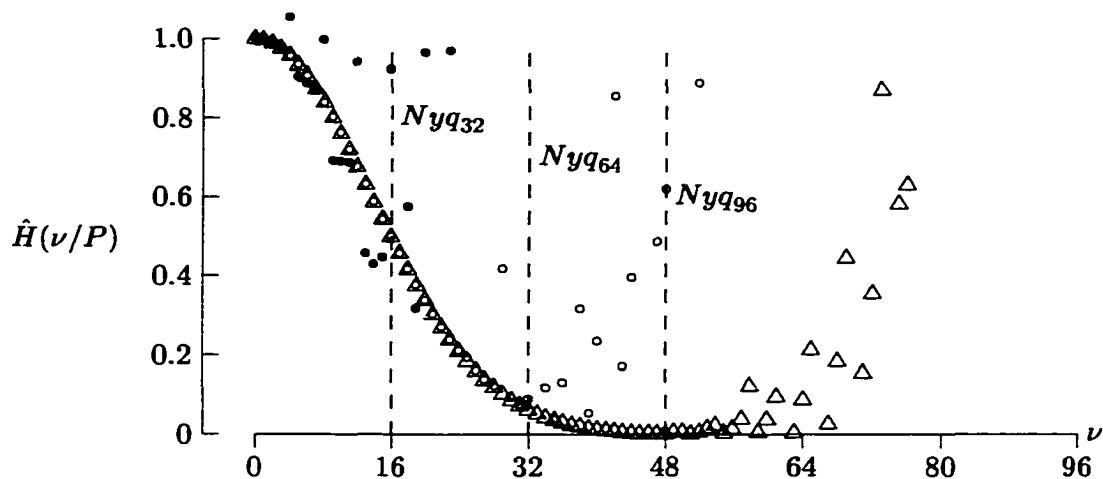


Figure 4.11: \hat{H} Estimate, $\alpha = 0.6$

4.3 Bins Per Pixel (Noise vs. Nyquist Frequency)

An important tradeoff exists when estimating the OTF of an acquisition device using the technique described in the previous section. Given the possibility of noise degradation within the knife edge image, it is prudent to use many sampled data values per bin when computing the super resolution edge profile. However, for a fixed number of sampled edge profiles from the acquired knife edge image, averaging more data per bin would require decreasing ρ .

In Section 4.1, $\rho = 4$ was used. Reducing ρ to 2 and using the same composite data, as shown in Figure 4.12,

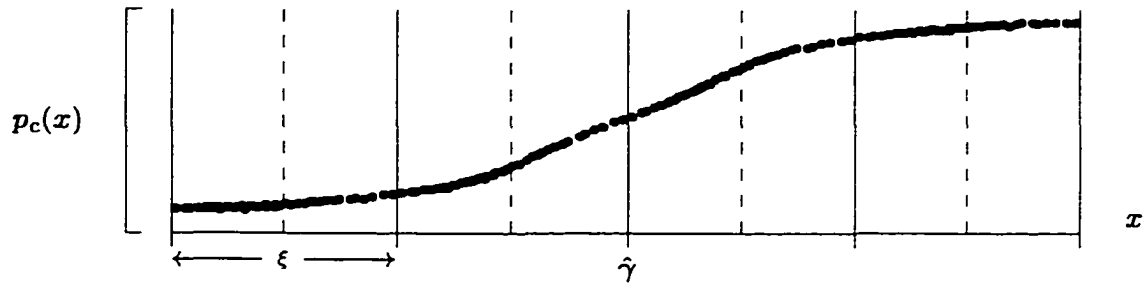


Figure 4.12: Binned data in 8 bins across 4 pixels

produces the estimate of the LSF in Figure 4.13 at its center-most 4 pixels.

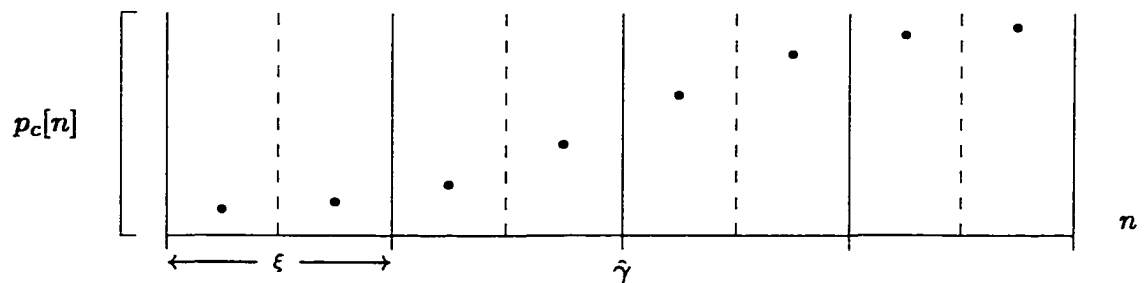


Figure 4.13: Averaged bin values in 8 bins across 4 pixels

Although averaging more sampled data per bin will reduce the effects of noisy data on the estimated LSF, reducing ρ decreases the number of estimated LSF data values. Accordingly, as shown in the previous section, the extent to which the OTF can be estimated beyond the Nyquist frequency decreases. Thus, an optimal value of ρ must be determined such that noise is suppressed yet a maximal OTF estimate is achieved.

4.4 Data Density Along Frequency (ω) Axis

Based on the discussion in the previous section suppose it is concluded that more sampled data from the super-resolution edge should be used to extend the estimate further beyond the Nyquist frequency while still keeping noise effects to a minimum. In support of this supposition we might conjecture that simply padding samples of low intensity data to the low intensity side of the super-resolution edge and padding samples of high intensity data to the high intensity side of the super-resolution edge will extend the estimation of the OTF to higher frequencies. That is, given the data from the super-resolution edge in the previous section, shown in Figure 4.14,

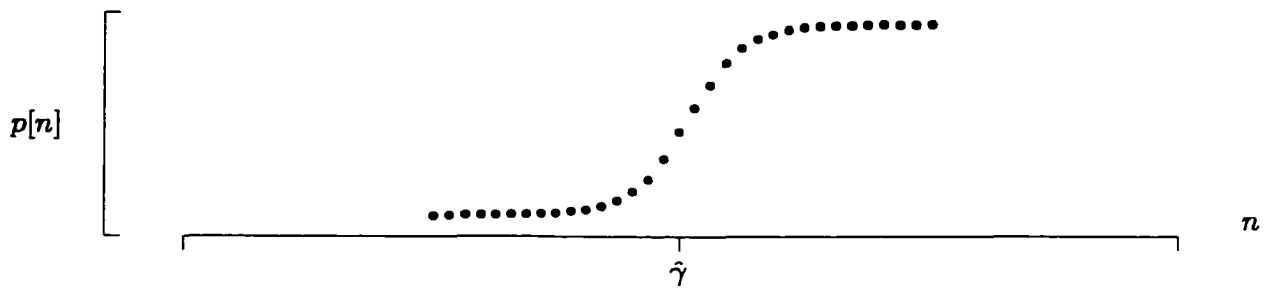


Figure 4.14: Super-Resolution Edge Profile

and padding low intensity and high intensity data, indicated by the \circ , as illustrated in Figure 4.15, we might conjecture that the estimate of the OTF will extend to higher frequencies than an estimate without the padded data.

This conjecture is false. Instead, simply including padded sampled data to the super-resolution edge produces an OTF estimate which contains more data points within the same frequency range. That is, this technique will simply refine the OTF estimation by having a higher density of values along the frequency axis. In the example shown, data has been

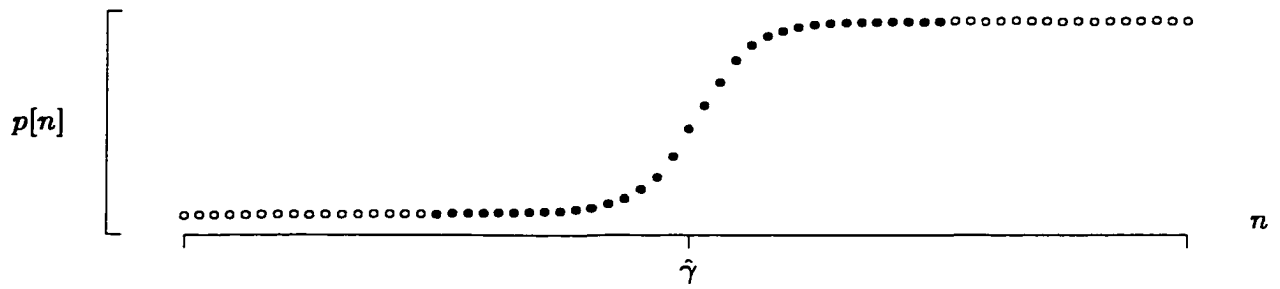


Figure 4.15: Super-Resolution Edge Profile with Padded Data

padding so that the number of samples has doubled. The effect of doubling the number of samples can be seen in Figure 4.16 where the \bullet 's represent the original data points and the \circ 's represent the additional data values produced by extending the super-resolution edge through padding of low intensity and high intensity values.

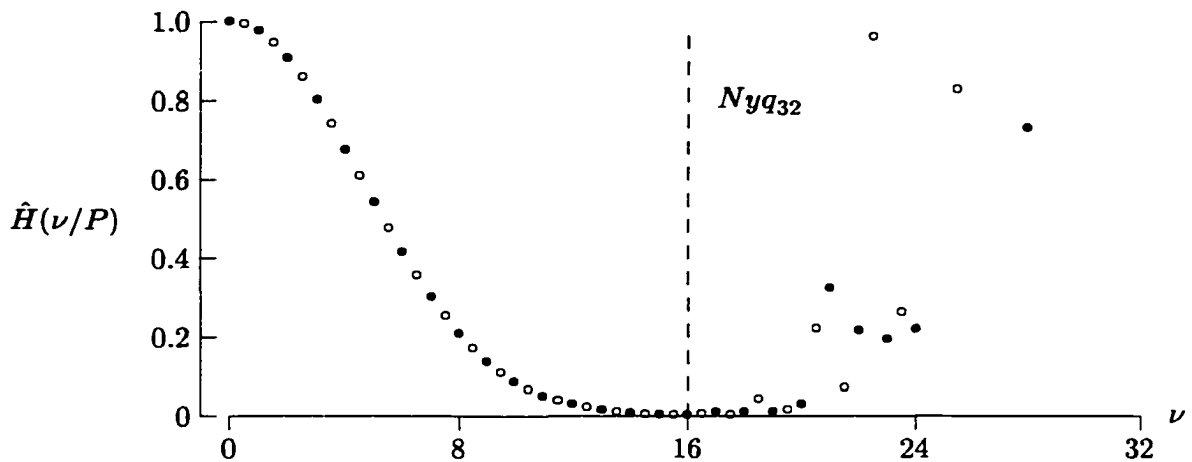


Figure 4.16: \hat{H} Estimate with Padded Data

The algorithm used to produce the OTFs illustrated in this chapter is shown in Section 4.5. Given an acquired image containing a knife edge, the algorithm assumes that a rectangular subset of the acquired sampled data is used. Thus the parameters *left* and *right* refer to the indices of the left and right edges of the rectangular subset with respect to the acquired

image as a whole. For estimating the OTF in the x_1 direction, these parameters would be replaced by **top** and **bottom**.

4.5 Algorithm

```

void main()
{
    long rowNum, colNum;
    long **input_pic; // input scene containing edge profiles
    long edge_profile[right-left]; // individual edge profile
    long up_ind; // index of sample just to the right of
                // the estimated edge location
    // summation of data values in each bin
    long binned_data[NumBinsPerPixel*(right-left)];
    // number of data values in each bin
    long binned_count[NumBinsPerPixel*(right-left)];

    for (rowNum = 0; rowNum < NumRows; rowNum++)
    {
        // Get sampled edge profile
        for (colNum = left; colNum < right; colNum++)
        {
            edge_profile[colNum-left] = input_pic[rowNum][colNum];
        }

        // Estimate the edge location
        gamma_hat = EstimateEdgeLocation( edge_profile );

        // Bin the line based on the edge location
        BinIndex = CommonReferencePoint - (long)(NumBinsPerPixel * gamma_hat);
        for (i = 0; i < right-left; i++)
        {
            binned_data[BinIndex] += edge_profile[i];
            binned_count[BinIndex]++;
            BinIndex += NumBinsPerPixel;
        }
    }
    AverageBins(); // binned_data[i]/binned_count[i]
                  // Computes LSF (super-resolution edge profile)
    ComputePSF();
    FourierTransformPSF(); // Computes OTF
}

```

4.6 Alternative Approach to OTF Estimation

There are alternatives to the OTF estimation algorithm presented in this chapter[19, 34, 49, 54, 60]. One alternative to the algorithm described in Sections 4.2-4.5 is presented by Tzannes and Mooney[62]. Tzannes and Mooney also start with an acquired image of a knife-edge. An estimate of the edge location for each sampled edge profile is computed. The edge profiles are aligned to the common reference point based on their estimated edge location. At this point, however, the algorithms differ. Tzannes and Mooney fit a summation of three Fermi functions to the composite edge profile.

$$F(x) = d + \sum_{i=0}^2 \frac{a_i}{\exp[(x - b_i)/c_i] + 1}$$

This function, $F(x)$, is used as the estimate of the line spread function of the acquisition device. The 10 free parameters are determined by a non-linear least squares fit to the sampled composite edge profile. The function $F(x)$ is then differentiated to produce a point spread function. The point spread function is sampled and the sampled data is processed through a direct fourier transform and scaled to produce the OTF estimate.

4.7 Fitting a Model Function to the Estimate

Regardless of the technique used to estimate an OTF, the resulting estimate will be discrete valued. To be used in a restoration filter, it is conventional to fit the estimated OTF to a continuous function. Figure 4.17 shows the acquired image of a square that has been

rotated slightly. Using 50 sample rows and focusing on the left edge of the square, the technique described in Section 4.1 is performed.

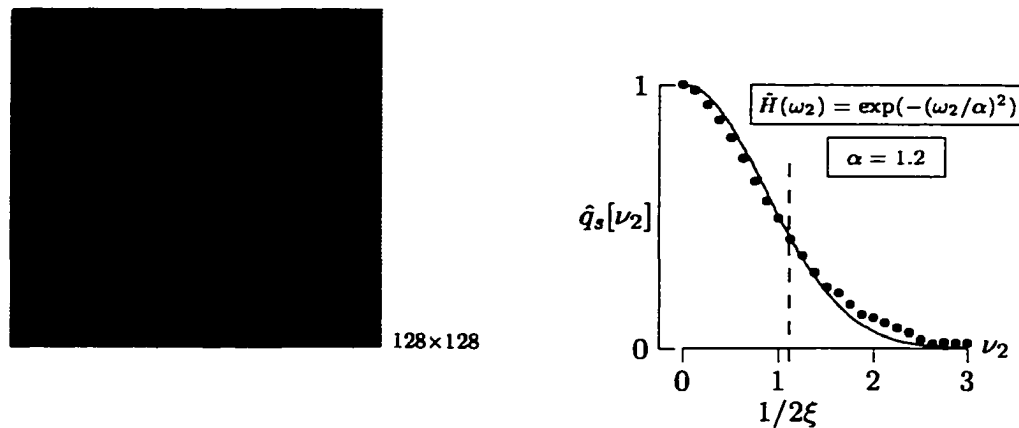


Figure 4.17: Acquired Square Image and Estimated \hat{H} in x_2 direction

The estimated OTF of the device which acquired the image of the square is shown by the graph on the right (indicated by the •'s). A model function is then fit to the estimated OTF. The solid line in Figure 4.17 represents the model used for this acquisition device. The gaussian function, with one free parameter $\alpha = 1.2$, closely represents the estimated OTF data.¹ This example represents an estimation of the OTF of the acquisition device in the x_2 direction. To produce a complete representation of the OTF, an estimation in the x_1 direction should also be performed using the top or bottom edge of the square. The model functions for each direction could then be combined to produce a complete 2-dimensional OTF estimate, $\hat{H}(\omega_1, \omega_2)$.

¹Although a gaussian function is used here, other model functions, such as sinc, are valid.

4.8 \hat{H} Is 2-dimensional

In Section 4.1, we described a technique by which one could determine a 1-dimensional estimate of the OTF of a particular acquisition device. The OTF, however, is a 2-dimensional function having both x_1 and x_2 components[66]. If we assume that the OTF is *circularly symmetric*, then estimating the OTF in one direction is sufficient for defining the model function. That is, given the model function estimating the OTF in one direction

$$\hat{H}(\omega) = \exp(-(\omega/\alpha)^2),$$

the circular version of the model function can be obtained.

$$\hat{H}(\omega_1, \omega_2) = \exp(-(\sqrt{\omega_1^2 + \omega_2^2}/\alpha)^2)$$

A necessary, but not sufficient, condition for the assumption of circular symmetry is for the estimate of the OTF in the x_1 direction to be equal to the estimate in the x_2 direction. Focusing on the upper edge of the square in the acquired image shown in the previous section, the same estimation technique is performed to generate an estimate of the OTF in the x_1 direction, as illustrated in Figure 4.18.

In this particular case it appears as though the estimates in both directions are the same. These two estimates may not always be equal, however, particularly if the detectors are not perfectly square. If the detectors are rectangular, yet not square, then the frequency response in one direction will differ from the frequency response in the orthogonal direction. In the case of differing responses with respect to orthogonal directions, it is usually assumed that the OTF is at least *separable*. That is, given that the estimate of the OTF in the x_1

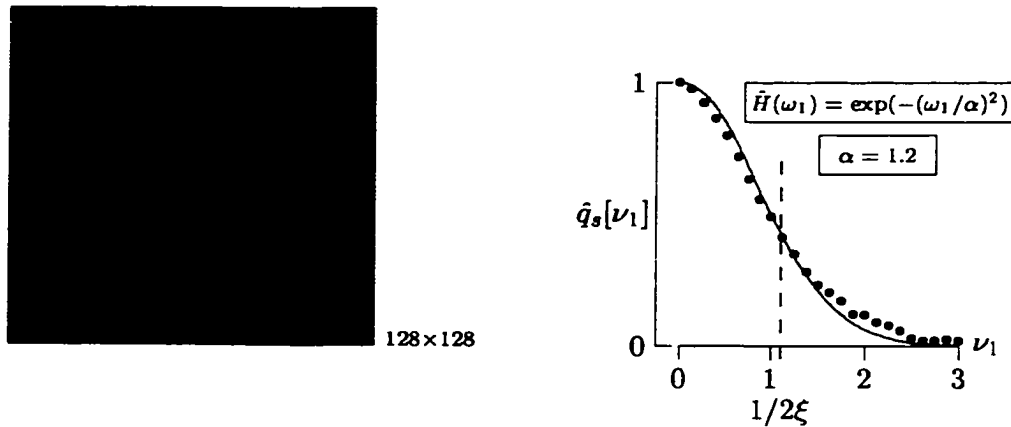


Figure 4.18: Acquired Square Image and Estimated \hat{H} in x_1 direction

direction is

$$\hat{H}(\omega_1) = \exp(-(\omega_1/\alpha_1)^2)$$

and that the estimate of the OTF in the x_2 direction is

$$\hat{H}(\omega_2) = \exp(-(\omega_2/\alpha_2)^2)$$

one can assume that the 2-Dimensional estimate of the OTF is

$$\hat{H}(\omega_1, \omega_2) = \exp(-(\omega_1/\alpha_1)^2) \exp(-(\omega_2/\alpha_2)^2) \quad (4.6)$$

Once again, a necessary, but not sufficient, condition for the OTF to be separable is that the estimated OTF along the 45 degree axis agrees with the separable equation (Equation 4.6). In order to test this theory, a knife edge must be acquired which is slightly inclined from a 45 degree angle. Using this type of input scene and sampling along a diagonal produces edge profiles. It is important to note that if a comparison is to be done between the x_1 or

x_2 directions and the 45 degree sampled edge profiles, it must be taken into consideration that the intersample distance has changed from ξ_1 or ξ_2 to $\sqrt{\xi_1^2 + \xi_2^2}$.

Using the technique described in this chapter on an image of a square that has been rotated 45 degrees, the OTF is estimated in the diagonal direction. Edge profiles for this estimation were formed by using samples along the orthogonal 45 degree angle to the knife edge. After scaling by a factor of $\sqrt{\xi_1^2 + \xi_2^2}$ due to the increased intersample distance, it can be seen in Figure 4.19 that the estimated 1-Dimensional OTF in the diagonal direction has a strong correlation to the estimates in the x_1 and x_2 directions.

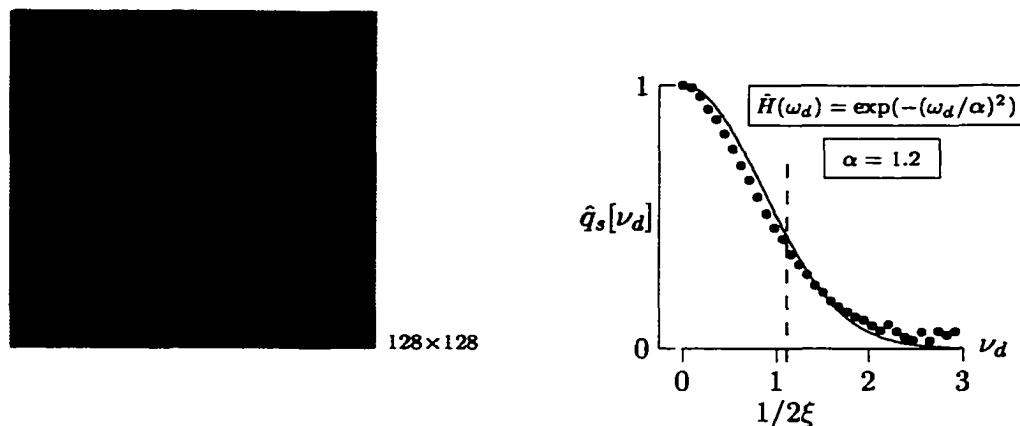


Figure 4.19: Acquired Square Image and Estimated \hat{H} in diagonal direction

From the results presented, it can be strongly argued that the acquisition device represented here has a circularly symmetric OTF.

4.9 Importance of Edge Location Estimation

In this chapter we presented a step-by-step process by which the Optical Transfer Function of a digital acquisition device can be estimated. The ability to accurately estimate an OTF relies heavily on the estimation of the edge location from sampled edge profile arrays. We

presented this estimation as a generic equation. That is, from an input of sampled edge profile data an edge location estimate $\hat{\gamma}$ is calculated. In the next chapter we discuss various algorithms which can be used to calculate the edge location estimate.

Chapter 5

Edge Location Algorithms

A critical step in estimating the optical transfer function of a scene sampling device is the computation of the edge location for each representative edge profile taken from the “knife edge” input scene. Park and Idema[43] describe three possible algorithms to estimate the edge location for a given (sampled) edge profile. In this chapter we expand on their findings and discuss various additional algorithms which can be used to estimate the edge location of a sampled edge profile. Although “edge detection” represents a different research interest than “edge location,” there is some overlapping theory. Various studies have been performed using edge profile data [10, 26, 40, 48, 68] and occasionally localization is investigated in conjunction with edge detection[5, 6, 31, 61].

The algorithms presented here are grouped into four categories depending on how the data from the edge profile are used. “Global” algorithms use all the data from the edge profile to estimate the edge location. “Local” algorithms use a subset of the edge profile to estimate the edge location. Within each of these two categories we have defined two types of algorithms. Algorithms which use the edge profile directly are labeled Edge Spread

Function (ESF) algorithms. Algorithms which use the discrete differentiation of the edge profile are labeled Point Spread Function (PSF) algorithms.

5.1 System Model

5.1.1 Scene Model

The ideal edge profile function defined in Section 4.2.1 is used as the scene model for all of the result presented in this chapter. As in Section 4.2.1 the *three-parameter* ideal edge-function is defined as

$$s(x) = \delta + (\beta - \delta)u(x - \gamma) = \begin{cases} \beta, & \text{if } x \geq \gamma \\ \delta, & \text{if } x < \gamma \end{cases}$$

and is depicted in Figure 5.1.

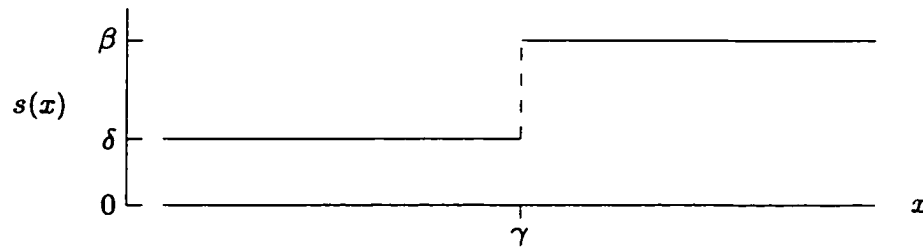


Figure 5.1: Ideal Edge Profile

The algorithms presented in this chapter attempt to estimate the edge location γ given a noisy, blurred, sampled version of $s(x)$. In some of the algorithms presented, an estimation of γ requires an estimation of δ and β .

5.1.2 PSF/ESF Model

If the ideal unit-step edge function u is used as the input scene to the c/d/c system model, the resulting image function $H = u \otimes h$ defines the ESF of the imaging system[43]. This ESF is defined by

$$H(x) = \int_{-\infty}^{\infty} u(x - x')h(x') dx' = \int_{-\infty}^x h(x') dx' \quad -\infty < x < \infty.$$

Because the PSF h is non-negative, the ESF H is monotone increasing. Additionally, the PSF is the derivative of the ESF or

$$\frac{dH(x)}{dx} = h(x) \quad -\infty < x < \infty.$$

It is common to assume that the PSF is normalized so that

$$\int_{-\infty}^{\infty} h(x) dx = 1 \tag{5.1}$$

and symmetric about $x = 0$ so that

$$\int_{-\infty}^{\infty} xh(x) dx = 0 \quad \text{and} \quad H(0) = \int_{-\infty}^0 h(x) dx = 1/2. \tag{5.2}$$

Consistent with the c/d/c system model, given the three-parameter ideal edge function model

$$s(x) = \delta + (\beta - \delta)u(x - \gamma),$$

the corresponding blurred (pre-sampled) image function $g = s \otimes h$ is

$$g(x) = \int_{-\infty}^{\infty} s(x - x')h(x') dx'$$

$$\begin{aligned}
&= \delta \int_{-\infty}^{\infty} h(x') dx' + (\beta - \delta) \int_{-\infty}^{\infty} u(x - \gamma - x') h(x') dx' \\
&= \delta + (\beta - \delta) H(x - \gamma) \quad -\infty < x < \infty.
\end{aligned}$$

For the purposes of results presented in this chapter, we chose to define the pre-sample PSF/ESF component of the c/d/c system model as the one-parameter Gaussian PSF

$$h(x) = \frac{1}{\alpha} \exp(-\pi x^2/\alpha^2) \quad -\infty < x < \infty.$$

The corresponding ESF model is

$$H(x) = \Phi(\sqrt{2\pi}x/\alpha) \quad -\infty < x < \infty$$

where $\Phi(\cdot)$ is the Gaussian cumulative distribution function defined by

$$\Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^z \exp(-t^2/2) dt \quad -\infty < z < \infty.$$

As shown in Figure 5.2, the real-valued parameter $\alpha > 0.0$ is related to the width of the PSF or, equivalently, the spread of the ESF.

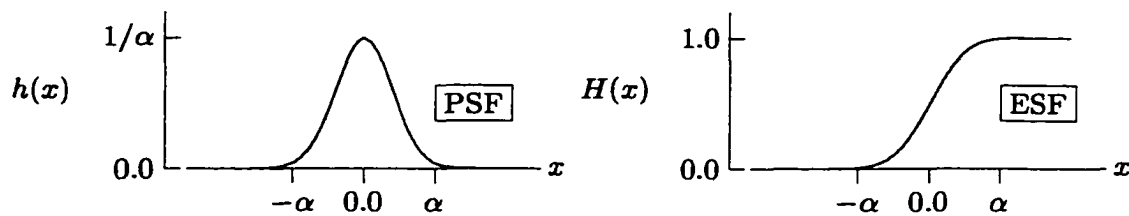


Figure 5.2: PSF and ESF graphs

5.1.3 Noise Model

As with [43], the real-valued image function $g = s \otimes h$,

$$g(n) = \delta + (\beta - \delta)H(n - \gamma)$$

is sampled at $n = 0, \pm 1, \pm 2, \dots$. Due to random noise error, including quantization error, present in the acquisition process the resulting integer-valued image sequence p is defined by the equation[43]

$$p[n] = \lfloor g(n) + \phi[n] + 0.5 \rfloor \quad n = 0, \pm 1, \pm 2, \dots$$

where the noise sequence ϕ represents real-valued measurement error prior to quantization. The $\lfloor (\cdot) + 0.5 \rfloor$ function models the real-to-integer conversion by rounding.

Independent realizations of a *Normal*(0, σ) random variable with mean zero and standard deviation $\sigma > 0.0$ defines the measurement error model. Given the pre-quantization noise sequence ϕ , the noise sequence e defined by

$$e[n] = p[n] - g(n) \quad n = 0, \pm 1, \pm 2, \dots$$

is used to characterize all simulation results[43]. Specifically, the *signal-to-noise ratio* (SNR) is defined as

$$\text{SNR} = \frac{\beta - \delta}{2\sigma_e}$$

where σ_e represents the standard deviation of the noise sequence e .

5.1.4 System Simulation

A simulation based on the model presented in the previous sections requires the specification of the real-valued parameters δ , β , α , and σ . For simulation purposes, the image sequence p and noise sequence e are finite – not infinite as defined in the previous section. Therefore, both sequences are assumed to be N -point sequences where N is an even integer. Thus, the pixel range for each of these sequences is $n = 0, 1, 2, \dots, N - 1$. The remaining free parameter γ represents the edge location and will be estimated by the algorithms presented in this chapter.

Although the parameter γ is unknown, an intelligent selection of the sampled region (by visual inspection of the input scene, if necessary) can position γ close to the center of the sequence. That is, γ will be close to $(N - 1)/2$. Given possible variations in SNR and α , we allow for a ± 1.5 pixel uncertainty in the edge location so that $(N - 4)/2 < \gamma < (N + 2)/2$ [43]. Consistent with the assumption of the edge location, a *Uniform*(0, 1) random number generator is used to generate values of γ . That is, γ is computed as $(N - 4)/2$ plus the sum of three independent *Uniform*(0, 1) random variates. This computation guarantees that the expected value of γ is $(N - 1)/2$ and the standard deviation of γ is 0.5.

The image sequence p and the associated noise sequence e are simulated as indicated.

$$\begin{aligned} &\gamma = (\text{Uniform}(0.0, 1.0) + \text{Uniform}(0.0, 1.0) + \text{Uniform}(0.0, 1.0) + (N - 4)/2); \\ &\text{for } (n = 0; n < N; n++) \\ &\{ \\ &\quad g = \delta + (\beta - \delta) \times H(n - \gamma); \\ &\quad \phi = \text{Normal}(0.0, \sigma); \\ &\quad p[n] = \lfloor g + \phi + 0.5 \rfloor; \\ &\quad e[n] = p[n] - g; \\ &\} \end{aligned}$$

The function *Uniform*(0, 1) represents a random number generator. The function *Normal*(0, σ)

represents a random variate generator.

5.1.5 Ensemble Simulation

In a simulation environment, all system parameters and components, including the edge location γ , are known. In practice, however, most parameters and components can only be estimated. Thus, in order to ensure that an edge location algorithm is robust and accurate, it must be presented with a variety of edge profile inputs which vary in *signal-to-noise ratio* and α . The accuracy of the algorithm can then be based on the average error incurred across all presented input edge profiles.

Consistent with the generic algorithm presented in the previous section, an *ensemble* of image sequences is generated. The edge location algorithm computes an estimate $\hat{\gamma}$ for each image sequence. The accuracy of the algorithm is based on the two statistical metrics,

$$E[\gamma - \hat{\gamma}] = \text{estimator bias}$$

and

$$\sqrt{\frac{E[(\gamma - \hat{\gamma})^2]}{E[(\gamma - \hat{\gamma})^2]}} = \text{estimator root-mean-square error (RMSE)}$$

where $E[\cdot]$ represents the expected value (average) over the ensemble of edge sequences[43].

When properly defined (see Section 5.2), all sub-pixel edge location estimation algorithms presented are unbiased. Therefore, all comparison results presented will be based solely on the estimator RMSE.

5.2 Half-Pixel PSF Algorithm Bias

The N -point image sequence p represents the noisy, sampled data from the edge profile. Some of the algorithms presented in this chapter, however, rely on the discrete differentiation of the edge profile. For these algorithms, the sampled edge profile data must be discretely differentiated prior to estimation of the edge location. We chose to use the convolution kernel $[1, -1]$ to differentiate the sampled data. Therefore, if Figure 5.3

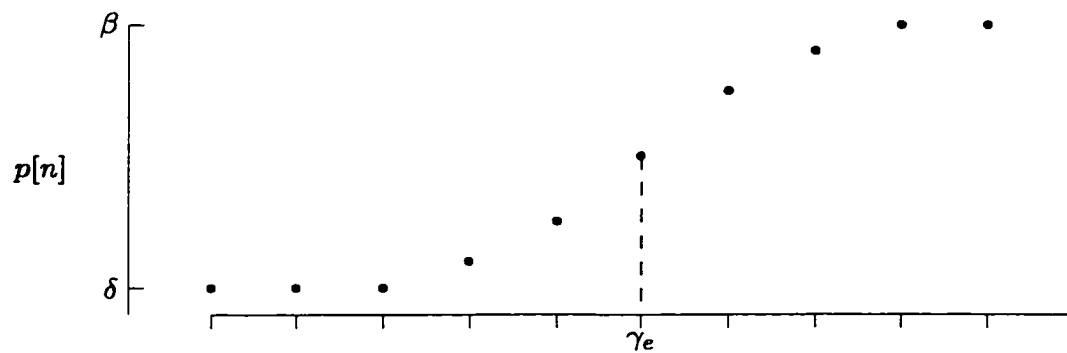


Figure 5.3: Sampled Edge Profile Data

represents the sampled edge profile data, then Figure 5.4

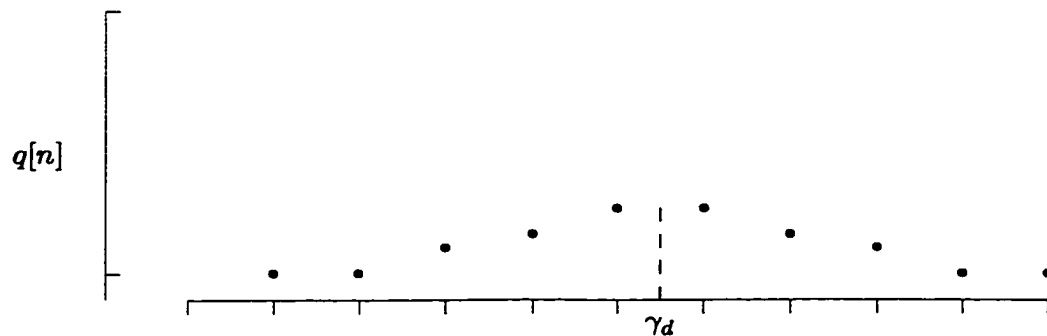


Figure 5.4: Discretely Differentiated Edge Profile Data

is the corresponding, discretely differentiated sampled edge profile data where $q[n] = p[n] -$

$p[n - 1]$. By overlapping these two figures in Figure 5.5, one can easily see that this convolution kernel introduces a half-pixel bias in the estimation of the edge location.

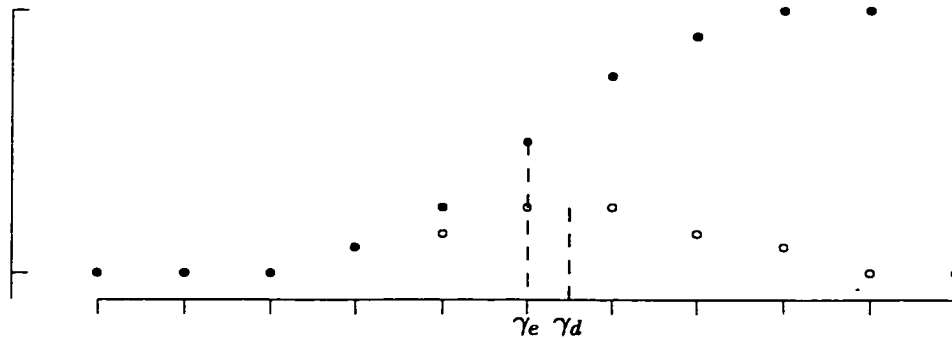


Figure 5.5: Half-pixel Bias Depiction

Therefore, all algorithms presented in this chapter that rely on the sampled PSF data will subtract off one half of the intersample distance from the estimation of the edge location in order to correct for this bias. The next section provides a table of the algorithms presented in this chapter followed by a more in depth discussion of each algorithm.

5.3 Edge Location Algorithms

Type	Label	Description
Global PSF	GPSF1	Discrete Differentiation: Inspired by calculus, this algorithm estimates the edge location by simple discrete differentiation followed by a search for a maximum.
	GPSF2	Discrete Differentiation with Means: The previous algorithm relies heavily on two independent data values only, making it highly susceptible to noise. This algorithm represents an attempt to lessen the impact of the noise while still holding to the theory of GPSF1.
Local PSF	LPSF1	Quadratic Interpolation: Using the values from $q[m-1]$, $q[m]$, and $q[m+1]$ (where $q[m]$ is the largest PSF value) a quadratic function is uniquely defined. From this quadratic equation, a maximum value is computed and used as the edge location estimate.
	LPSF2	Weighted Mean (Center of Mass): The expected value (or center of mass) is defined by $q[m-1]$, $q[m]$, and $q[m+1]$, where the lesser of $q[m-1]$ and $q[m+1]$ is used as the lower bound. A weighted mean value is used as the edge location estimate.
	LPSF3	Weighted Mean (Center of Mass) Revisited: This algorithm is a repeat of the previous algorithm with the lower bound equal to zero.
Global ESF	GESF1	Moment Matching: This algorithm attempts to estimate the edge location by matching theoretical statistical moments (mean, variance, skewness) to their corresponding experimental (sampled) statistical moments.
Local ESF	LESF1	Linear Interpolation: Estimation of the edge location is calculated by interpolating a line between two data values and returning the edge location estimate as the point at which the line crosses an average intensity value.
	LESF2	Cubic Convolution: Four consecutive ESF values are used to construct a spline between the inner two data values. A binary search is performed to determine the location at which the spline crosses an average intensity value.
	LESF3	Hermite Spline: This algorithm also uses a spline to estimate the edge location. The Hermite Spline computed is based on two ESF data values and their respective first derivative. A binary search is performed to determine the location at which the spline crosses an average intensity value.

5.3.1 Global PSF Algorithms

The algorithms presented in this section use the entire sampled edge profile data to estimate the edge location.

5.3.1.1 Approximation Using Discrete Differentiation (GPSF1)

The motivation for this algorithm comes directly from calculus. Calculating the edge location of a continuous ESF function requires differentiation of the edge function followed by a search for the maximum. Using the definition of the edge described in Subsection 5.1.2

$$g(x) = \delta + (\beta - \delta)H(x - \gamma)$$

and its corresponding derivative

$$g'(x) = (\beta - \delta)h(x - \gamma),$$

the computation of the expected value of $g'(x)$ is

$$\int_{-\infty}^{\infty} xg'(x) dx = \dots = (\beta - \delta)\gamma + (\beta - \delta) \int_{-\infty}^{\infty} zh(z) dz$$

where $z = x - \gamma$. From Equation 5.2

$$\int_{-\infty}^{\infty} zh(z) dz = 0.$$

Thus,

$$\int_{-\infty}^{\infty} xg'(x) dx = (\beta - \delta)\gamma.$$

Also

$$\int_{-\infty}^{\infty} g'(x) dx = (\beta - \delta)$$

and thus the edge location estimate is based on the centroid equation

$$\frac{\int_{-\infty}^{\infty} x g'(x) dx}{\int_{-\infty}^{\infty} g'(x) dx} = \frac{(\beta - \delta)\gamma}{(\beta - \delta)} = \gamma. \quad (5.3)$$

As stated previously, Equation 5.1 results from the assumption of a continuous ESF function. However, the data acquired from the edge profile are discrete. An equivalent version of Equation 5.1 which accounts for discrete data is ¹

$$\hat{\gamma}/\xi = \frac{\frac{1}{N} \sum_{n=1}^{N-1} n q[n]}{\frac{1}{N} \sum_{n=1}^{N-1} q[n]} - \frac{1}{2}, \quad (5.4)$$

where ξ represents the intersample distance. Simplification of Equation 5.4 results in the following:

$$\hat{\gamma}/\xi = \frac{\sum_{n=1}^{N-1} n(p[n] - p[n-1])}{\sum_{n=1}^{N-1} (p[n] - p[n-1])} - \frac{1}{2} = \dots = \left(\frac{p[N-1] - \bar{p}}{p[N-1] - p[0]} \right) N - \frac{1}{2} \quad (5.5)$$

where

$$\bar{p} = \frac{1}{N} \sum_{n=0}^{N-1} p[n].$$

Simulation confirms that this (bias-corrected) estimator is unbiased. One interesting feature of this algorithm is its apparent insensitivity to the width of the edge (α). Due to the

¹Recall that $q[n] = p[n] - p[n-1]$ for $n = 1, 2, \dots, N-1$ and the $1/2$ bias correction is due to the convolution of the data.

algorithm's reliance on the accuracy of $p[0]$ and $p[N - 1]$ and the possibility of these values being corrupted by the random noise inherently present, however, it generally produces poor results even when SNR is high. This indicates that even though the theory was sound in the continuous case, the model tends to break down when applied to discrete data.

Figure 5.6 shows the *root mean square error* of this algorithm based on a variety of input scenes. Each algorithm in this chapter was presented with the same input scenes. Various values of α (or edge width) were used ranging from approximately 0.5 to 2.0 pixel widths. In addition, 5 levels of SNR were used: 8 (indicated by ' \bullet '), 16 (indicated by ' \circ '), 32 (indicated by ' Δ '), 64 (indicated by ' \star '), and 128 (indicated by ' \times '). The results of each algorithm will be shown for comparison. The basis for comparison is the RMSE between the estimate produced by the algorithm and the actual edge location.

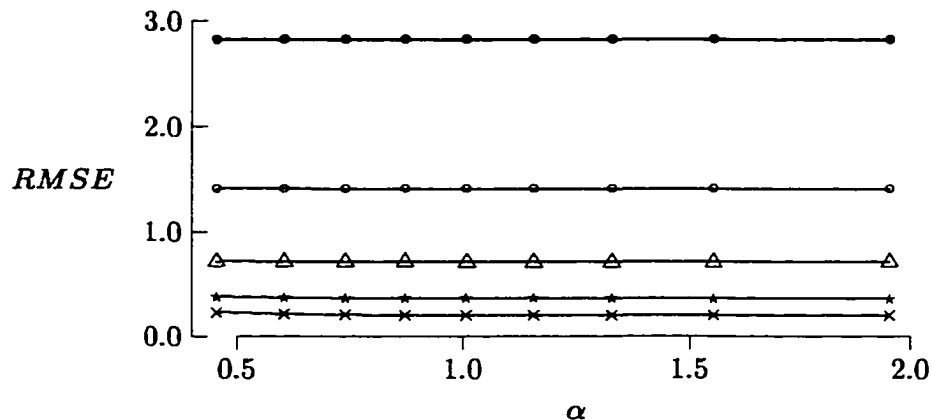


Figure 5.6: Discrete Differentiation Algorithm RMSE Results

Since the problem with this algorithm arises from the reliance on the accuracy of $p[0]$ and $p[N - 1]$, perhaps a modification can be made to make the algorithm more insensitive to noise. This observation prompted the next algorithm presented (GPSF2).

5.3.1.2 Approximation Using Discrete Differentiation with Means (GPSF2)

The motivation for this algorithm is based on the obvious sensitivity of the previous algorithm (GPSF1) to noise. If the sensitivity to noise can be decreased then perhaps the theory behind the design of the algorithm, i.e., the relationship to a continuous system and the computation of the maximum value of the discrete derivative, will be more influential and result in a reduced RMSE. With this in mind, the previous equation used to calculate $\hat{\gamma}$,

$$\hat{\gamma}/\xi = \left(\frac{p[N-1] - \bar{p}}{p[N-1] - p[0]} \right) N - \frac{1}{2},$$

is altered slightly.

The problem with the previous algorithm came from the possibility of noise corrupting the first ($p[0]$) or last ($p[N-1]$) values in the edge profile sequence. Because the equation to calculate $\hat{\gamma}$ relies heavily on the accuracy of these two values it would be expected that the algorithm would produce poor results when SNR is low. Thus, in an attempt to compensate for this sensitivity to noise, the averages of the upper and lower intensity values are computed and used instead.

We define the following quantities

$$\hat{\delta} = \frac{1}{K} \sum_{n=0}^{K-1} p[n] \quad \text{and} \quad \hat{\beta} = \frac{1}{K} \sum_{n=N-K}^{N-1} p[n]$$

where K is an input parameter chosen so that the $N - 2K$ data values in the middle of the input sequence represent all of the edge pixel values and the K data values at each end of the input sequence only contain the high intensity and low intensity background pixel values, as shown in Figure 5.7.

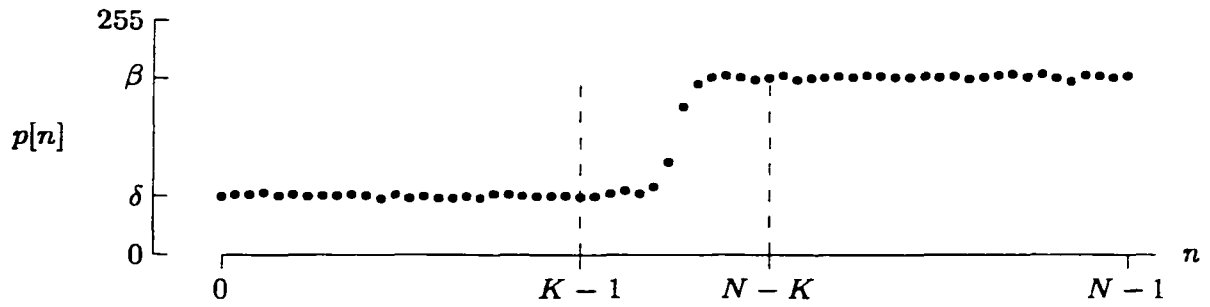


Figure 5.7: Representation of K in Edge Profile Data

Using these defined quantities, Equation 5.3 is changed to

$$\hat{\gamma}/\xi = \left(\frac{\hat{\beta} - \bar{p}}{\hat{\beta} - \hat{\delta}} \right) N - \frac{1}{2}. \quad (5.6)$$

Simulation confirms that this estimator is unbiased. In addition, this modification of the previous algorithm produces more accurate results, as indicated in Figure 5.8.

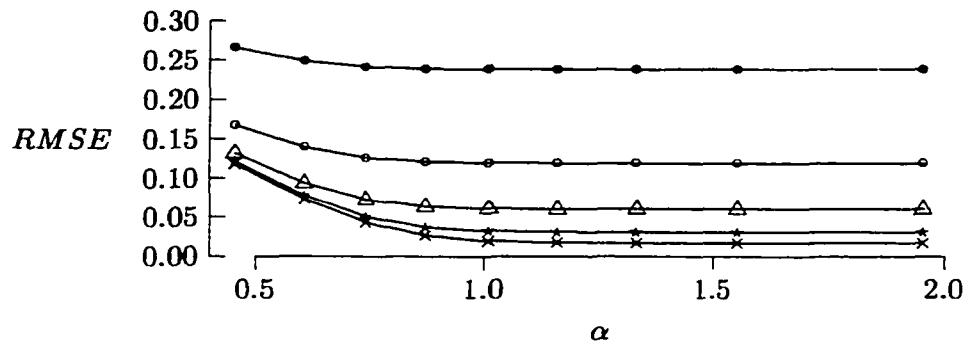


Figure 5.8: Discrete Differentiation with Means Algorithm RMSE Results

However, when compared to some of the other algorithms presented here, this algorithm does not perform as well.

5.3.2 Local PSF Algorithms

The algorithms presented in this section use a subset of the sampled edge profile data to estimate the edge location.

5.3.2.1 Quadratic Interpolation (LPSF1)

The motivation for this algorithm is that the first derivative of the (continuous) image function $g = s \otimes h$ is a multiple of the PSF. That is,

$$g'(x) = (\beta - \delta) h(x - \gamma).$$

If we assume that $h(x)$ has a unique maximum at $x = 0$ then the function g' has its maximum at $x = \gamma$. That is, the subpixel edge location γ can be found by maximizing $g'(x)$. In practice, however, the function g is unknown; what we know instead is the N -point sequence p — a noisy, sampled representation of g [43]. The *local quadratic PSF interpolation* algorithm deals with this problem by using the $(N - 1)$ -point q sequence. A search is performed on the sequence q for a maximum to find the approximate location of γ and local quadratic interpolation is used to refine the location estimate as illustrated in Figure 5.9.

To be specific, let m be the pixel for which $q[m]$ is largest. If there are several such pixels, the pixel closest to $(N - 1)/2$ is selected. A quadratic polynomial is then uniquely defined by the three pixel values $q[m - 1]$, $q[m]$, and $q[m + 1]$. The location of the maximum value of this quadratic polynomial is determined by standard calculus techniques; the result is

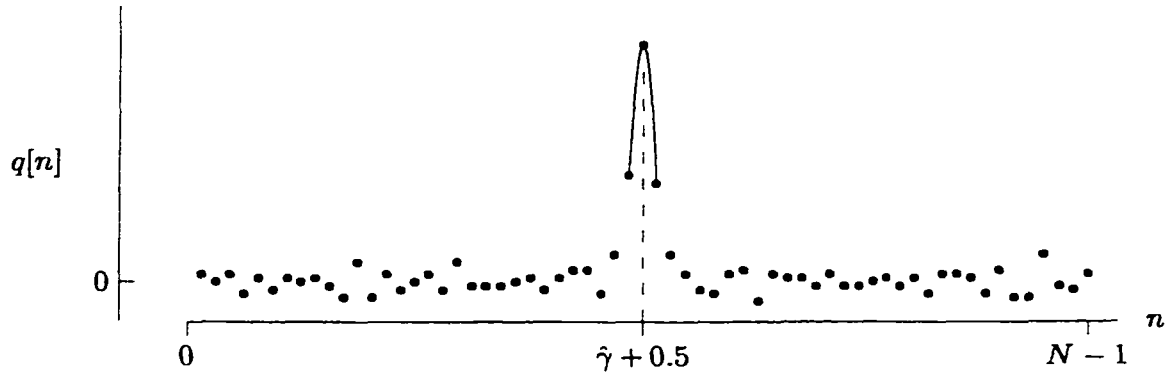


Figure 5.9: Quadratic Interpolation

a *biased* estimate of the subpixel edge location estimate. If the half-pixel bias is removed, computation of the estimated edge location is based on the equation

$$\hat{\gamma}/\xi = m + \frac{q[m+1] - q[m]}{2q[m] - q[m+1] - q[m-1]}. \quad (5.7)$$

Simulation confirms that this (bias-corrected) estimator is unbiased, provided the PSF has a unique maximum at $x = 0$.

Because this algorithm depends on differencing noisy data, it is intuitive that this algorithm may perform poorly if the SNR is low. Furthermore, if the PSF width is not large the algorithm may perform poorly because the quadratic interpolation will not be appropriate.

As shown in Figure 5.10, both of these conjectures are true. The algorithm performs poorly when SNR is low. Also, as the PSF width (α) decreases, the RMSE increases.

5.3.2.2 Weighted Mean (Center of Mass) (LPSF2)

The motivation for this algorithm is similar to the Local Quadratic PSF Interpolation algorithm. Perhaps finding the center of mass of the three internal PSF points will produce

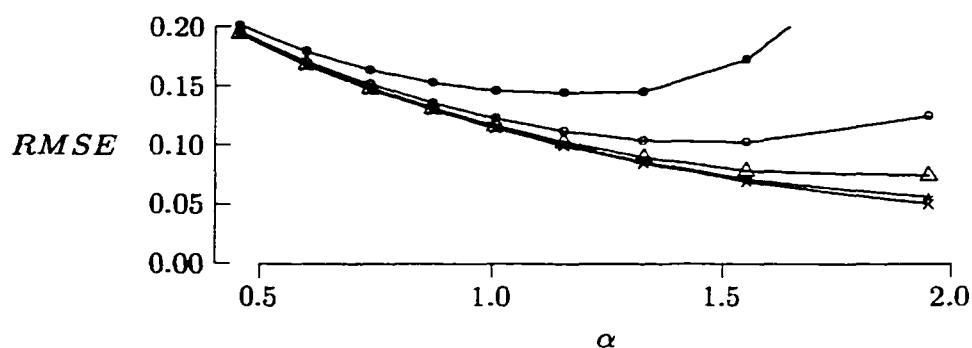


Figure 5.10: Quadratic Interpolation Algorithm RMSE Results

a better estimate than fitting a quadratic to them. Using the points $q[m-1]$, $q[m]$, and $q[m+1]$ where m is the pixel for which $q[m]$ is the largest, the edge estimate $\hat{\gamma}$ is calculated. This algorithm, taken from Seitz[56], uses the smallest of the three values as a lower bound on the computation of the center of mass.

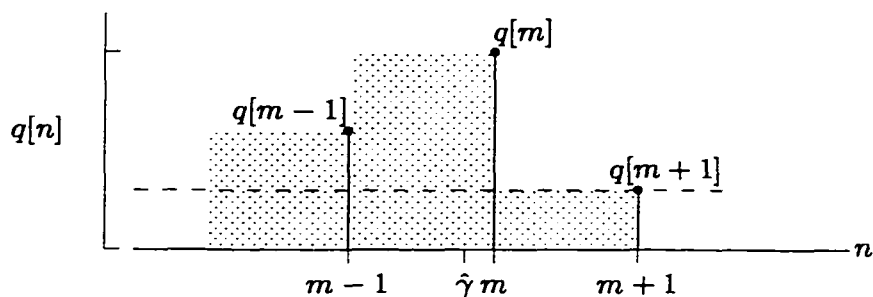


Figure 5.11: Center of Mass Calculation

The calculation of the center of mass of the region defined by the shaded area in Figure 5.11 is

$$\hat{\gamma}/\xi = \frac{(m-1)q[m-1] + mq[m] + (m+1)q[m+1]}{q[m-1] + q[m] + q[m+1]} - \frac{1}{2}$$

However, since Seitz chose to make the smallest of the three values the lower bound, the equation to calculate the edge estimate is based on the center of mass defined by the shaded

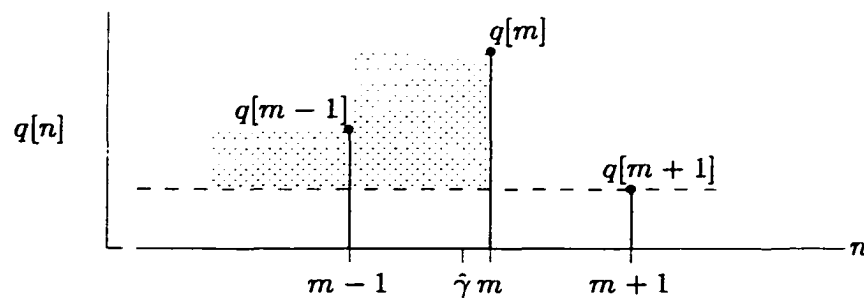


Figure 5.12: Center of Mass Calculation, smallest q value lower-bound

The area of the shaded region is calculated as

$$\hat{\gamma}/\xi = \frac{(m-1)(q[m-1] - q[m+1]) + m(q[m] - q[m+1])}{(q[m-1] - q[m+1]) + (q[m] - q[m+1])} - \frac{1}{2}$$

or

$$\hat{\gamma}/\xi = m + \frac{q[m+1] - q[m-1]}{2q[m+1] - q[m] - q[m-1]} - \frac{1}{2}.$$

Thus, taking into consideration that either $q[m-1]$ or $q[m+1]$ might be smaller, or that they may be equal, the following equation is used to determine the edge estimate.

$$\hat{\gamma}/\xi = \begin{cases} m + \frac{q[m-1] - q[m+1]}{2q[m+1] - q[m] - q[m-1]} - \frac{1}{2}, & \text{if } q[m+1] < q[m-1] \\ m + \frac{q[m-1] - q[m+1]}{2q[m-1] - q[m] - q[m+1]} - \frac{1}{2}, & \text{if } q[m+1] > q[m-1] \\ m - \frac{1}{2}, & \text{if } q[m+1] = q[m-1] \end{cases} \quad (5.8)$$

Simulation confirms that this (bias-corrected) estimator is unbiased. Figure 5.13 shows, however, that this estimator is most accurate when α is approximately 1.0. This estimator decreases in accuracy as α increases and SNR decreases.

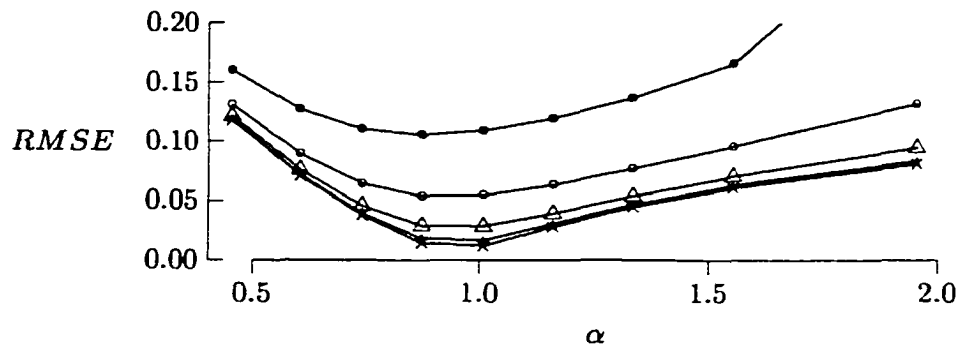


Figure 5.13: Center of Mass Algorithm RMSE Results

5.3.2.3 Weighted Mean (Center of Mass) Revisited (LPSF3)

This algorithm is an extended investigation prompted by Seitz's algorithm presented in the previous subsection. Once again, the algorithm uses the largest value of the PSF sequence q and its two nearest neighboring values to calculate the center of mass. Seitz's algorithm uses the smallest of these three values as its lower bound when computing the edge estimate. The difference between this algorithm and the previous algorithm is that the lower bound used for this algorithm is 0. Thus, $\hat{\gamma}$ is calculated as the center of mass of the shaded region illustrated in Figure 5.14.

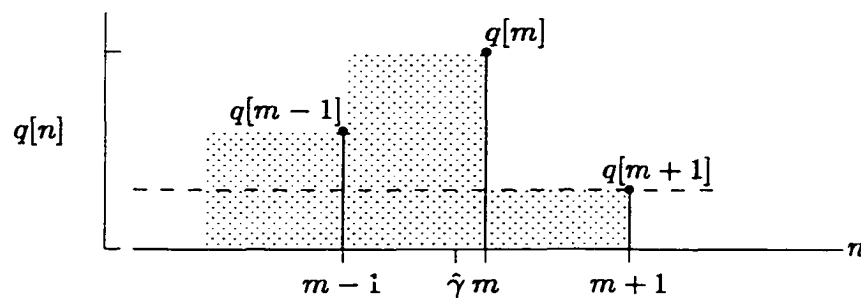


Figure 5.14: Center of Mass Calculation, zero lower-bound

The theory and most of the computation are identical to the previous algorithm. The equation used to determine the edge estimate, however, is slightly different. The equation

used for this algorithm is

$$\hat{\gamma}/\xi = \frac{(m-1)q[m-1] + m q[m] + (m+1)q[m+1]}{q[m-1] + q[m] + q[m+1]} - \frac{1}{2}$$

or

$$\hat{\gamma}/\xi = m + \frac{q[m+1] - q[m-1]}{q[m-1] + q[m] + q[m+1]} - \frac{1}{2}. \quad (5.9)$$

Simulation confirms that this (bias-corrected) estimator is unbiased. Comparison with LPSF2 shows that the performance of LPSF3 is essentially the same as that of LPSF2. The difference between the two algorithms, namely the assignment of the lower bound, appears to have little effect on the accuracy of the algorithm except when α is large (≈ 2.0) and the SNR is low (≈ 8.0). Figure 5.15 shows the RMSE results produced by this algorithm.

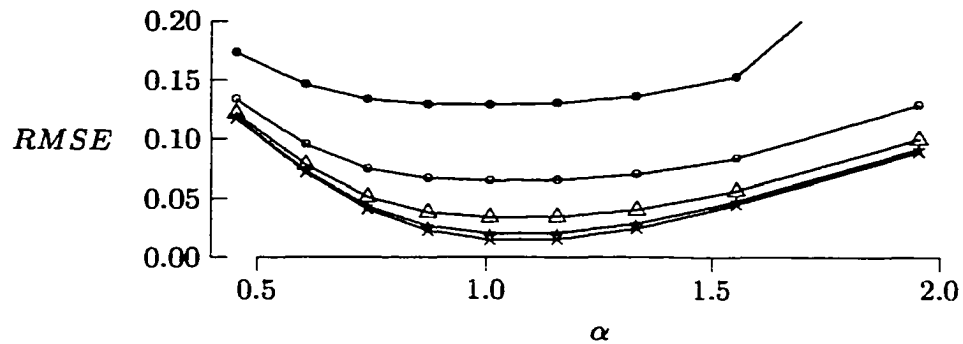


Figure 5.15: Center of Mass Revisited Algorithm RMSE Results

5.3.3 Global ESF Algorithms

The algorithm presented in this section uses the original ESF sampled edge profile data.

The entire sequence is used to estimate the edge location.

5.3.3.1 Moment Matching (GESF1)

Unlike the previous algorithms, the *global ESF moment matching* algorithm is motivated by analogy to a common statistical approach in which theoretical statistical moments (mean, variance, skewness, etc.) are matched to their corresponding experimental (sample) statistical moments[43]. One moment is matched for each unknown model parameter; the resulting set of equations, one per matched moment, is then solved to determine the parameter values. No search of the p (or q) sequence is required[43].

In a subpixel edge location application the three ideal edge parameters β , γ , δ can be determined by matching the first three moments. That is, the mean, variance and skewness of the three-parameter ideal edge function $s(x) = \delta + (\beta - \delta)u(x)$ are defined in terms of β , γ , δ and N as

$$\begin{aligned}\bar{s} &= \frac{1}{N} \int_{-1/2}^{N-1/2} s(x) dx \\ &= \frac{(\gamma - 1/2)\delta + (N - \gamma - 1/2)\beta}{N},\end{aligned}$$

$$\begin{aligned}\langle (s - \bar{s})^2 \rangle &= \frac{1}{N} \int_{-1/2}^{N-1/2} (s(x) - \bar{s})^2 dx \\ &= \frac{(\gamma - 1/2)(\delta - \bar{s})^2 + (N - \gamma - 1/2)(\beta - \bar{s})^2}{N},\end{aligned}$$

and

$$\langle (s - \bar{s})^3 \rangle = \frac{1}{N} \int_{-1/2}^{N-1/2} (s(x) - \bar{s})^3 dx$$

$$= \frac{(\gamma - 1/2)(\delta - \bar{s})^3 + (N - \gamma - 1/2)(\beta - \bar{s})^3}{N}$$

respectively. The corresponding mean, variance and skewness of the image sequence p are computed directly from the three equations

$$\bar{p} = \frac{1}{N} \sum_{n=0}^{N-1} p[n] \quad \langle (p - \bar{p})^2 \rangle = \frac{1}{N} \sum_{n=0}^{N-1} (p[n] - \bar{p})^2 \quad \langle (p - \bar{p})^3 \rangle = \frac{1}{N} \sum_{n=0}^{N-1} (p[n] - \bar{p})^3$$

respectively. Then, given numerical values for \bar{p} , $\langle (p - \bar{p})^2 \rangle$ and $\langle (p - \bar{p})^3 \rangle$, the three equations

$$\bar{s} = \bar{p} \quad \langle (s - \bar{s})^2 \rangle = \langle (p - \bar{p})^2 \rangle \quad \langle (s - \bar{s})^3 \rangle = \langle (p - \bar{p})^3 \rangle$$

can be solved for β , γ and δ . The solution for γ , interpreted as an estimator for the subpixel edge location, is

$$\hat{\gamma} = \xi \left(\frac{N-1}{2} + \frac{N}{2} \left(\frac{z}{\sqrt{1+z^2}} \right) \right) \quad (5.10)$$

where z is the dimensionless ratio

$$z = \frac{\langle (p - \bar{p})^3 \rangle}{2 \langle (p - \bar{p})^2 \rangle^{3/2}}.$$

Simulation confirms that this estimator is unbiased.² Figure 5.16 shows the RMSE results produced by this algorithm.

²Although not used explicitly, if needed, the dark and bright background parameter estimates are

$$\hat{\delta} = \bar{p} - s \left(\sqrt{1+z^2} - z \right) - 1/2 \quad \text{and} \quad \hat{\beta} = \bar{p} + s \left(\sqrt{1+z^2} + z \right) + 1/2$$

respectively.

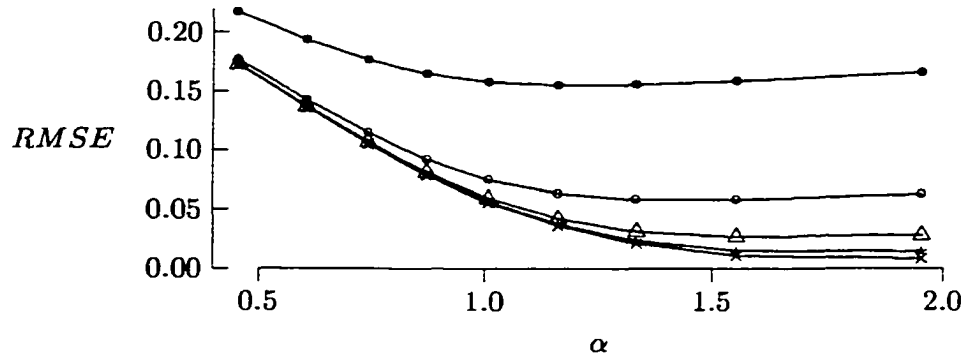


Figure 5.16: Moment Matching Algorithm RMSE Results

5.3.4 Local ESF Algorithms

The algorithms presented in this section use the original ESF sampled edge profile data. A subset of the sequence p is used to estimate the edge location.

5.3.4.1 Linear Interpolation (LESF1)

The motivation for this algorithm is that the (continuous) image function $g = s \otimes h$ is monotone increasing from δ to β and so must cross the line $g(x) = (\delta + \beta)/2$ at exactly one value of x . Moreover, this crossing occurs when

$$g(x) = \delta + (\beta - \delta)H(x - \gamma) = \frac{\delta + \beta}{2}$$

which is equivalent to the condition

$$H(x - \gamma) = \frac{1}{2}.$$

As discussed previously, provided the PSF is symmetric about $x = 0$, the condition $H(x - \gamma) = 1/2$ is equivalent to $x = \gamma$. That is, the subpixel edge location can be found by solving the equation $g(x) = (\delta + \beta)/2$. In practice, however, the function g is unknown; what we

know instead is the N -point sequence p — a noisy, sampled representation of g . The *local linear ESF interpolation* algorithm deals with this problem by searching the sequence p to find the approximate location of γ , then using local linear interpolation to refine the location estimate[43], as indicated in Figure 5.17.

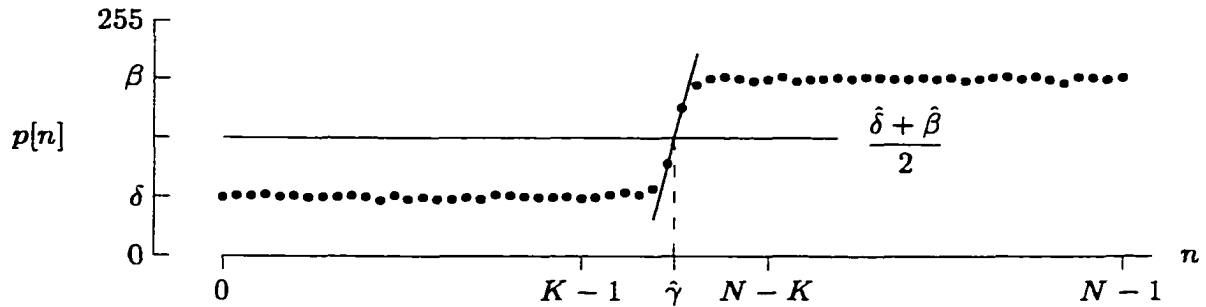


Figure 5.17: Linear Interpolation Calculation

In addition to the sequence p , this algorithm also requires knowledge of the positive integer K , described in subsection 5.3.1.2. The intent is to guarantee that the image sequence has been selected so that the $N - 2K$ pixels in the middle of the sequence contain all the edge pixel values and the K pixels at each end of the sequence contain background pixel values only.

Consistent with the three-parameter ideal edge function model, the image sequence rises from dark to bright. That is, the first K pixels represent dark background pixel values, the last K pixels represent bright background pixel values, and generally — although not necessarily, because the image sequence pixel values are noisy — the image sequence is monotone increasing for the $N - 2K$ pixels in the middle of the sequence. As defined previously, the dark and bright background edge parameters δ , β are estimated from the

background pixel values as

$$\hat{\delta} = \frac{1}{K} \sum_{n=0}^{K-1} p[n] \quad \text{and} \quad \hat{\beta} = \frac{1}{K} \sum_{n=N-K}^{N-1} p[n]$$

respectively. The average $(\hat{\delta} + \hat{\beta})/2$ is used as the basis for a linear search of the image sequence to find the pixel m such that $p[m-1] < (\hat{\delta} + \hat{\beta})/2 \leq p[m]$. If there are several such pixels, the one closest to $(N-1)/2$ is selected. The subpixel edge location estimate is then computed as

$$\hat{\gamma} = \xi \left(m - 1 + \frac{(\hat{\delta} + \hat{\beta})/2 - p[m-1]}{p[m] - p[m-1]} \right). \quad (5.11)$$

Simulation confirms that this estimator is unbiased (provided the PSF is symmetric about $x = 0$.) Figure 5.18 shows the RMSE results produced by this algorithm.

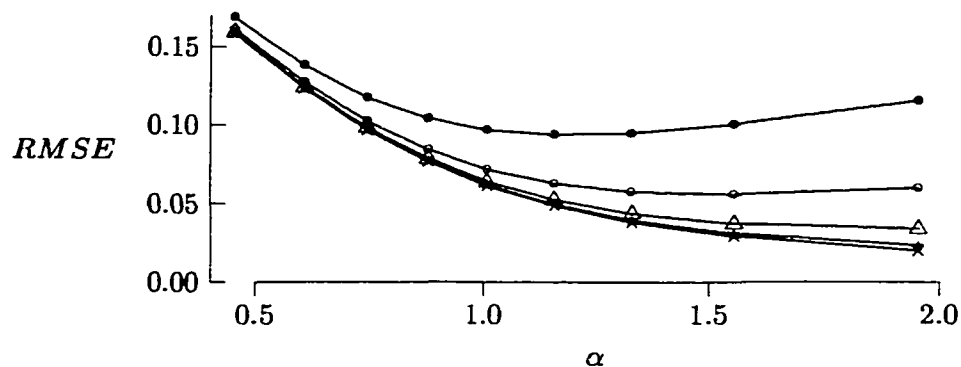


Figure 5.18: Linear Interpolation Algorithm RMSE Results

Note that the scale of the RMSE represented in the graphs changes based on the algorithm presented. For example, the error incurred by this algorithm is generally less than the error incurred by the previous algorithms presented.

5.3.4.2 Cubic Convolution (LESF2)

The motivation for the next two algorithms comes from an attempt to improve on the performance of the linear interpolation algorithm presented in Subsection 5.3.4.1. Perhaps a different means of connecting the two internal points on the ESF will produce a smaller error as compared to a simple linear interpolation.

A first attempt at this improvement comes from Reichenbach [50]. A search is performed to find the four edge profile data values $p[m-2]$, $p[m-1]$, $p[m]$ and $p[m+1]$ such that $p[m-1] < (\hat{\beta} + \hat{\delta})/2 < p[m]$. Using these four data values, a spline is constructed between the two internal pixels, as indicated in Figure 5.19.

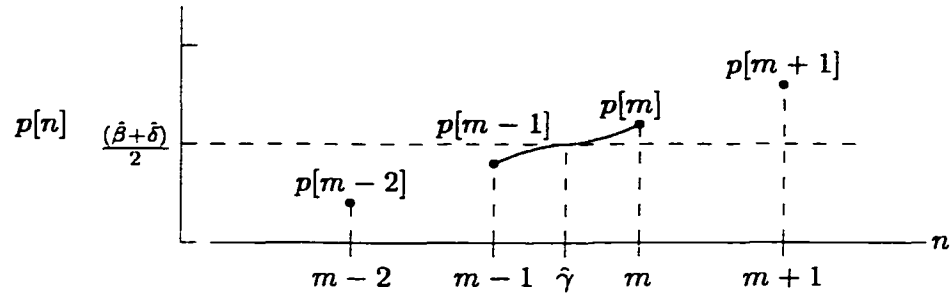


Figure 5.19: Cubic Convolution Calculation

The equation for the spline that Reichenbach[50], and subsequently we, use is

$$\begin{aligned}
 p(x) &= p[m-2](-(x+1)^3 + 5(x+1)^2 - 8(x+1) + 4) \\
 &\quad + p[m-1](x^3 - 2x^2 + 1) \\
 &\quad + p[m]((1-x)^3 - 2(1-x)^2 + 1) \\
 &\quad + p[m+1](-(-2-x)^3 + 5(2-x)^2 - 8(2-x) + 4).
 \end{aligned}
 \tag{5.12}$$

The spline equation is defined such that

$$p(0) = p[m - 1]$$

and

$$p(1) = p[m].$$

Thus, a search for the x value, such that $0 \leq x \leq 1$, is needed. The value returned by the algorithm as the estimated edge location is calculated as $m - 1$ plus the x value at which $p(x)$ (from Equation 5.10) crosses $(\hat{\beta} + \hat{\delta})/2$. Therefore, the algorithm calculates the x value such that

$$p(x) = \frac{\hat{\beta} + \hat{\delta}}{2} \quad \text{or} \quad p^{-1}\left(\frac{\hat{\beta} + \hat{\delta}}{2}\right) = x.$$

The estimated edge location is thus defined to be

$$\hat{\gamma} = (m - 1 + x)\xi.$$

Computation of the inverse of the spline function is required to obtain the desired x value. Due to the complexity of computing $p^{-1}(x)$, however, we chose instead to converge to a solution by bisecting the interval 0 to 1 (between $p[m - 1]$ and $p[m]$) until an acceptable accuracy³ was achieved.

Figure 5.20 confirms that this unbiased estimator actually results in a lower RMSE than the linear interpolation when the edge is sharp (i.e., α is small). If, however, α is large, this estimator performs worse than the linear interpolation algorithm.

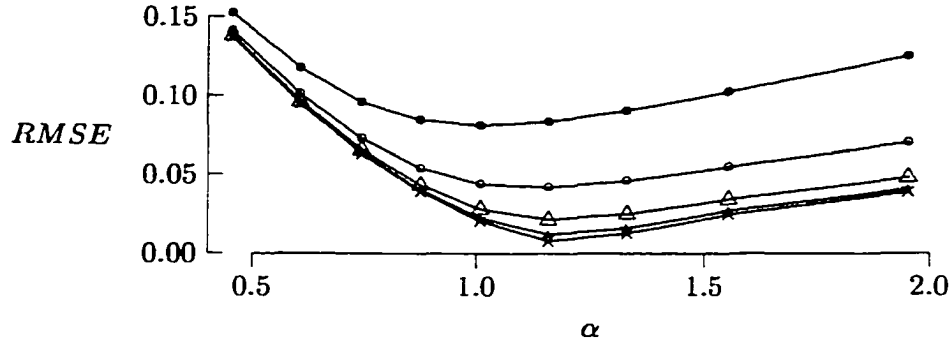


Figure 5.20: Cubic Convolution Algorithm RMSE Results

5.3.4.3 Hermite Spline (LESF3)

The motivation for this algorithm was not only to improve on the linear interpolation algorithm, but also to correct the apparent failure of the previous spline algorithm to mimic the ESF models used as input. The *local ESF Hermite Spline* algorithm is an attempt to more closely fit a spline to the input ESF models. Again using the central four pixels from the sampled edge profile data, the discrete derivatives of the inner two pixels are computed and used in conjunction with the values of the pixels. The equation used to construct the spline is

$$\begin{aligned}
 p(x) = & p[m-1](2x^3 - 3x^2 + 1) + Dp[m-1](x^3 - 2x^2 + x) \\
 & + p[m](-2x^3 + 3x^2) + Dp[m](x^3 - x^2)
 \end{aligned}$$

where $Dp[m]$ represents the discrete derivative at pixel m and is calculated as

$$Dp[m] = \frac{p[m+1] - p[m-1]}{2}.$$

³For the results presented here, accuracy of the bisection method was taken to 1/10000th of a pixel width.

As with the previous spline algorithm $p(0) = p[m - 1]$ and $p(1) = p[m]$. Thus, once again, a search for the x value, between 0 and 1, at which point $p(x)$ crosses $(\hat{\beta} + \hat{\delta})/2$ is needed.

A linear search to find the m such that

$$p[m - 1] < \frac{\hat{\beta} + \hat{\delta}}{2} \leq p[m]$$

is performed. After finding m through a linear search of the sequence p , the x value satisfying

$$p(x) = \frac{\hat{\beta} + \hat{\delta}}{2} \quad \text{or} \quad p^{-1}\left(\frac{\hat{\beta} + \hat{\delta}}{2}\right) = x,$$

must be found. The estimated edge location is graphically shown in Figure 5.21 and is computed as

$$\hat{\gamma} = (m - 1 + x)\xi.$$

Bisection was again used in this algorithm to determine x , and subsequently $\hat{\gamma}$.

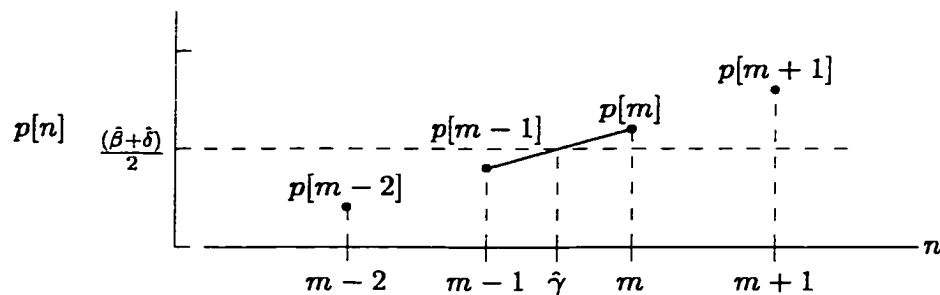


Figure 5.21: Hermite Spline Calculation

Simulation confirms that this estimator is unbiased. Results from the simulation, depicted in Figure 5.22, also show that this algorithm performs better than the previous spline

algorithm when the ESF's edge width is wide ($\alpha > 1$), however, it never produces a lower RMSE than the linear interpolation algorithm for a given α and SNR combination.

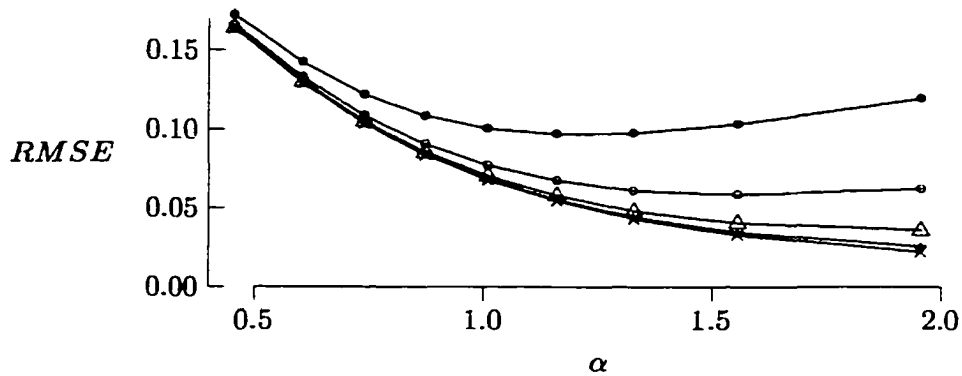


Figure 5.22: Hermite Spline Algorithm RMSE Results

One potential disadvantage of this algorithm (and the previous) is the use of a bisection method in order to obtain the x value needed. This process, depending on the accuracy desired, will compute the estimated edge location slower than other algorithms presented in this chapter that simply use the data in an equation to compute the edge location.

5.4 Conclusion

In this chapter we presented various algorithms that can be used to estimate the edge location from a given sampled edge profile. These algorithms were grouped into four categories based on how the edge profile array data is used. Each algorithm was used to estimate edge locations from pre-defined edge profiles based on the SNR and edge width (α). Figure 5.23 shows a table depicting the algorithms which produced the best results (lowest RMSE) for the SNR and α specified.

		Signal-to-Noise Ratio				
		8	16	32	64	128
Edge	0.4549	LESF2	LPSF2	LPSF2	LPSF2	LPSF2
	0.6052	LESF2	LPSF2	LPSF2	LPSF2	LPSF2
	0.7406	LESF2	LPSF2	LPSF2	LPSF2	LPSF2
	0.8725	LESF2	LPSF2	LPSF2	LPSF2	LPSF2
Width	1.0083	LESF2	LESF2	LESF2	LPSF2	LPSF2
	(α)	1.1561	LESF2	LESF2	LESF2	LESF2
		1.3286	LESF2	LESF2	LESF2	LESF2
		1.5542	LESF1	LESF2	GESF1	GESF1
		1.9530	LESF1	LESF1	GESF1	GESF1

Figure 5.23: Algorithms Producing Lowest RMSE

Based on the results presented in this table, it can be seen that algorithms that use a subset of the edge profile data, localized near the edge location, perform better than algorithms that use all of the edge profile data. Only when the SNR is high (> 16) and the edge width is large (> 1.3) does a global algorithm (Moment Matching: GESF1) perform better than the local algorithms. Also, algorithms that use the Point Spread Function generate a lower RMSE when SNR is high and the edge width is small. As the SNR decreases and the edge width increases, the algorithms that use the Edge Spread Function begin to perform better.

Consistent with [43]

- All algorithms (except GPSF1) are sensitive to changes in α .

- All algorithms are sensitive to changes in SNR.
- Algorithm performance improves when α increases (except for some algorithms when SNR is low).

Although the table presented here may be useful, generally the SNR and edge width information is unknown. Therefore, it is prudent to choose an algorithm that performs well (if not optimally) across all possible scenarios. The results presented in this chapter indicate that the Linear Interpolation Algorithm (LESF1) represents a good candidate for this choice. Not only does LESF1 perform well across most of the scenarios presented, but it is also a computationally simple algorithm that requires only the specification of one additional parameter (K).

In the previous chapters we discussed the means by which the blurring which occurs during acquisition (based on the Optical Transfer Function) can be filtered through restoration. Another form of degradation which occurs during acquisition is aliasing. Aliasing is a noise effect that occurs when a scene is undersampled. The next chapter discusses a means by which aliasing can be suppressed.

Chapter 6

Microscanning

6.1 System Model with Microscanning

The system model defined in Chapter 2 is the basis for a *single scan* simulation. That is, a single scene is blurred, sampled, restored, and reconstructed. The modification to the model in this chapter is the addition of a microscan feature. Microscanning (or “multi-frame image acquisition”[1, 30]) consists of acquiring multiple digital images of the same scene with subpixel shift differences among the acquired images. Thus, the acquisition part of the model (Figure 6.1)

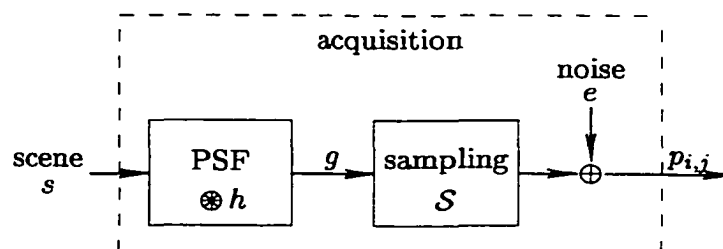


Figure 6.1: Acquisition Model

is performed for each of the microscanned images p_{m_1, m_2} for $0 \leq m_1 < M_1$ and $0 \leq m_2 < M_2$ where M_1 is the number of microscans in the x_1 direction and M_2 is the number of microscans in the x_2 direction. As shown in Figure 6.2, once all the microscanned images have been acquired, a composition of the images is performed to produce p .

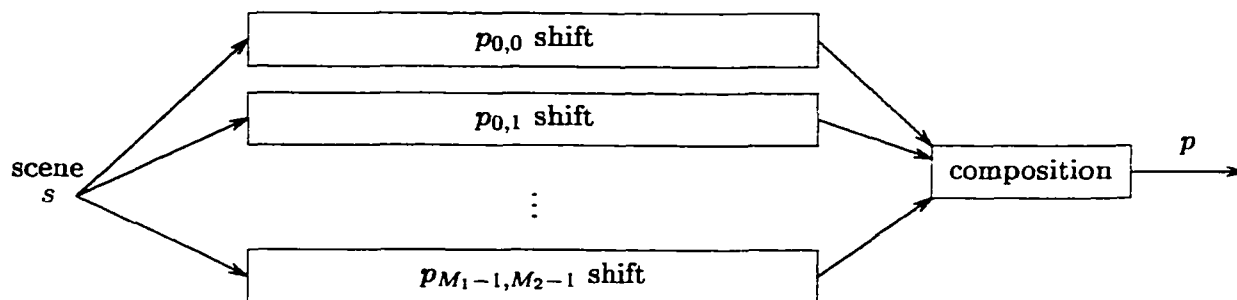


Figure 6.2: Microscan Acquisition and Composition

It is assumed the subpixel shifts between scans are equal. Thus, using an $M_1 \times M_2$ microscan method, each scan would differ by a ξ_1/M_1 shift of the acquisition device in the x_1 direction and a ξ_2/M_2 shift in the x_2 direction. In the spatial domain, the composition is performed by an interlacing of the acquired images into one composite image that has $M_1 M_2$ times the sampling density of the individual scans[2, 27]. A spatial domain composition of these $M_1 M_2$ scans would be computed by

$$p_c[M_1 n_1 + m_1, M_2 n_2 + m_2] = p_{m_1, m_2}[n_1, n_2] \quad (6.1)$$

where p_c represents the $M_1 N_1 \times M_2 N_2$ composite image. Composition of the images in the Fourier domain requires the summation of phase shifted transforms of the individual scans. That is, the following derives the equation for composition in the frequency domain. A specific 2×2 version of this can be found in [52]. A generalized equation for any microscanning

rate follows.

From Equation 2.12 we know that

$$\hat{p}_c[\nu_1, \nu_2] = \frac{1}{M_1 N_1 M_2 N_2} \sum_{n_1=0}^{M_1 N_1 - 1} \sum_{n_2=0}^{M_2 N_2 - 1} p_c[n_1, n_2] \exp(-i2\pi\nu_1 n_2 / M_1 N_1) \exp(-i2\pi\nu_2 n_2 / M_2 N_2).$$

The above equation can be rewritten and simplified as

$$\begin{aligned} \hat{p}_c[\nu_1, \nu_2] &= \frac{1}{M_1 M_2} \sum_{m_1=0}^{M_1-1} \sum_{m_2=0}^{M_2-1} \frac{1}{N_1 N_2} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} p_c[M_1 n_1 + m_1, M_2 n_2 + m_2] \\ &\quad \exp\left(-i2\pi\nu_1 \left(\frac{M_1 n_1 + m_1}{M_1 N_1}\right)\right) \exp\left(-i2\pi\nu_2 \left(\frac{M_2 n_2 + m_2}{M_2 N_2}\right)\right) \\ &= \frac{1}{M_1 M_2} \sum_{m_1=0}^{M_1-1} \sum_{m_2=0}^{M_2-1} \frac{1}{N_1 N_2} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} p_{m_1, m_2}[n_1, n_2] \\ &\quad \exp\left(-i2\pi\nu_1 \left(\frac{M_1 n_1 + m_1}{M_1 N_1}\right)\right) \exp\left(-i2\pi\nu_2 \left(\frac{M_2 n_2 + m_2}{M_2 N_2}\right)\right) \\ &= \frac{1}{M_1 M_2} \sum_{m_1=0}^{M_1-1} \sum_{m_2=0}^{M_2-1} \left(\frac{1}{N_1 N_2} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} p_{m_1, m_2}[n_1, n_2] \right. \\ &\quad \left. \exp(-i2\pi\nu_1 n_1 / N_1) \exp(-i2\pi\nu_2 n_2 / N_2) \right) \\ &\quad \exp(-i2\pi\nu_1 m_1 / M_1 N_1) \exp(-i2\pi\nu_2 m_2 / M_2 N_2). \end{aligned}$$

Once again, from Equation 2.12 we recognize that

$$\hat{p}_{m_1, m_2}[\nu_1, \nu_2] = \frac{1}{N_1 N_2} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} p_{m_1, m_2}[n_1, n_2] \exp(-i2\pi\nu_1 n_2 / N_1) \exp(-i2\pi\nu_2 n_2 / N_2).$$

Thus, the composite image, computed in the Fourier domain, is defined as the summation of the Fourier transforms of the individual scans phase shifted relative to their microscan

position, or

$$\hat{p}_c[\nu_1, \nu_2] = \frac{1}{M_1 M_2} \sum_{m_1=0}^{M_1-1} \sum_{m_2=0}^{M_2-1} \hat{p}_{m_1, m_2}[\nu_1, \nu_2] \exp \left(-i2\pi \left(\frac{\nu_1 m_1}{N_1 M_1} + \frac{\nu_2 m_2}{N_2 M_2} \right) \right). \quad (6.2)$$

The rest of the system model remains unchanged, except with respect to the size of the input expected. An $M_1 \times M_2$ microscan would result in an increase to N_1 by a factor of M_1 in the x_1 direction and to N_2 by a factor of M_2 in the x_2 direction. Equivalently, the intersample distance ξ_1 in the x_1 direction would decrease by a factor of M_1 and the intersample distance ξ_2 in the x_2 direction would decrease by a factor of M_2 . Due to this inverse relationship between the sample density and the intersample distance, the period for microscanned images P_{ms} is the same as that for single scanned images P_{ss} , i.e., $P_1 \times P_2$ never changes:

$$P_{1,ms} = (M_1 N_1) \left(\frac{\xi_1}{M_1} \right) = N_1 \xi_1 = P_{1,ss}$$

and similarly for P_2 .

6.2 CLS Filter with Microscanning

Based on the observation that the period remains constant despite an increased microscan rate, the application of the CLS restoration filter to the composite image can be accomplished by scaling the sample sizes N_1, N_2 by their respective microscan rate M_1, M_2 and by scaling the intersample distances ξ_1, ξ_2 by the inverse of their respective microscan rate $1/M_1, 1/M_2$. Therefore,

$$\hat{f}[\nu_1, \nu_2] = \frac{\xi'_1 \xi'_2 \langle \hat{H}^* \hat{D}^* \rangle(\omega_1, \omega_2)}{\left| \langle \hat{H} \hat{D} \rangle(\omega_1, \omega_2) \right|^2 + \alpha \langle |\hat{C} \hat{D}|^2 \rangle(\omega_1, \omega_2)} \quad (6.3)$$

where

$$\langle \hat{H}^* \hat{D}^* \rangle = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} \hat{H}^*(\omega_1 - k_1/\xi'_1, \omega_2 - k_2/\xi'_2) \hat{D}^*(\omega_1 - k_1/\xi'_1, \omega_2 - k_2/\xi'_2),$$

$$\langle |\hat{C} \hat{D}|^2 \rangle = \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} \left| \hat{C}(\omega_1 - k_1/\xi'_1, \omega_2 - k_2/\xi'_2) \hat{D}(\omega_1 - k_1/\xi'_1, \omega_2 - k_2/\xi'_2) \right|^2,$$

$\xi'_1 = \xi_1/M_1$, and $\xi'_2 = \xi_2/M_2$. Equation 6.3 represents the Constrained Least Squares Restoration filter for microscanned images.

6.3 The Nyquist Frequency and Aliasing

Images that have been acquired using a single scan can not be restored and reconstructed accurately beyond the Nyquist frequency of the acquisition device. Suppose, however, that for a particular input scene, energy is present at frequencies beyond the Nyquist frequency. Without physically adding more detectors to the sampling device, accurate reconstruction of those high-frequency features could rely on the use of microscanning (or some other means of increasing the sampling density). Microscanning, image composition, and restoration can be used to accurately restore pixel-scale features[44].

6.3.1 Nyquist Frequency relative to the OTF

Figure 6.3 shows three possible optical transfer functions for a digital image acquisition device.

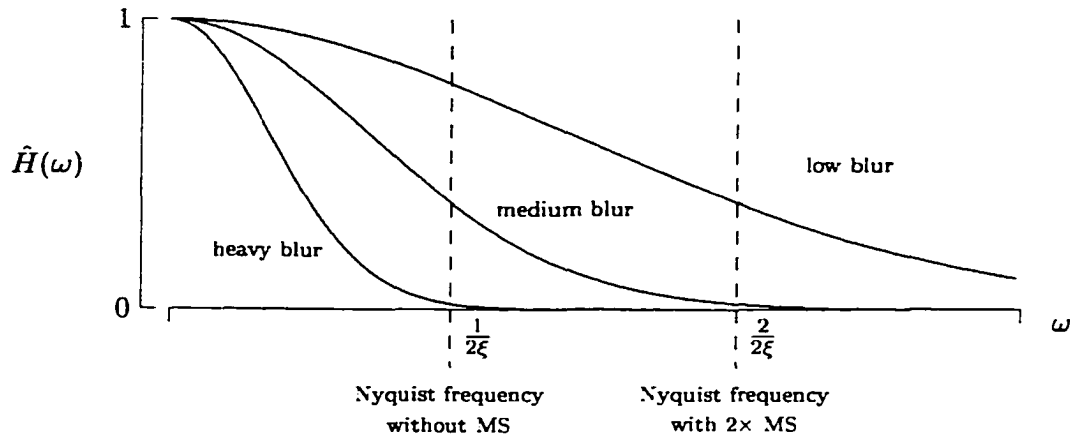


Figure 6.3: Various OTFs and Nyquist Frequency

The first dashed vertical line shows the Nyquist frequency of the acquisition device. The second dashed line shows where the Nyquist frequency would be pushed out to when a $2\times$ microscan is used. Microscanning with respect to the “heavy-blur” optical transfer function would not produce any improvement in the reconstructed image. Any high-frequency features that exist in the input scene are already lost due to the blurring effects of the formation filter. If the optical transfer function resembles the “medium-blur” line with respect to the Nyquist frequency, then a $2\times$ microscan would improve the resolution of reconstructed images that contain energy at frequencies beyond the Nyquist frequency. Any additional microscanning performed with respect to this “medium-blur” OTF would not improve the reconstructed image. A $3\times$ or $4\times$ microscan could be used to restore energy at frequencies beyond the $2\times$ Nyquist frequency in the “low-blur” case. This chapter represents an investigation into aliasing effects, therefore, for the results presented in this chapter, we will assume that the OTF of the acquisition device is an “All Pass” filter. That is, $\hat{H}(\omega_1, \omega_2) = 1 \quad \forall(\omega_1, \omega_2)$. We have already discussed (Chapter 3) that the CLS restoration filter is designed to account for the degradations that occur with respect to the

formation filter h and the reconstruction filter d . Therefore, in this chapter we will focus on the remaining degradation inherent in the model defined in Chapter 2: aliasing.

6.3.2 Effects of Aliasing

In addition to increasing the effective Nyquist frequency in a mathematical sense, in a visual sense microscanning suppresses aliased noise. Aliasing occurs when energy at frequencies beyond the Nyquist frequency are “folded” back onto lower frequencies. This causes aliased noise features to appear in the reconstructed image. For illustration, we will use the input scene shown in Figure 6.4.

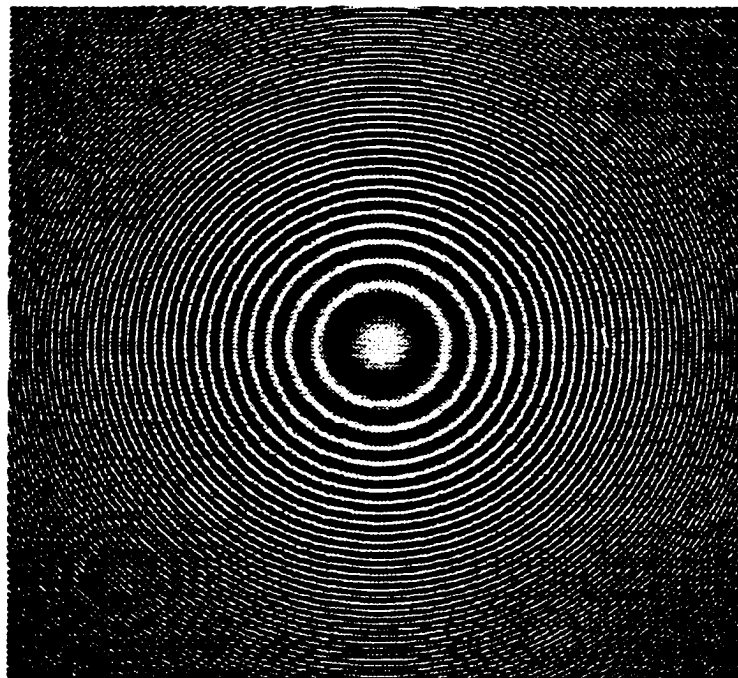


Figure 6.4: $s(x_1, x_2)$

This scene is defined in the spatial domain by the following relationship:

$$s(x_1, x_2) = 100 \left(1 + \cos \left(\frac{\pi(x_1 - P_1/2)^2}{2P_1} + \frac{\pi(x_2 - P_2/2)^2}{2P_2} \right) \right)$$

for $0 \leq x_1 \leq P_1$ and $0 \leq x_2 \leq P_2$ where $P_1 = P_2 = 512$. The input scene is sampled onto a $N_1 \times N_2 = 128 \times 128$ sampling grid. Thus, the Nyquist frequency for this (simulated) acquisition device is

$$\omega_{Nyq} = \frac{1}{2\xi} = \frac{N}{2P} = \frac{128}{2(512)} = 0.125.$$

Figure 6.5 shows the frequency response $|\hat{S}(\omega_1, \omega_2)|$ where $\omega_1 = \omega_2$. The leftmost dashed line indicates the Nyquist frequency of the acquisition device. It is obvious that a significant amount of energy is present beyond the Nyquist frequency. This energy will be folded back onto the energy within the Nyquist frequency creating aliasing artifacts in the restored and reconstructed images.

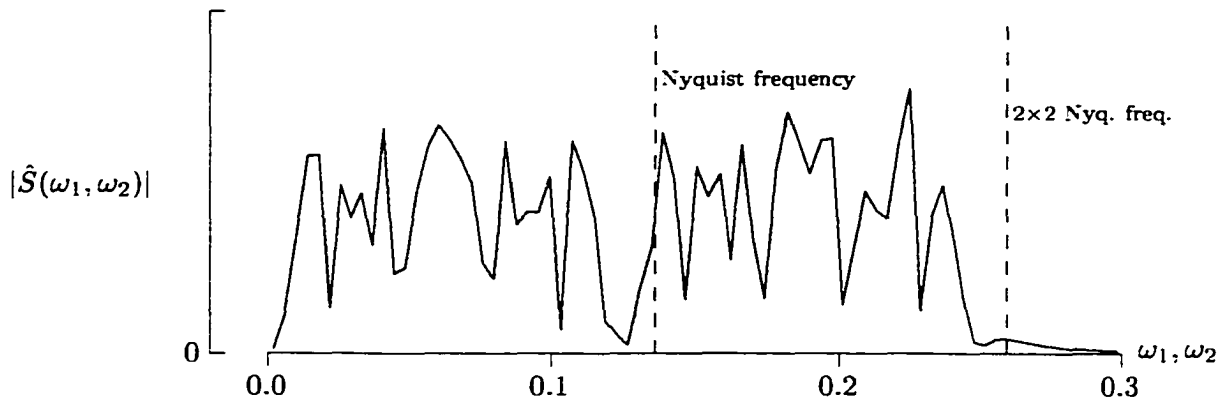


Figure 6.5: Single Scan Nyquist Frequency and 2×2 Microscan Nyquist Frequency

This sampling density is insufficient to accurately sample the higher frequencies found in the image. The energy at frequencies outside the passband form (subbands) fold back into the passband and cause aliasing in the sampled image[8]. The effects of aliasing can already be seen in the sampled image in Figure 6.6. The presence of the additional circle artifacts in the acquired digital image result from the aliasing that occurred due to insufficient sampling.

No restoration filter, CLS or otherwise, can restore the higher frequencies that have been folded back onto the lower frequencies. A parametric cubic convolution reconstruction filter used on the sampled image produces the continuous output image shown in Figure 6.7.

Microscanning at a 2×2 density doubles the effective Nyquist frequency (also shown in Figure 6.5). Only a small amount of energy exists at frequencies greater than the effective microscan Nyquist frequency. Thus, the PCC reconstructed image in Figure 6.8 shows that the increased sampling density suppresses the aliased noise.

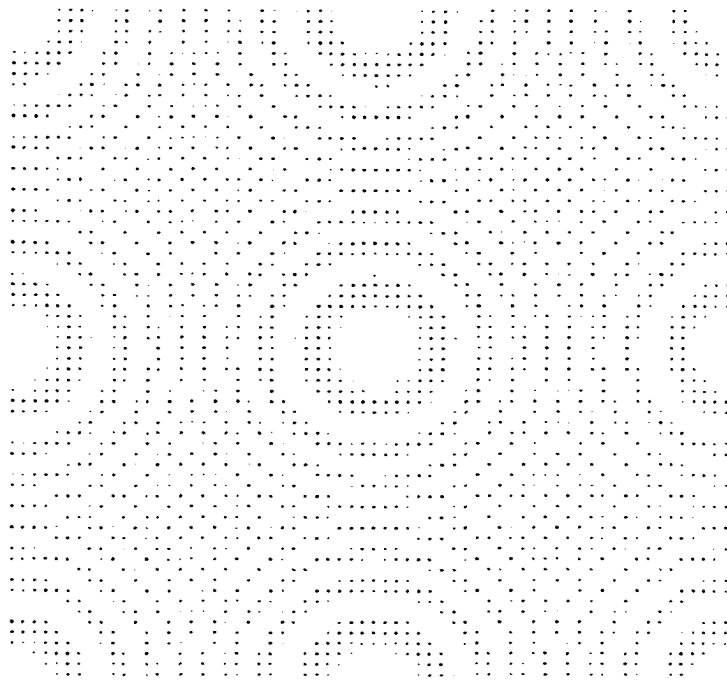


Figure 6.6: $p[n_1, n_2]$, 128×128 Sample

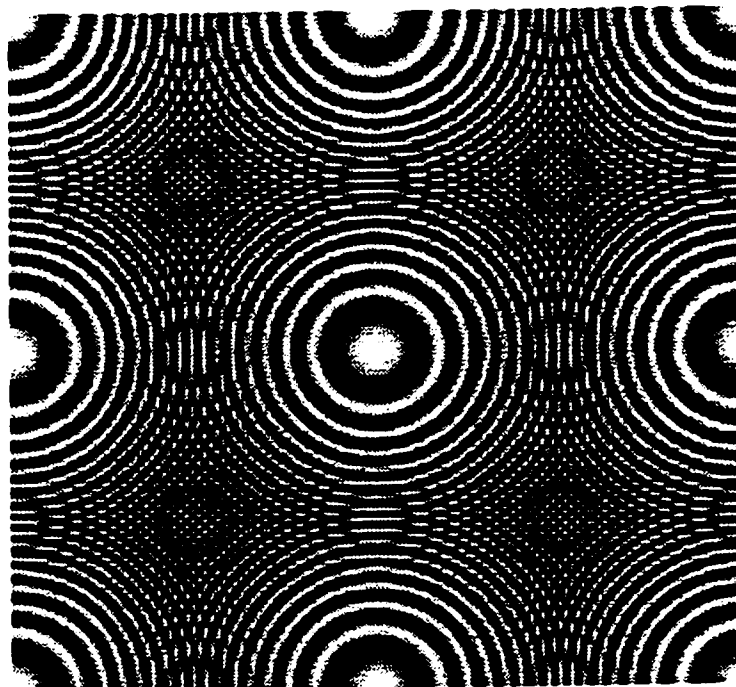


Figure 6.7: $r(x_1, x_2), M = 1$

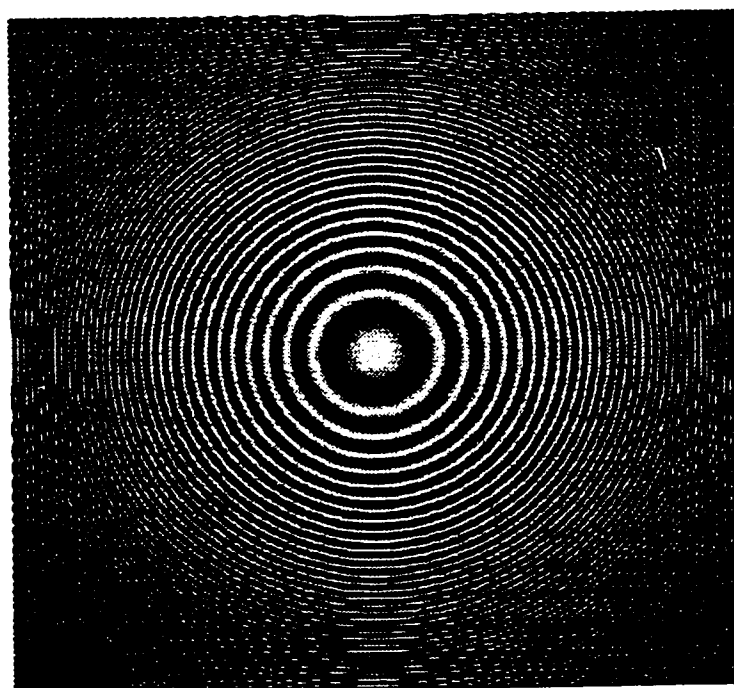


Figure 6.8: $r(x_1, x_2), M = 2$

6.4 1-Dimensional Microscan Example

The purpose of microscanning is to acquire features of an image (or signal) which occur at higher frequencies than the Nyquist frequency of the acquisition device, thus improving the resolution of the composite image relative to the individually scanned images[63]. These high-frequency features can then be restored and reconstructed. Suppose an input had the characteristics shown in Figure 6.9. The frequency domain representation of this input scene has energy only at the frequency $\omega = 4/512$. Thus, in order for a sample to accurately acquire the signal, the Nyquist frequency must be at least $4/512$.

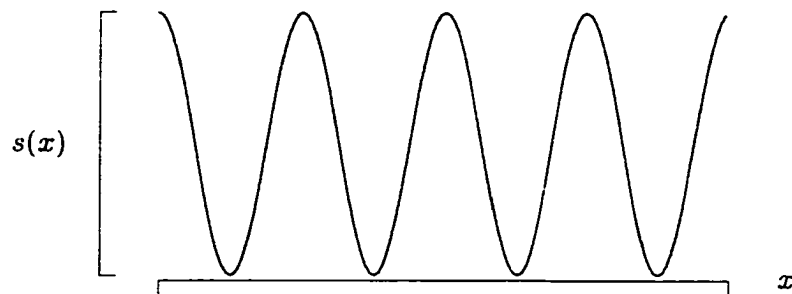


Figure 6.9: Sample Input Signal

Suppose the acquisition device had the unfortunate characteristic of a sampling density of $N = 4$. The Nyquist frequency is

$$\omega_{Nyq} = \frac{1}{2\xi} = \frac{N}{2P} = \frac{4}{2(512)} = 0.0039$$

and, thus, the energy present at $\omega = 4/512 = 0.0078$ would be folded back onto frequencies below the Nyquist frequency. In the spatial domain, this can be seen by the samples occurring every time the signal is at its highest intensity (shown in Figure 6.10 by the •'s). The reconstructed signal, corrupted by aliased noise, would consist of the sampled intensity value only.

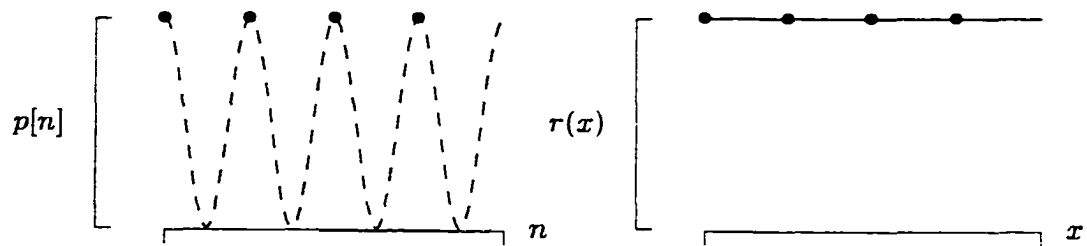
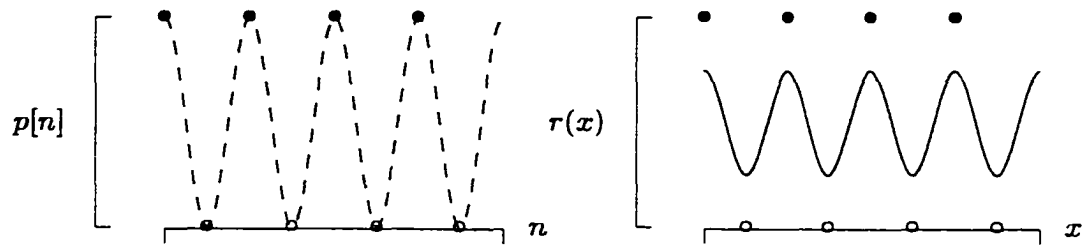


Figure 6.10: Single Scan Sample and Reconstruction

To compensate for this problem, a $2\times$ microscan could be used. A $2\times$ microscan would increase the Nyquist frequency to $\omega_{Nyq} = 8/2(512) = 0.0078$. Thus, no energy exists beyond the effective Nyquist frequency and aliasing would be suppressed. In this example, the first scan would remain the same. By using a second scan which occurs at a half intersample distance shift from the first scan (indicated by the \circ 's), the signal can be reconstructed, as illustrated in Figure 6.11.

Figure 6.11: $2\times$ Microscan Samples and Reconstruction

As shown in Figure 6.11, the original signal is not reconstructed exactly. This is due to the blurring effects of the PCC reconstruction filter d (similar to the effects of the blurring incurred due to the formation filter h). If a restoration filter, such as the CLS restoration filter, had been used to process the sampled signal the reconstructed signal would more closely resemble the input signal (indicated in Figure 6.12)

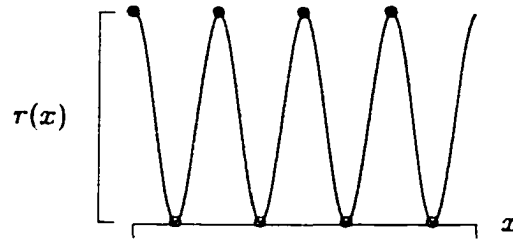


Figure 6.12: $2\times$ Microscan Samples, Restoration and Reconstruction

6.5 Complete vs. Partial Reconstitution of Input Scene

We have shown that there are three main degradations which affect the processing of a digital image (in addition to random noise effects). Those degradations consist of the blurring effects of the OTF, the blurring effects of the reconstruction filter, and the aliased noise artifacts generated by undersampling. In this section we show that only by compensating for all of these degradations will the output image most closely resemble the input scene (based on the root mean square error metric).

Suppose the input scene is the aerial image shown in Figure 6.13. The OTF blurred image, where the OTF is defined as

$$\hat{H}(\omega_1, \omega_2) = \exp(-(\omega_1/0.4)^2) \exp(-\omega_2/0.4)^2),$$

is shown in Figure 6.14.

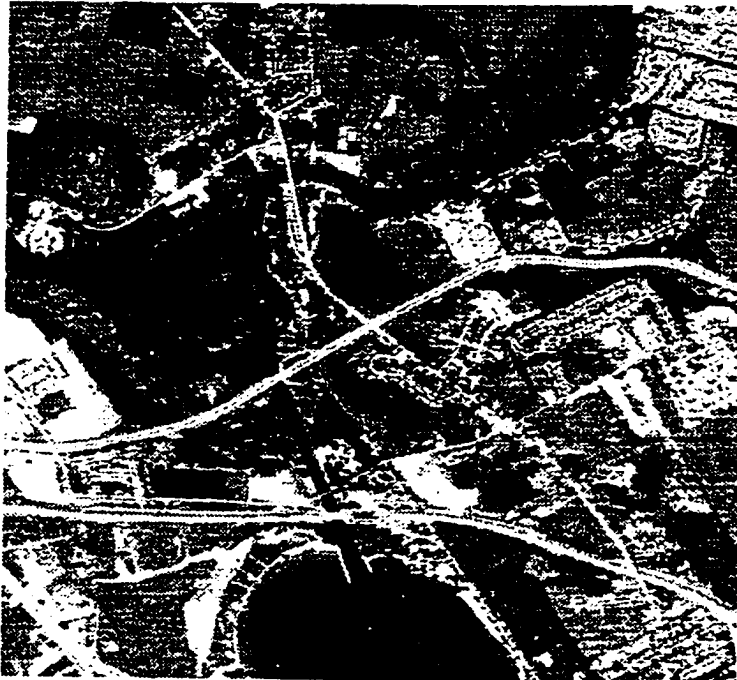


Figure 6.13: $s(x_1, x_2)$



Figure 6.14: $g(x_1, x_2)$

The frequency response of this input scene where $\omega_1 = \omega_2$ is shown in Figure 6.15. The Nyquist frequency is depicted based on a sampling density of $N_1 \times N_2 = 128 \times 128$. The effective Nyquist frequency of a 2×2 microscan is also shown.

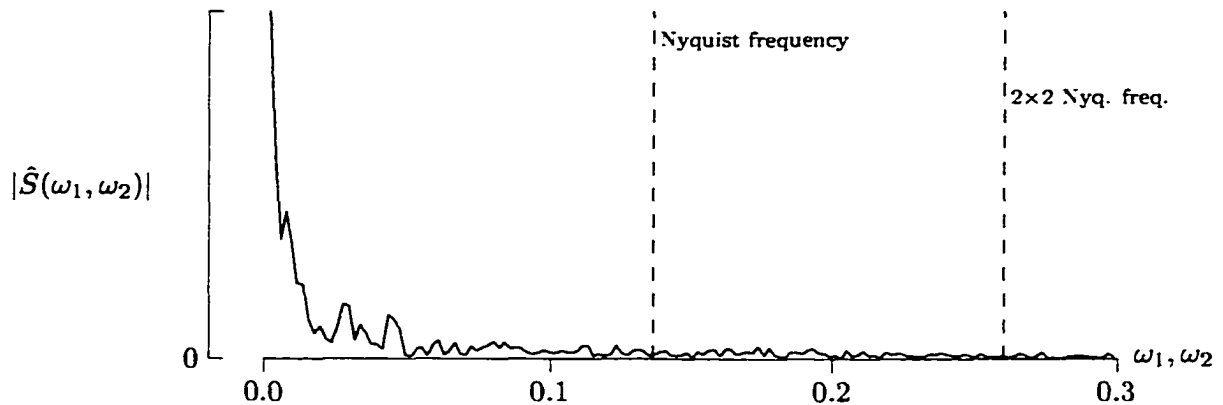


Figure 6.15: Frequency response $|\hat{S}|$ of Aerial image with $\omega_1 = \omega_2$

Figure 6.16 shows the sampled image p where $N_1 \times N_2 = 128 \times 128$.



Figure 6.16: $p[n_1, n_2], N = 128$

PCC reconstruction of Figure 6.16 without any microscanning or restoration is shown in Figure 6.17. This reconstruction fails to account for either the blurring effects of the OTF and RTF or the aliased noise present due to undersampling.

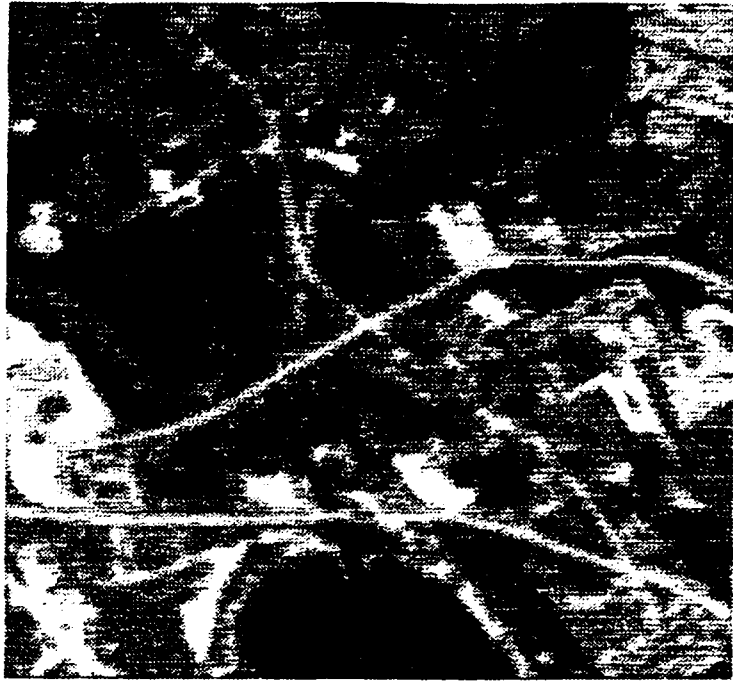


Figure 6.17: $r(x_1, x_2)$, No Restoration, No Microscanning

Using the CLS restoration filter on Figure 6.16 and reconstructing the image using the PCC reconstruction filter produces Figure 6.18. This reconstruction fails to account for the aliased noise present due to undersampling. Figure 6.19 shows a 2×2 microscan of the input scene.



Figure 6.18: $r(x_1, x_2)$, CLS Restoration, No Microscanning



Figure 6.19: $p[n_1, n_2]$, 2×2 Microscan

PCC reconstruction of Figure 6.19 without restoration is shown in Figure 6.20. This reconstruction fails to account for the blurring produced by the OTF and RTF.

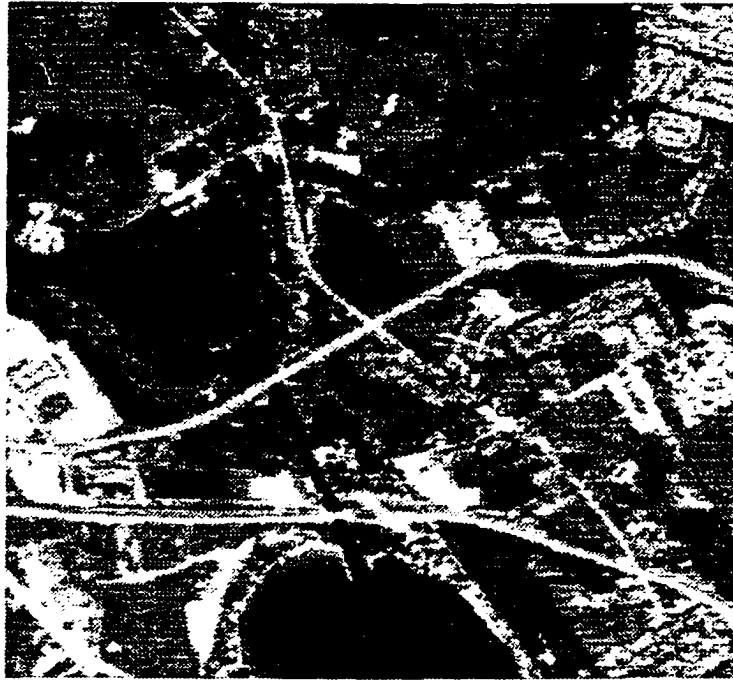


Figure 6.20: $r(x_1, x_2)$, No Restoration, 2×2 Microscanning

Figure 6.21 represents a complete reconstitution of the input scene using the CLS restoration filter and a 2×2 microscan. The effects of the blurring from the OTF and RTF are suppressed by the CLS restoration filter. The effects of aliasing are suppressed by the increased sampling density.



Figure 6.21: $r(x_1, x_2)$, CLS Restoration, 2×2 Microscanning

Visual confirmation of the increased spatial resolution of the reconstructed images processed by microscanning and restoration is important, however, a closer look at the root mean square error between the various output images r and the input scene s quantifies these improvements.

	RMSE relative to s
g	15.672498
r , No Restoration, No Microscan	16.343288
r , CLS Restoration, No Microscan	14.506453
r , No Restoration, 2×2 Microscan	12.029806
r , CLS Restoration, 2×2 Microscan	9.172435

6.6 Real Digital Image Processing

In this and previous chapters we have presented theory designed and implemented to restore digital images subject to a specific system model. All results presented have been based on an exact knowledge of the input scene s and the OTF h of a simulated acquisition device. This is not a realistic expectation for real digital image processing, however. In the next chapter we use the theory presented here on real acquired digital images.

Chapter 7

Real Data Application and Conclusions

It can be argued based on the quantitative (RMSE) and qualitative (visual) results presented in the previous chapters that the theory presented in this dissertation is sound. For the theory to be proven, however, investigation of its application to real acquired digital images is required. In this chapter we present the processing of real digital image data acquired using the digital camera feature of a Compact VHS JVC Camcorder containing a staring array of 640×480 pixels.

Because there are no means to quantify the actual input scene in terms of its Fourier coefficients (as presented in Chapter 2) there does not exist a method by which a quantitative error metric can be defined. Thus, the results presented in this chapter will be based entirely on a qualitative analysis.

As discussed in Chapter 4, the first step in constructing the Constrained Least Squares

filter to restore the acquired images is to calculate the optical transfer function (OTF) of the acquisition device. The next section presents the estimation of the OTF for the acquisition device used.

7.1 OTF Estimation

The acquisition device used to generate the digital images presented in this chapter contained 640×480 physical detectors that sampled the scene. To effectively estimate the OTF of the acquisition device, as described in Chapter 4, a knife-edge image must be acquired such that the edge lies at a slight angle to the direction of the sampling grid. The angle of the inclined knife-edge should be created such that the maximum number of sub-pixel edge locations (described in Section 4.1) within the edge profiles is achieved. The square image shown in Figure 7.1 was used as the knife-edge image. The image was acquired by placing a white, square piece of paper on a black background.

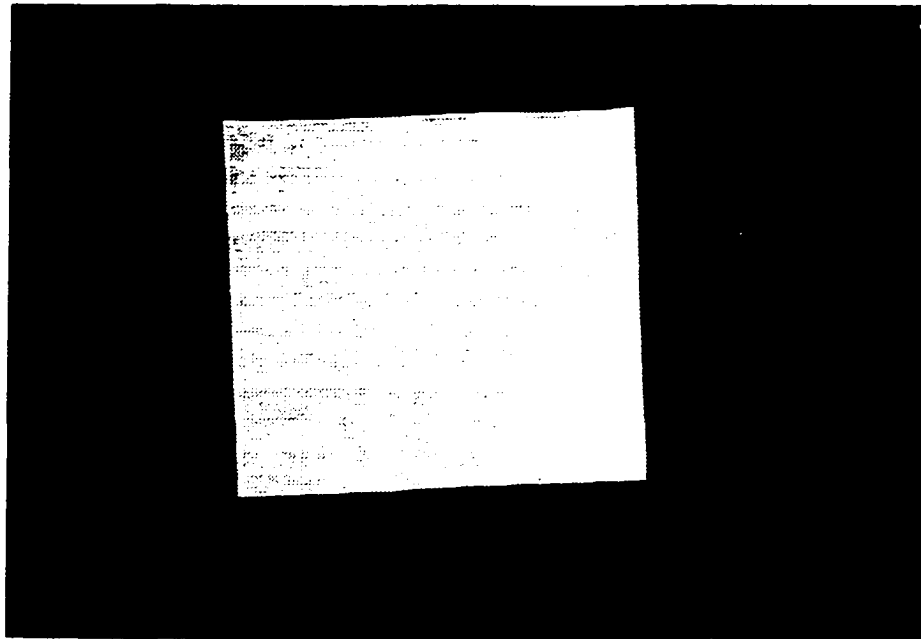


Figure 7.1: Knife-Edge Image

Next, approximately 100 edge profiles were extracted from the left edge of the square. Based on the results presented in Chapter 5, which comprised a comprehensive investigation of various edge location algorithms taken from current literature, the edge location for each edge profile acquired from Figure 7.1 was estimated using the Linear Interpolation algorithm. The edge profiles were aligned relative to their estimated edge location and the edge profile data was binned based on $\rho = 4$ (or 4 bins per pixel width). The average of the binned data (within each bin) represents the super-resolution LSF and is depicted in Figure 7.2.

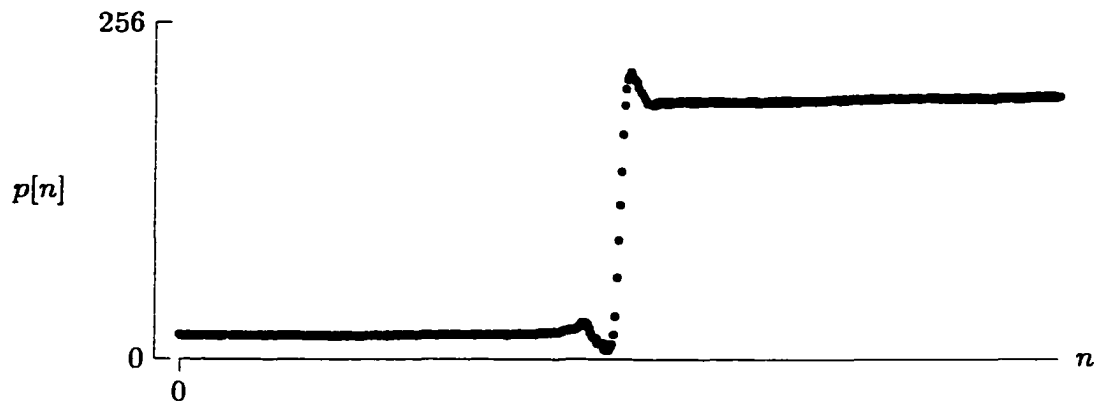


Figure 7.2: Super-Resolution LSF

For illustration, the super-resolution PSF, computed by $q[n] = p[n] - p[n - 1]$, is shown in Figure 7.3.

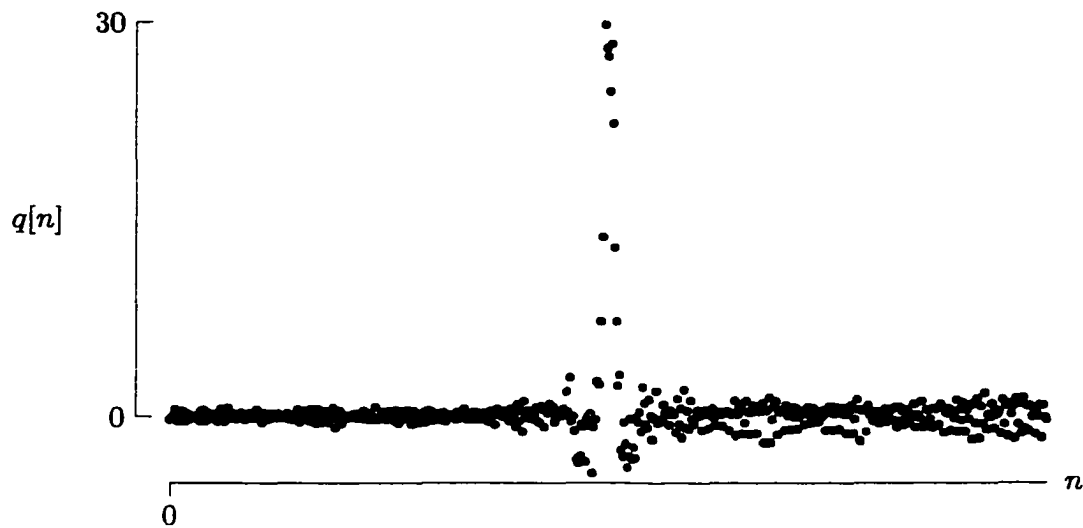


Figure 7.3: Super-Resolution PSF

Using the left and top edges of the square from Figure 7.1, the method described in Chapter 4 was utilized to calculate an estimation of the OTF of the acquisition device in the x_1 and x_2 directions. The results of those estimations can be seen in Figure 7.4.

Note that the estimated OTFs in both the x_1 and x_2 directions calculate $\hat{H}[0] = 1$. Typically the estimates would taper off to 0 as the frequency index ν increases. Due to

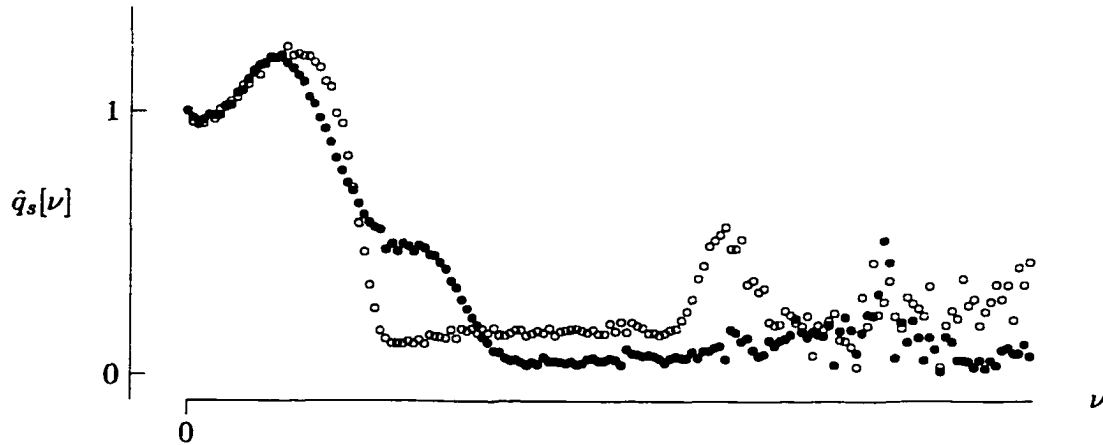


Figure 7.4: OTF Estimates, x_1 (o) and x_2 (•) directions

the under-shoot and over-shoot present in the super-resolution LSF, however, these OTF estimates contain energy increases at low frequency indexes before tapering off to 0. Thus, in order to fit a model function to the sampled OTF estimates, the following shifted Gaussian function containing one free parameter δ was used.

$$\hat{H}(\omega) = \exp(-(\omega - 0.03125)/\delta)^2$$

In order to most closely fit the function to the estimated data, the free parameter δ was defined to be 0.04 in both the x_1 and x_2 directions. Figure 7.5 depicts the OTF estimates along with an overlay of the model function used to represent \hat{H} .

Thus, the 2-dimensional OTF estimate for this acquisition device was calculated to be

$$\hat{H}(\omega_1, \omega_2) = \exp(-(\omega_1 - 0.03125)/\delta_1)^2 \exp(-(\omega_2 - 0.03125)/\delta_2)^2 \quad (7.1)$$

where $\delta_1 = \delta_2 = 0.04$.

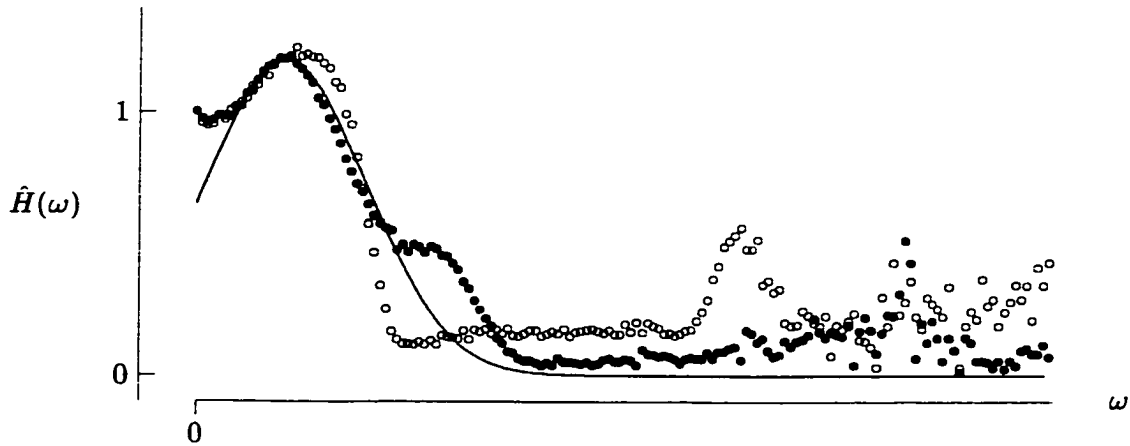


Figure 7.5: OTF Estimates, x_1 and x_2 directions with \hat{H} model function

7.2 Real Data System Model

It is important to note that for a real data investigation the system model is truncated. The system model defined in Chapter 2 (illustrated in Figure 7.6) represents a complete processing of the input scene to an output image.

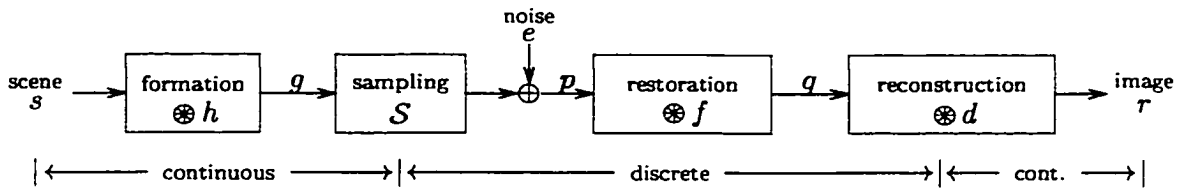


Figure 7.6: System Model

An acquisition device, however, captures the sampled, blurred digital image referred to in the system model as p . Therefore, the system model as defined by a real data environment begins with p as the input and processes the digital image as shown in Figure 7.7.

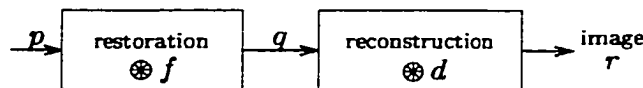


Figure 7.7: Real System Model

Figure 7.8 shows a 256×256 8-bit digital image acquired by the digital camera acquisition device described in the previous section.

Figure 7.8: $p_{0,0}$ - First Scan

Figure 7.9 and Figure 7.10 show the PCC reconstruction of the single scanned image (from Figure 7.8) with and without CLS restoration, respectively.



Figure 7.9: No Restoration, Single Scan



Figure 7.10: CLS Restoration, Single Scan

The CLS restoration filter used in Figure 7.10 uses the estimated OTF calculated in the previous section. Note that the restoration present in Figure 7.10 sharpens “line” features present in the image. These results, however, represent only a partial reconstitution of the input scene. Figure 7.11 and Figure 7.12 show the PCC reconstruction of a 2×2 microscan of the input scene with and without CLS restoration, respectively. Figure 7.11 illustrates the effectiveness of microscanning. Subpixel scale features, such as the white space present within the “R” in “E PLURIBUS”, are now accurately represented in the reconstructed image. Figure 7.12 represents a complete reconstitution of the input scene. Not only has microscanning the input scene produced subpixel scale features, but the implementation of the CLS filter has refined the image to eliminate the blur caused by the formation filter \hat{H} and the reconstruction filter \hat{D} thus sharpening the edges.



Figure 7.11: No Restoration, Microscan



Figure 7.12: CLS Restoration, Microscan

7.3 Conclusions

Based on the model presented in Chapter 2, we have shown that a complete reconstitution of an input scene can only be achieved through subpixel techniques. The constrained least squares restoration filter (defined in Chapter 3) represents a compromise restoration filter between an inverse filter, which may boost random noise inherent in the acquisition process, and the theoretically optimal Wiener filter, which relies on a priori knowledge of the scene Fourier spectrum. One free parameter within the CLS restoration filter allows for a fine tuning of fidelity versus smoothness in the restored image. Previous research using the CLS filter has relied upon the constraint of a unity intersample distance. This research has presented a derivation of the CLS filter that generalizes the intersample distance ξ and has shown how this change affects the rest of the system model. An important component of the CLS restoration filter is the optical transfer function (OTF) of the acquisition device. Therefore, an estimation technique was presented in Chapter 4 to accomplish this requirement.

An important step in determining the OTF of a particular acquisition device is to estimate edge locations of edge profiles acquired from a knife-edge image. We presented numerous algorithms in Chapter 5 as candidates for this estimation. Through a comprehensive investigation of various algorithms based on edge width α and signal-to-noise ratio SNR it was determined that although not always optimal, the Linear Interpolation algorithm was the most robust and accurate.

Image restoration is an important process for reconstructing an input scene; however, it was shown in Chapter 6 that no restoration filter (CLS or otherwise) can restore the

pixel-scale features that are lost due to aliasing. Complete reconstitution of the input scene relies upon a technique that increases the sampling density of the acquisition device. Microscanning has been presented as a solution to this problem. By acquiring multiple images of the scene with subpixel shift differences among the acquired images, a composition of the images could be performed to push the Nyquist frequency to higher frequencies, thus allowing for sub-pixel scale feature restoration.

The research presented here represents a novel investigation, through the use of sub-pixel techniques, to comprehensively identify and eliminate (to the extent possible) all of the degradations inherent in the system model defined. We have shown that, through the use of a root-mean-square-error (RMSE) metric, mathematically the results presented offer justification for the use of subpixel techniques to improve spatial resolution. More importantly, Chapter 7 presents qualitative results based on real acquired data.

Bibliography

- [1] I. E. ABDU. Image restoration of multiple frame sequences. In *Proceedings of SPIE Applications of Digital Image Processing XIX*, volume 2847, 1996.
- [2] K. J. BARNARD, E. A. WATSON, AND P. F. MCMANAMON. Nonmechanical microscanning using optical space-fed phased arrays. *Optical Engineering*, 33(9):3063–3071, September 1994.
- [3] M. BELGE, M. E. KILMER, AND E. L. MILLER. Wavelet domain image restoration with adaptive edge-preserving regularization. *IEEE Transactions on Image Processing*, 9(4):597–608, April 2000.
- [4] T. BERGER, J. O. STROMBERG, AND T. ELTOFT. Adaptive regularized constrained least squares image restoration. *IEEE Transactions on Image Processing*, 8(9):1191–1203, September 1999.
- [5] K. L. BOYER AND S. SARKAR. On the localization performance measure and optimal edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(1):106–110, January 1994.
- [6] J. S. CHEN AND G. MEDIONI. Detection, localization, and estimation of edges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(2):191–198, February 1989.
- [7] W. CHEN, M. CHEN, AND J. ZHOU. Adaptively regularized constrained total least-squares image restoration. *IEEE Transactions on Image Processing*, 9(4):588–596, April 2000.
- [8] Z. CHEN AND M. A. KARIM. Oversampled optical subband decomposition. *Optical Engineering*, 39(9):2422–2430, September 2000.
- [9] B. COHEN, V. AVRIN, M. BELITSKY, AND I. DINSTEIN. Generation of a restored image from a video sequence recorded under turbulence effects. *Optical Engineering*, 36(12):3312–3317, December 1997.
- [10] E. DEMICHELI, B. CAPRILE, P. OTTONELLO, AND V. TORRE. Localization and noise in edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(10):1106–1117, October 1989.

- [11] G. DEMOMENT. Image reconstruction and restoration: Overview of common estimation structures and problems. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(12):2024–2036, December 1989.
- [12] C. L. FALES, F. O. HUCK, R. ALTER-GARTENBERG, AND Z. RAHMAN. Image gathering and digital restoration. *Philosophical Transaction of the Royal Society of London Series A*, 354:2249–2287, 1996.
- [13] A. FRIEDENBERG. Microscan in infrared staring systems. *Optical Engineering*, 36(6):1745–1749, June 1997.
- [14] N. P. GALATSANOS AND R. T. CHIN. Digital restoration of multichannel images. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(3):415–421, March 1989.
- [15] N. P. GALATSANOS AND A. K. KATSAGGELOS. Methods for choosing the regularization parameter and estimating the noise variance in image restoration and their relation. *IEEE Transactions on Image Processing*, 1(3):322–336, July 1992.
- [16] J. C. GILLETTE, T. M. STADTMILLER, AND R. C. HARDIE. Aliasing reduction in staring infrared imagers utilizing subpixel techniques. *Optical Engineering*, 34(11):3130–3137, November 1995.
- [17] R. C. HARDIE, K. J. BARNARD, J. G. BOGNAR, E. E. ARMSTRONG, AND E. A. WATSON. High-resolution image reconstruction from a sequence of rotated and translated frames and its application to an infrared imaging system. *Optical Engineering*, 37(1):247–260, January 1998.
- [18] R. HAZRA. Constrained least-squares digital image restoration. *The College of William and Mary, Ph.D. Dissertation*, 1995.
- [19] J. HU, M. SONG, Y. SUN, AND Y. LI. Measurement of modulation transfer function of charge-coupled devices using frequency-variable sine grating pattern. *Optical Engineering*, 38(7):1200–1204, July 1999.
- [20] F. O. HUCK, C. L. FALES, R. ALTER-GARTENBERG, S. K. PARK, AND Z. RAHMAN. Information-theoretic assessment of sampled imaging systems. *Optical Engineering*, 38(5):742–762, May 1999.
- [21] F. O. HUCK, C. L. FALES, J. A. MCCORMICK, AND S. K. PARK. Image-gathering system design for information and fidelity. *Journal of the Optical Society of America A*, 5:285–299, 1988.
- [22] F. O. HUCK, C. L. FALES, AND Z. RAHMAN. An information theory of visual communication. *Philosophical Transactions of the Royal Society of London Series A*, 354:2193–2248, 1996.
- [23] F. O. HUCK, C. L. FALES, AND Z. RAHMAN. *Visual Communication*. 1997.

- [24] G. JACQUEMOD, C. ODET, AND R. GOUTTE. Image resolution enhancement using subpixel camera displacement. *Signal Processing*, pages 139–146, 1992.
- [25] B. D. JEFFS AND M. GUNSAY. Restoration of blurred star field images by maximally sparse optimization. *IEEE Transactions on Image Processing*, 2(2):202–211, April 1993.
- [26] R. KAKARALA AND A. O. HERO. On achievable accuracy in edge localization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(7):777–781, July 1992.
- [27] C. KAO, X. PAN, P. LA RIVIERE, AND M. A. ANASTASIO. Fourier-based optimal recovery method for antialiasing interpolation. *Optical Engineering*, 38(12):2041–2044, December 1999.
- [28] A. K. KATSAGGELOS. Iterative image restoration algorithms. *Optical Engineering*, 28(7):735–748, July 1989.
- [29] A. K. KATSAGGELOS, J. BIEMOND, R. W. SCHAFER, AND R. M. MERSEREAU. A regularized iterative image restoration algorithm. *IEEE Transactions on Signal Processing*, 39(4):914–929, April 1991.
- [30] S. P. KIM AND WEN-YU SU. Recursive high-resolution reconstruction of blurred multiframe images. *IEEE Transactions on Image Processing*, 2(4):534–539, October 1993.
- [31] M. KISWORO, S. BENKATESH, AND G. WEST. Modeling edges at subpixel accuracy using the local energy approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(4):405–410, April 1994.
- [32] S. KUO AND R. J. MAMMONE. Image restoration by convex projections using adaptive constraints and the l_1 norm. *IEEE Transactions on Signal Processing*, 40(1):159–168, January 1992.
- [33] R. L. LAGENDIJK, J. BIEMOND, AND D. E. BOEKKEE. Regularized iterative image restoration with ringing reduction. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 36(12):1874–1888, December 1988.
- [34] R. L. LAGENDIJK, J. BIEMOND, AND D. E. BOEKKEE. Identification and restoration of noisy blurred images using expectation-maximization algorithm. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 38(7):1180–1191, July 1990.
- [35] R. L. LAGENDIJK, A. M. TEKALP, AND J. BIEMOND. Maximum likelihood image and blur identification: a unifying approach. *Optical Engineering*, 29(5):422–435, May 1990.
- [36] K. T. LAY AND A. K. KATSAGGELOS. Image identification and restoration based on the expectation-maximization algorithm. *Optical Engineering*, 29(5):436–445, May 1990.

- [37] H. C. LEE. Review of image-blur models in a photographic system using the principles of optics. *Optical Engineering*, 29(5):406–421, May 1990.
- [38] C. H. LI AND P. K. S. TAM. Probabilistic nonlinear diffusion image filter. *Optical Engineering*, 39(9):2381–2386, September 2000.
- [39] M. K. OZKAN, A. T. ERDEM, M. I. SEZAN, AND A. M. TEKALP. Efficient multiframe wiener restoration of blurred and noisy image sequences. *IEEE Transactions on Image Processing*, 1(4):453–476, October 1992.
- [40] D. J. PARK, K. M. NAM, AND R. PARK. Multiresolution edge detection techniques. *Pattern Recognition*, 28(2):211–229, 1995.
- [41] S. C. PARK AND M. G. KANG. Noise-adaptive edge-preserving image restoration algorithm. *Optical Engineering*, 39(12):3124–3137, December 2000.
- [42] S. K. PARK. Flir range enhancement by constrained least-squares restoration. *NVL Final Report*, 1995.
- [43] S. K. PARK AND M. R. IDEMA. Algorithms for sub-pixel edge reconstruction. *SPIE Proceedings*, April 1996.
- [44] S. K. PARK AND M. R. IDEMA. Pixel-scale feature restoration from microscanned image data. *SPIE Proceedings*, April 1997.
- [45] S. K. PARK AND R. A. SCHOWENGERDT. Image reconstruction by parametric cubic convolution. *Computer Vision, Graphics, and Image Processing*, 23:258–272, 1983.
- [46] STEPHEN K. PARK AND ZIA UR RAHMAN. Fidelity analysis of sampled imaging systems. *Optical Engineering*, 38(5):786–800, May 1999.
- [47] Y. PARK AND M. SOUMEKH. Reconstruction from unevenly spaced sampled data using the iterative methods. *IEEE Transactions on Signal Processing*, 43(1):303–308, January 1995.
- [48] T. PELI AND D. MALAH. A study of edge detection algorithms. *Computer Graphics and Image Processing*, 20:1–21, 1982.
- [49] S. J. REEVES AND R. M. MERSEREAU. Blur identification by the method of generalized cross-validation. *IEEE Transactions on Image Processing*, 1(3):301–311, July 1992.
- [50] S. REICHENBACH. Small-kernel image restoration. *The College of William and Mary, Ph.D. Dissertation*, 1989.
- [51] S. E. REICHENBACH, S. K. PARK, AND R. NARAYANSWAMY. Characterizing digital image acquisition devices. *Optical Engineering*, 30(2):170–177, February 1991.
- [52] S. RHEE AND M. G. KANG. Discrete cosine transform based regularized high-resolution image reconstruction algorithm. *Optical Engineering*, 38(8):1348–1356, August 1999.

- [53] S. SAMADDAR AND R. J. MAMMONE. Image restoration using a row action projection method with adaptive smoothing. *Optical Engineering*, 34(4):1132–1147, April 1995.
- [54] A. E. SAVAKIS AND H. J. TRUSSELL. Blur identification by residual spectral matching. *IEEE Transactions on Image Processing*, 2(2):141–151, April 1993.
- [55] A. E. SAVAKIS AND H. J. TRUSSELL. On the accuracy of psf representation in image restoration. *IEEE Transactions on Image Processing*, 2(2):252–259, April 1993.
- [56] P. SEITZ. Optical superresolution using solid-state cameras and digital signal processing. *Optical Engineering*, 27(7):535–540, July 1988.
- [57] M. I. SEZAN AND A. M. TEKALP. Survey of recent developments in digital image restoration. *Optical Engineering*, 29(5):393–405, May 1990.
- [58] D. L. SNYDER, A. B. HAMMOUD, AND R. L. WHITE. Image recovery from data acquired with a charge-coupled-device camera. *Optical Society of America*, 10(5):1014–1023, May 1993.
- [59] D. L. SNYDER, T. J. SCHULZ, AND J. A. O’SULLIVAN. Deblurring subject to non-negativity constraints. *IEEE Transactions on Signal Processing*, 40(5):1143–1150, May 1992.
- [60] A. J. TABATABAI AND O. R. MITCHELL. Edge location to subpixel values in digital imagery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(2):188–201, March 1984.
- [61] H. D. TAGARE AND R. J. P. DEFIGUEIREDO. On the localization performance measure and optimal edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(12):1186–1190, December 1990.
- [62] A. P. TZANNES AND J. M. MOONEY. Measurement of the modulation transfer function of infrared cameras. *Optical Engineering*, 34(6):1808–1817, June 1995.
- [63] H. UR AND D. GROSS. Improved resolution from subpixel shifted pictures. *Computer Vision, Graphics, and Image Processing*, 54(3):181–186, March 1992.
- [64] Z. WANG AND D. ZHANG. Novel evolutionary method for gray-level image restoration. *Optical Engineering*, 38(4):626–629, April 1999.
- [65] R. G. WHITE AND R. A. SCHOWENGERDT. Effect of point-spread functions on precision edge measurement. *Optical Society of America*, 11(10):2593–2603, October 1994.
- [66] O. YADID-PECHT. Geometrical modulation transfer function for different pixel active area shapes. *Optical Engineering*, 39(4):859–865, April 2000.
- [67] Y. YITZHAKY, G. BOSHUSHA, Y. LEVY, AND N. S. KOPEIKA. Restoration of an image degraded by vibrations using only a single frame. *Optical Engineering*, 39(8):2083–2091, August 2000.
- [68] D. ZIOU AND S. TABBONE. A multi-scale edge detector. *Pattern Recognition*, 26(9):1305–1314, 1993.

VITA

Mark Robert Idema

Born in Grand Rapids, Michigan, February 18, 1971. Entered Pepperdine University in 1989. Became a member of the Alpha Chi and Golden Key Honor Societies. Graduated cum laude from Pepperdine University with a Bachelor of Science degree in Mathematics/Computer Science in 1992. Entered the College of William & Mary with a teaching assistantship in 1993. Switched to a research assistantship under the guidance of Dr. Stephen Park and published two papers with Dr. Park. Earned a Master of Science degree from the College of William & Mary in Computer Science in 1995.