Autofocusing and Astigmatism Correction in the Scanning Electron Microscope

UNIVERSITY OF CAMBRIDGE

*A dissertation presented to the Faculty of the
Department of Engineering, University of Cambridge
in partial fulfillment of the requirements for the
degree Master of Philosophy in Electrical Engineering*

Submitted by
Christopher F. Batten
August 2000

Declaration

I hereby declare that my thesis entitled *Autofocusing and Astigmatism Correction in the Scanning Electron Microscope* is not substantially the same as any that I have submitted for a degree or diploma or other qualification at any other University. I further state that no part of my thesis has already been or is being concurrently submitted for any such degree, diploma, or other qualification.

I hereby declare that my thesis does not exceed the limit of length prescribed in the Special Regulations of the M.Phil. examination for which I am a candidate. The length of my thesis is approximately 14,800 words.

Signed

Christopher F. Batten

Acknowledgements

Abstract

This work investigates both the theoretical and practical aspects of autofocusing and astigmatism correction in the scanning electron microscope. A general framework is used to divide the problem into four primary areas of concern: noise reduction, regions of interest, sharpness measures, and maximum sharpness search algorithms. Each of these areas is investigated in detail and several novel concepts are introduced including: the use of reduced domain median filters to mitigate limited bandwidth distortion, a method for confining the region of interest to specimen features, more sophisticated maximum sharpness search algorithms such as the variable stepsize search and the Fibonacci search, and interpolation based on a model of variance as a function of defocus.

A development testbed was established which allowed for rapid prototyping in MATLAB, implementation in Visual C++, and then final packaging as an ActiveX control. The theoretical work was implemented in a modular component (called SEMimage) for use with the Leo 440 SEM located in the Scientific Imaging Group at the Cambridge University Engineering Department. The framework was used to develop algorithms for full autofocusing, fine autofocusing, real-time autofocusing, and astigmatism correction. Real-time autofocusing enables the software to automatically determine when an image has become defocused and to then take appropriate action to move the image back into focus. Real-time beam alignment and signal to noise calculation were also implemented.

SEMimage was tested on the instrument and provided an effective means for automated focusing and astigmatism correction. The fine autofocusing provided by SEMimage is relatively fast and accurate, and the real-time autofocusing provides a unique method for keeping the instrument in focus.

Contents

List of Figures

Chapter 1

Introduction

Over the past fifty years, the scanning electron microscope (SEM) has grown from a specialized research device to a universal industrial tool. SEMs can be found on the IC fabrication assembly line, in aircraft engine maintenance hangars, and in small research laboratories. This wide array of applications means that SEM users are no longer solely seasoned experts, but also include technicians, mechanics, and professionals. There is a growing need for simpler SEM interfaces with greater automated control to effectively hide the device's complexity.

This is a key time for work on such automated SEM control, since recent advances in general purpose microprocessors have made complex software image processing a feasible alternative to traditional hardware and simpler software approaches. In an effort to take advantage of this opportunity, this thesis will investigate various autofocusing and astigmatism correction algorithms and then discuss a software prototype implemented for the LEO 440 SEM. This chapter will provide a general background on the scanning electron microscope and previous automated approaches, and will also introduce a framework for algorithm development.

1.1   The Scanning Electron Microscope

This section provides a brief overview of the scanning electron microscope and is intended to familiarize the reader with the basic structure and operation of an SEM. Oatley and Wells provide more comprehensive discussions in [17] and [24] respectively. An SEM has three main components: the optical column, the specimen chamber, and the appropriate electronics. These components are illustrated in Figure 1.1.

Section 1.1: The Scanning Electron Microscope
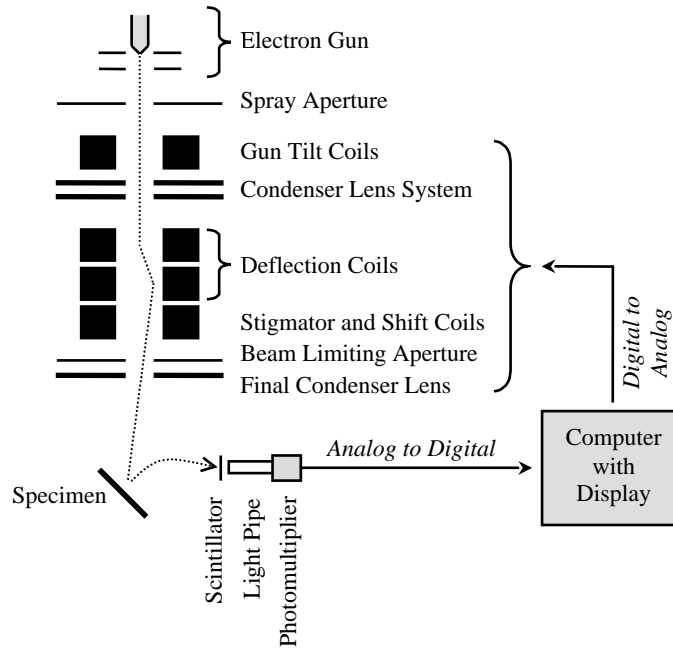
Figure 1.1: The Modern Scanning Electron Microscope

A focused electron beam is scanned over a specimen such that the interaction between the beam and the specimen excites various forms of radiation including backscattered electrons, secondary electrons, and x-rays. This radiation is then detected and analyzed to reveal information concerning the specimen's composition and topography. A surface SEM detects low energy secondary electrons (less than 50eV) by accelerating these electrons into a scintillator. The scintillator gives off visible light that is detected by a photomultiplier and converted into an amplified current signal. In traditional SEMs this signal is further amplified and used as the input to a cathode-ray tube (CRT). The specimen and the CRT are scanned synchronously, and magnification is achieved by varying the ratio between the CRT size and the beam scan area. In this way, the brightness of any point on the CRT is proportional to the number of secondary electrons emitted when the beam scans the corresponding point on the specimen. Modern SEMs digitize the current signal from the detector and use a digital computer to display and analyze the image. This computer is responsible for various other functions including control of the deflection coils and the focusing lenses. The work presented here is applicable to an SEM with such computer controlled scanning and image processing.

The electron gun should be a stable source of electrons, which can be achieved through thermal or field emission. Common emitter materials include tungsten and lanthanum hexaboride (LaB_6). This work assumes a thermal emission LaB_6

source, although the fundamental concepts are generally applicable to any scanning electron microscope. In addition to the electron gun, the optical column includes various electromagnetic lenses and the deflection coils necessary for scanning the electron beam over the specimen. The lenses are responsible for demagnifying the electron beam in order to minimize the size of the beam at the specimen (spot size). Both two and three stage lens systems are possible, with the deflection coils usually being positioned directly before the final lens. Stigmators are used to adjust the shape of the electron bean, while alignment and shift coils are used to slightly reposition the center of the beam.

The specimen chamber includes a stage for mounting the specimen and any required detectors. The working distance is defined as the distance between the final lens and the specimen. When using the secondary electron detector, the specimen is usually tilted to increase the number of electrons reaching the detector. As mentioned above, the scintillator/photomultiplier system is a common secondary electron detector system.

Current amplifiers and other analog electronics can be used to enhance the signal before it is digitized. The digitized signal is usually stored in a frame store, and most SEMs include dedicated hardware to manipulate and analyze the image. Software running on the computer is used for image processing, monitoring, and control. Most SEMs provide a graphical user interface that allows an operator to easily manipulate the instrument.

1.2   Overview of Previous Approaches

Autofocusing techniques usually fall into one of two categories: active methods and passive methods. Active methods use a separate subsystem to determine the distance between the final lens and the specimen. Passive methods determine defocus information solely from the final image.

Active methods are more common in the optical domain. Most modern automatic cameras use a range finding subsystem to accurately measure the optimal focal length. Unfortunately, the scale of such subsystems make them impractical for electron microscopy.

Passive autofocusing methods can be divided into two subcategories: image sharpness measures and predictive imaging models. The traditional approach is to apply an image sharpness measure over a range of focal lengths, and select the focal length with the maximum sharpness as the best focus [3, 5, 7, 21]. Further detail on previous passive sharpness measure approaches will be presented later in this work where necessary. As an alternative to the sharpness measure approach,

a general imaging model has been developed that allows defocus information to be determined from just a few images [16]. This model relates the point spread function of an image to the focal length.

Astigmatism information can best be determined through passive methods. Work has been done on determining astigmatism information from sharpness measures and also directly from the shape of the Fourier transform of the image. To use a sharpness measure, the image sharpness is determined for a range of stigmator values. Then the stigmator values providing the maximum sharpness are chosen as yielding the best beam shape [7]. Iterating between focusing and astigmatism correction is usually required with this approach. Work has been done recently on using the shape of the Fourier transform to determine the shape of the beam [18, 19]. The shape of the beam indicates which elements of the stigmator need to be adjusted. Autofocusing and astigmatism correction can usually be adjusted at the same time with this approach.

Although automated, previous autofocusing approaches rely on an operator to indicate when an adjustment is necessary. In contrast, many optical systems implement real-time focusing. These systems are able to determine when an image is out of focus and automatically initiate a real-time focusing operation. Most modern video cameras have real-time focusing.

This work further develops the traditional passive method of using sharpness measures for focusing and astigmatism correction. Improvements in selecting a region of interest and the use of more efficient maximum sharpness search algorithms are investigated. This work will also investigate the potential for real-time focusing in scanning electron microscopy. A development testbed for the LEO 440 SEM located in the Scientific Imaging Group at the Cambridge University Engineering Department has been established to aid in the practical implementation of this theoretical work. A modular component based prototype has been developed that provides full focusing, fine focusing, real-time focusing, astigmatism correction, beam alignment, and signal to noise calculation.

1.3   Framework for Algorithm Development

The development of an effective and robust autofocusing or astigmatism correction algorithm can be considerably simplified if the problem is separated into different areas of concern. One can then investigate each area separately, before making design decisions in one area that will influence choices in other areas. This work separates the problem into the following four areas: noise reduction, regions of interest, sharpness measures, and maximum sharpness search algorithms.

These areas of concern are interrelated, and a collective set of choices should target the overall objective of the algorithm. To illustrate, consider two examples: developing a coarse astigmatism correction algorithm and developing a real-time fine focusing algorithm.

A coarse astigmatism correction algorithm does not require a high degree of accuracy, should be relatively fast, and should not make any assumptions on the current state of the stigmator. This means the developer may decide to tolerate a poor signal to noise ratio in an effort to reduce the number of frames needed for each image. Consequently, a more robust sharpness measure will be needed that can hopefully handle the additional noise. The developer may decide to use a reduced region of interest to further increase the speed of the algorithm. These design decisions mean that a simple fixed stepsize search will probably be used when determining the maximum sharpness.

A real-time fine focusing algorithm has a very different purpose than the previous example. Fine focusing requires a high degree of accuracy and can assume that the initial focal length is relatively near the best focus. Real-time algorithms must be particularly fast and the transition from a defocused to focused image should be relatively smooth. These constraints mean that the developer will probably make very different design decisions. The developer may choose an average amount of noise reduction to balance speed and effectiveness. A search algorithm that only moves in one direction may be used to guarantee a smooth transition, and thus a sharpness measure will be needed which steadily changes over a wide range of focal lengths.

The details of these design decisions will be discussed in the following chapters. The key concept is that once one understands each of these four areas of concern, one will have a fundamental framework that can then be used to meet the objectives of a specific autofocusing or astigmatism correction algorithm.

1.4   Outline of Thesis

The next four chapters (Chapter 2 to Chapter 5) will discuss each of the areas of concern that form the framework for algorithm development: noise reduction, regions of interest, sharpness measures, and maximum sharpness search algorithms. These chapters concentrate on the application of each area to focusing. Chapter 6 will broaden this work to astigmatism correction and also examine beam alignment as an additional real-time automated operation. Chapter 7 will discuss the development testbed and the prototype written for the LEO 440 SEM. Finally, general conclusions and suggestions for future work will be presented.

Section 1.4: Outline of Thesis

Chapter 2

Noise Reduction

The stochastic nature of electron microscopy means that the signal to noise ratio (SNR) for unprocessed high resolution secondary electron images is often very low. Various forms of noise reduction can be used to increase this signal to noise ratio, but such improvements require additional processing time. This tradeoff between noise reduction and processing time can significantly impact other design decisions when forming autofocusing algorithms.

This chapter discusses the different types of noise in scanning electron microscopy and also provides a method for quantitatively measuring noise in an image. Various noise reduction methods are then discussed including frame averaging, pixel averaging, and median filtering. Each of these methods are examined within the context of their impact on the signal to noise ratio and overall processing time.

2.1   Types of Noise

A basic understanding of the origin of noise in SEM images can aid in developing effective noise reduction techniques. There are two primary forms of noise in a SEM: noise due to the statistical nature of electron collision and emission, and noise due to the SEM detector and signal processing electronics [19, 24].

The number of electrons hitting the specimen and then the corresponding emission of secondary electrons is inherently a statistical process. This results in shot noise extending up to high frequencies in the image. Shot noise is usually assumed to be both zero mean and independent [6,9,19]. Noise can also originate from stray high-energy backscattered electrons colliding with either the detector

Section 2.1: Types of Noise

or the chamber walls and emitting additional secondary electrons.

Wells notes that the noise bottleneck for an SEM using a secondary electron detector is usually between the specimen and the detector, and that the noise due to a scintillator/photomultiplier system is largely negligible [24]. Even so, two practical limitations of the LEO 440 SEM electronics were observed to cause noise-related distortions in the final image: amplifier saturation and limited bandwidth.

A careful balance was needed between the photomultiplier voltage and any amplifier bias to avoid amplifier saturation. Amplifier saturation manifests itself in the final image as a disproportionate number of white pixels and can be seen in the image histogram (see Figure 2.1(c)). The limited bandwidth of the video chain at fast scan rates was determined to be responsible for the spreading of high frequency shot noise between several pixels in the same scan line [1]. This limited bandwidth distortion can be clearly seen as a one dimensional horizontal peak in the image's autocorrelation function (see Figure 2.1(b)). Care must taken when working with images containing limited bandwidth distortion since the noise in these images is obviously spatially correlated.

(a) Test Image

(b) Center 60 pixels of ACF

(c) Image Histogram
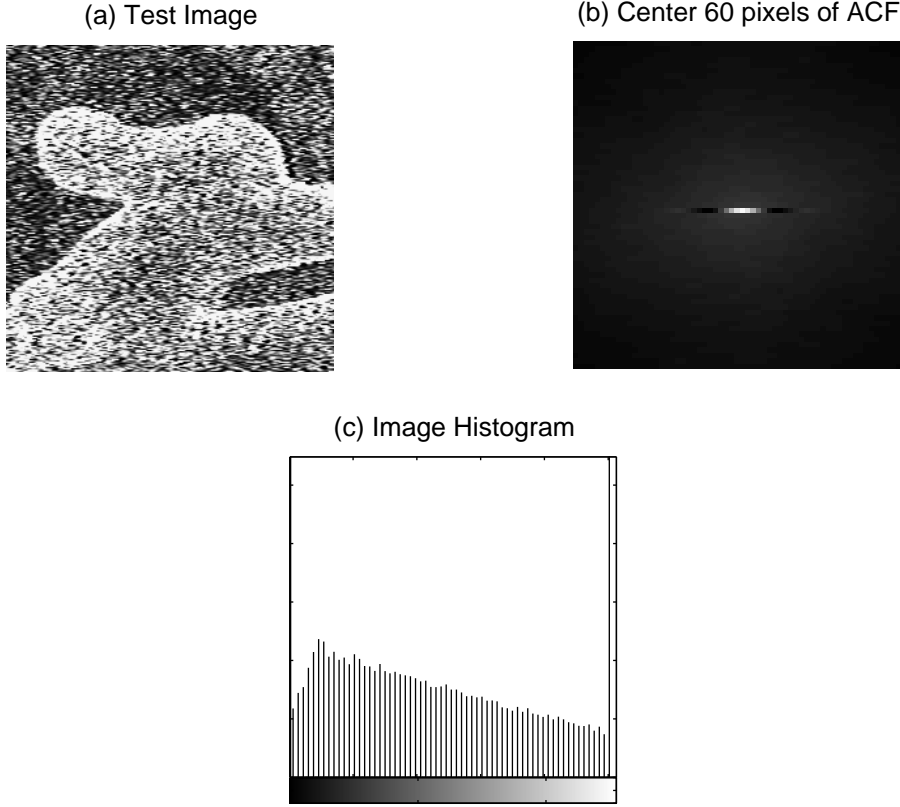
Figure 2.1: Example of Practical Distortions - The horizontal dashes in (a) are limited bandwidth distortion. This can be more clearly seen in the corresponding autocorrelation function (b). The disproportionate number of white pixels in (c) is due to amplifier saturation.

2.2 Measuring the SNR

If two different images are captured of the same specimen area within a short period of time, the correlation between these two images will be directly related to the SNR [9]. Assuming negligible drift effects, only the noise changes between images. For images g_1 and g_2 the cross-correlation coefficient is defined as

$$r = \frac{\sum\limits_{x=1}^{M}\sum\limits_{y=1}^{N}(g_1 - \overline{g}_1)(g_2 - \overline{g}_2)}{\sqrt{\sum\limits_{x=1}^{M}\sum\limits_{y=1}^{N}(g_1 - \overline{g}_1)^2 \sum\limits_{x=1}^{M}\sum\limits_{y=1}^{N}(g_2 - \overline{g}_2)^2}} \tag{2.1}$$

where $\overline{g}$ indicates the image's mean intensity. Since r is a measure of the correlation between two images, it is representative of the signal in the image, while $1-r$

Section 2.3: Frame Averaging

is the uncorrelated portion of the image and thus is representative of the noise in the image. The correlation coefficient can therefore be related to an image SNR as [7, 9]

$$SNR = \sqrt{\frac{r}{1-r}} \qquad (2.2)$$

2.3   Frame Averaging

Frame averaging is the most common type of noise reduction and most modern SEMs implement some form of hardware frame averaging. The simplest type of frame averaging involves summing pixel values over N frames and then dividing by N. The SNR improvement due to this form of frame averaging has been shown to be

$$SNR_{improvement} = \frac{SNR_N}{SNR_1} = \sqrt{N} \qquad (2.3)$$

where SNR_N is the SNR after averaging N frames [7]. The primary disadvantage, however, is that if scanning a frame requires T milliseconds, this technique only produces an image every $N \times T$ milliseconds. This means it will not be possible to perform significant frame averaging and still maintain a display at television refresh rates. It is for this reason that recursive averaging filters are usually used in most modern SEMs, including the LEO 440 SEM.

Figure 2.2 shows the theoretical SNR improvement and the experimental SNR improvement as a function of frame averaging. Frame averaging was performed using the LEO 440 image processing hardware and also in software using MATLAB. The SNR improvement for N frames of averaging in hardware was obtained as follows:

- Calculate SNR for no frame averaging ten times and use mean as SNR_1
- Use LEO 440 hardware to average N frames and capture averaged image
- Use hardware to average another N frames and capture a second image
- Determine SNR_N using these two images

The SNR improvement for N frames of averaging in software was obtained as follows:

- Capture $2N + 20$ raw images (no frame averaging)
- Use MATLAB and 20 of the captured images to determine SNR_1
- Use MATLAB and the remaining images to determine SNR_N

The experimental SNR improvement for frame averaging performed with the LEO 440 hardware actually exceeds the theoretical maximum. The author experimented with a variety of operating conditions, but the SNR improvement remained consistently higher than expected. It is suspected that this is an artificial improvement and is a consequence of the specific hardware implementation used by LEO. This can be confirmed by comparing hardware and software frame averaging. Figure 2.2 shows that the SNR improvement of images using software frame averaging corresponds well to theory. The author was unable to further investigate this discrepancy since detailed documentation on the LEO 440 image processing hardware was unavailable.
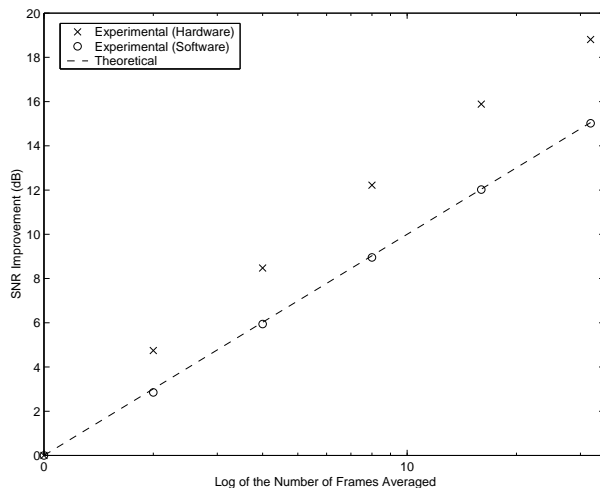
Figure 2.2: SNR Improvement vs. Frame Averaging

2.4   Pixel Averaging (Scan Rate)

In purely analog instruments, one could increase the SNR by decreasing the beam scan rate. A longer exposure per pixel increases the number of signal electrons reaching the detector, although this also slows the display refresh rate and can increase specimen damage [24]. In modern instruments a similar effect is implemented through pixel averaging. The beam rests on a specific pixel and a digitized value is stored at a constant rate. The average of these values is then recorded before the beam moves onto the next pixel. In the LEO 440 SEM there is a close relationship between pixel averaging and scan rate. The LEO notation for the scan rate level is an integer number n such that the number of pixels averaged is 2^n. This means that higher scan rate levels correspond to more pixel

11

averaging and slower scan rates. Figure 2.3 shows the SNR for increasing levels of pixel averaging.

The theoretical SNR improvement due to pixel averaging is the same as that achieved through frame averaging. Experimentally, however, it was found that decreasing the scan rate improved the SNR significantly less than expected, probably due to quantization and limited bandwidth effects. Decreasing the scan rate increases the available bandwidth in the video chain and thus reduces the pixel spreading observed at fast scan rates. This can be seen both in the autocorrelation function and in the frequency domain (see Figure 2.4). It is suspected that the limited bandwidth distortion acts as a high frequency filter and thus performs a form of smoothing. This smoothing would artificially raise the SNR at fast scan rates. This would explain the large discrepancy in SNR improvement at fast scan rates seen in Figure 2.3.
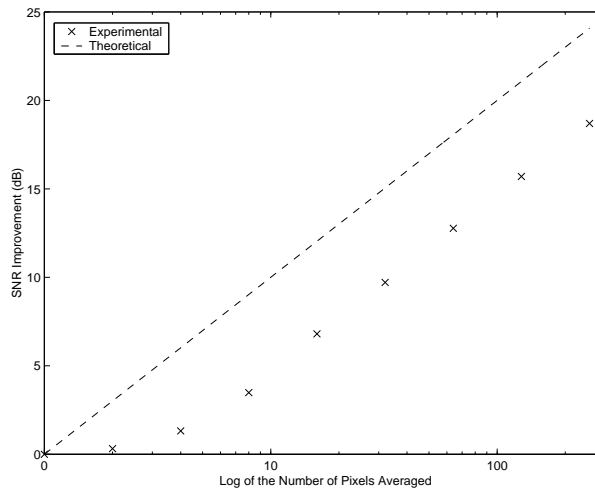
Figure 2.3: SNR Improvement vs. Pixel Averaging

2.5   Median Filtering

In addition to frame and pixel averaging, which are usually implemented in hardware, there is the potential for noise reduction preprocessing in software. This alternative has become more feasible with the drastic performance increases in general purpose microprocessors, as well as the recent addition of SIMD support in high end systems.[1] Although gaussian smoothing filters as well

---

[1]SIMD stands for Single-Instruction Multiple-Data and refers to computers that can perform the same operation on a large set of data in parallel. This is particular useful for image processing.
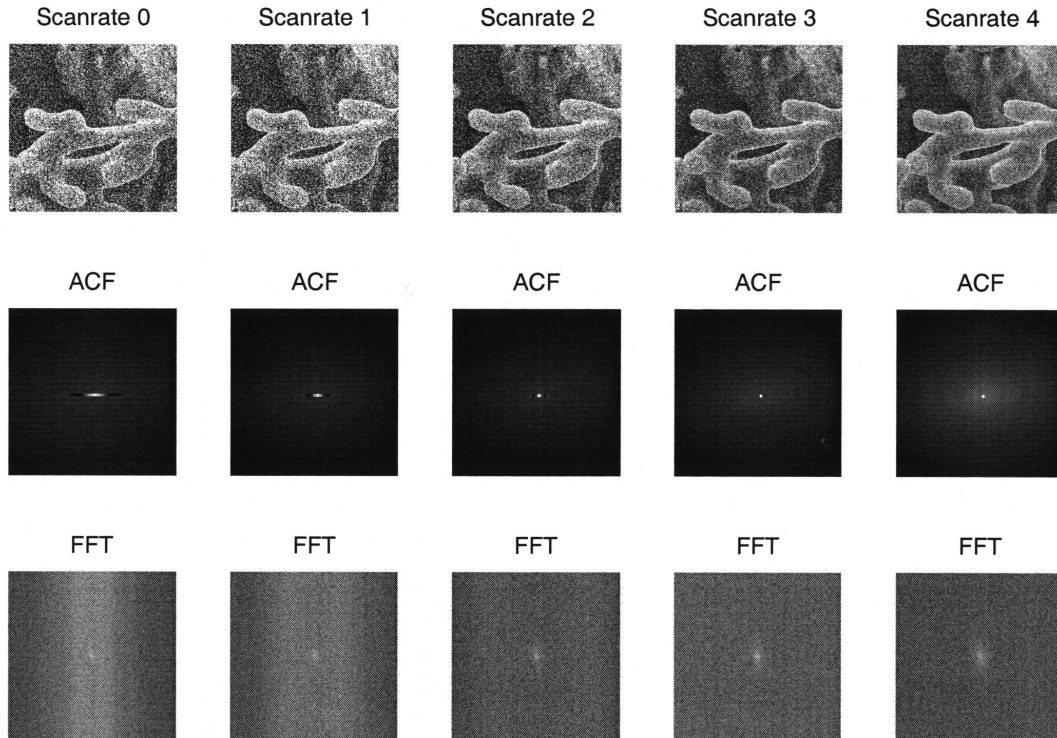
Figure 2.4: Limited Bandwidth Distortion vs. Scan Rate

as other more complicated non-linear filters were briefly investigated, median filtering seems to offer the best combination between performance, effective noise reduction, and preservation of specimen features.

If the noise in the LEO 440 SEM images was truly spatially uncorrelated, then the SNR improvement should be proportional to the number of elements over which the median was calculated and should be independent of the domain shape. It was predicted, however, that different domains would result in varying degrees of noise reduction due to the horizontal nature of the limited bandwidth distortion. The domains considered are shown in Figure 2.5. Computational constraints (processing time as well as the need for an odd number of domain elements to eliminate interpolation) dictated that a 3x3 pixel domain would be the best filter size. If the optimum partitioning algorithm is used, then the amount of processing time required to apply a median filter is directly proportional to the number of domain elements [4]. Thus the domains are shown in decreasing processing time from left to right.

Figure 2.6 shows the SNR for each of these domains as well as the SNR for images without median filtering. All of these images were captured with no pixel averaging. As expected the 3x3 square domain produces the greatest

Section 2.5: Median Filtering

gain in SNR, but the X domain produces nearly the same gain in SNR yet uses four less pixels. The vertical domain performs better than the cross domain even though it only uses three pixels. It was observed that this is a result of limited bandwidth distortion, since at slower scan rates (where limited bandwidth distortion is reduced) the cross domain begins to perform better than the vertical domain. This seems reasonable, since the vertical domain specifically addresses limited bandwidth distortion by using three pixels from three different scan lines.

A benchmark utility was written in C to determine the approximate time needed to perform each of the median filters on the target SEM platform. The benchmark data was largely consistent at various frame averaging and scan rate levels, and all tests were done on 512x512 pixel images. Results are shown in the following table:

Domain	Time (ms)
3x3 Square Domain	440
X Domain	220
Cross Domain	210
Vertical Domain	150

To better determine the tradeoff between the SNR increase from median filtering and the additional required processing time, the SNR for each domain is plotted against the total processing time in Figure 2.7. If the desired processing time is less than 300ms then median filtering is not a viable option. If a higher processing time is acceptable then the SNR increase due to X domain and vertical domain filtering outweighs the increased processing time.
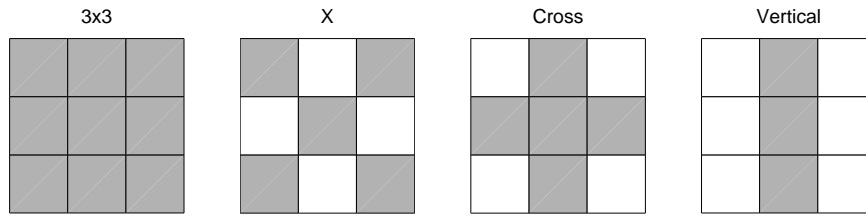
Figure 2.5: Median Filter Domains
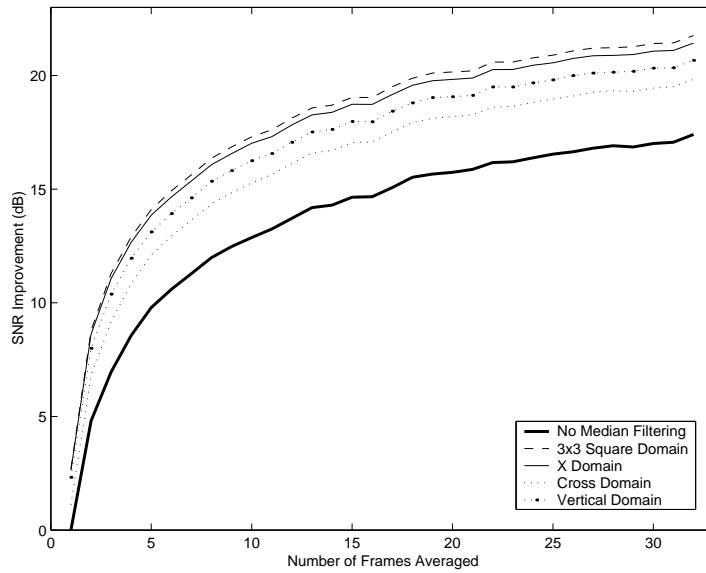
Figure 2.6: SNR Improvement for Various Domains vs. Frame Averaging
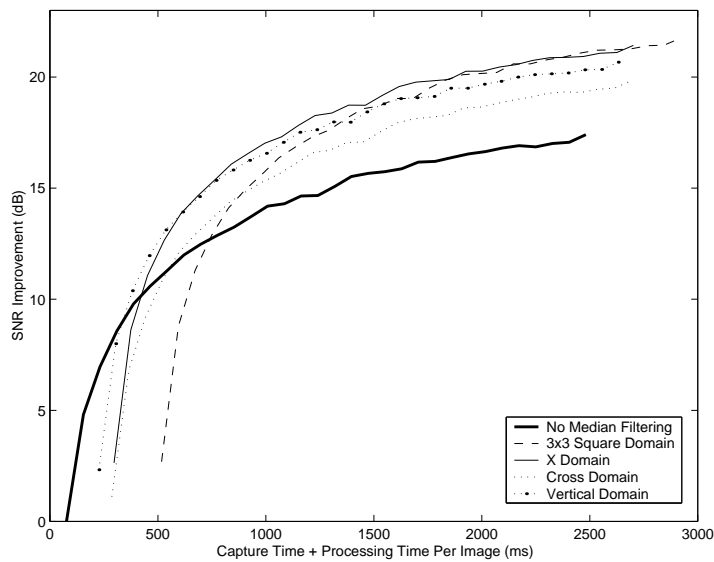
Figure 2.7: SNR Improvement for Various Domains vs. Frame Time

15

2.6 Comparisons and Analysis

Since frame averaging, pixel averaging, and median filtering all improve the SNR by different amounts and at different processing costs, it is important to consider a combination of the three in order to maximize the ratio between SNR and processing time. Figure 2.8 examines the SNR improvement for various combinations of frame averaging and pixel averaging. At any level of frame averaging, increasing the pixel averaging further increases the SNR improvement. Unfortunately, this increase in the SNR requires a significant increase in the processing time. To better analyze this tradeoff, Figure 2.9 shows the SNR improvement as a function of total processing time. Notice that the SNR improvement from decreasing the scan rate will never outweigh the increase in capture time. In other words, for any desired SNR, the optimal combination will be to use the fastest possible scan rate and then the appropriate amount of frame averaging. This also influences the usefulness of the median filters introduced in Section 2.5. Since the effectiveness of reduced domain median filters decreases as the scan rate slows down (owing to less bandwidth distortion), using median filters at slower scan rates will still not outweigh the required capture and processing time.

Figure 2.7 shows that for the fastest possible scan rate, median filtering with the X or vertical domain is a viable option if capture and processing times between 300ms and 1300ms are acceptable. Of course, these conclusions are specific to the particular system used for this work. Since the SEM control computer used for median filtering is several years old, newer systems may find a stronger argument for median filtering, especially to mitigate limited bandwidth distortion.
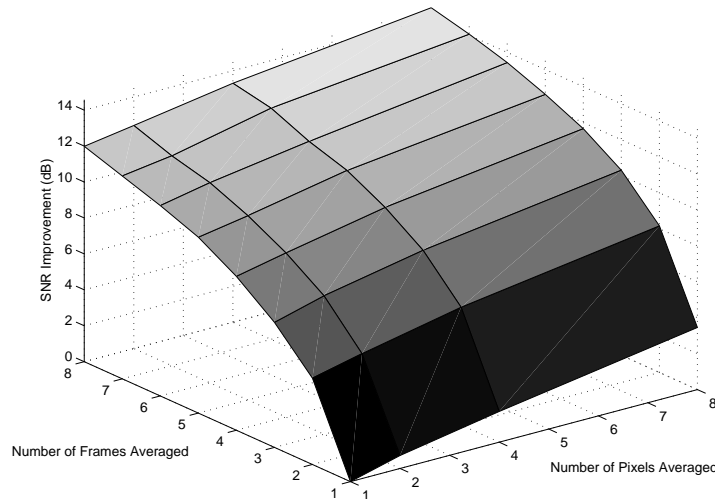
Figure 2.8: SNR Improvement vs. Scan Rate and Frame Averaging
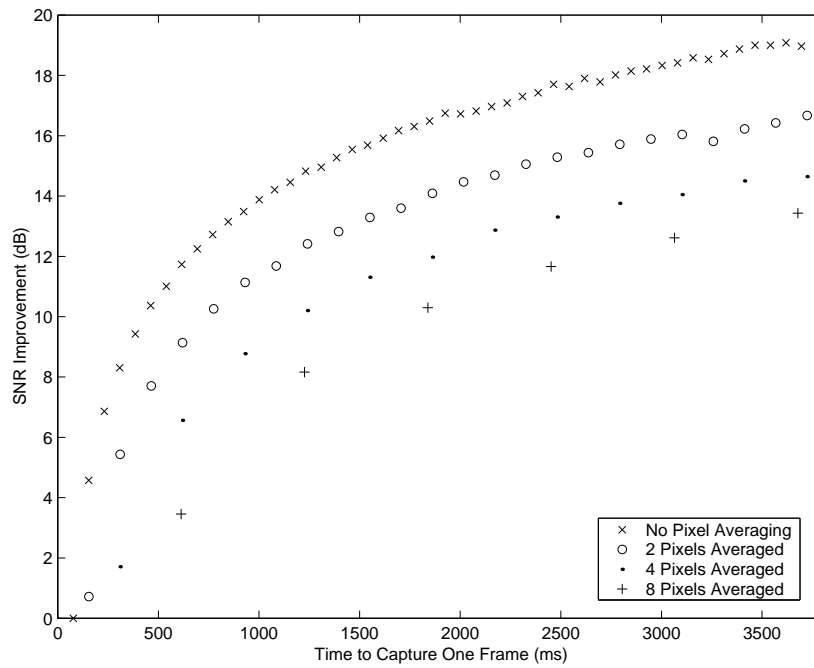
Figure 2.9: SNR Improvement vs. Frame Time

Section 2.6: Comparisons and Analysis

Chapter 3

Regions of Interest

Most autofocusing algorithms examine a reduced region of interest in an effort to decrease the time required for capturing and processing an image, but reduced regions of interest can also exclude essential sharpness information [21]. This chapter will briefly examine several possible alternatives including 1D spatial slices and reduced 2D regions. The ideal region of interest would be limited to the most important features in the image and exclude less significant features and featureless regions. Unfortunately, traditional feature detection algorithms are poorly suited to handle the level of noise commonly found in SEM images. This chapter introduces a sub-block method that identifies regions within the image that contain the greatest amount of sharpness information and thus allows for an efficient reduced region of interest.

3.1   1D vs. 2D Methods

Regions of interest used in previous autofocusing systems are diverse. Some early systems used dedicated hardware to process an entire 1024x512 pixel image [7], or used low level assembly programs to process reduced 64x64 pixel images [21]. Manufacturers have investigated 1D regions including lines along the primary axes, lines at 45° to the primary axes, and circular scans [5]. A more recent system uses the 1D data from a figure-eight beam scan for coarse focusing and then larger 2D regions for fine focusing [19]. The rising performance of general purpose microprocessors has allowed some researchers to analyze full images in software [15, 23].

Section 3.1: 1D vs. 2D Methods

Three 1D regions were considered and compared to a reduced 2D region. The three 1D regions are as follows:

- *Circle Method*: Pixels which lie on a circle whose radius is one quarter the width of the image

- *Two Line Method*: Pixels which lie on either of the two lines passing through the center of the image, one in the x direction and the other in the y direction

- *Four Line Method*: Same as above except also including those points which lie on lines which are at 45° to the x-y axes

The full-size image store for the LEO 440 SEM is 1024x768 pixels. It was decided that a reasonable reduced 2D region of interest would be a 512x512 centered square. A variance sharpness measure was applied over each of the three 1D domains as well as the entire 512x512 image for various noise levels. The specimen used is shown in Figure 3.1, and the results are shown in Figure 3.2 (see Section 4.5 for more on the variance sharpness measure). Note that the 1D domains were taken from the reduced 512x512 region and not from the entire image. As expected the 1D regions are more noisy than the full 512x512 domain, yet all four methods still peak at the same focus. More pixels in the domain result in a smoother sharpness curve. The 2D domain's robustness to noise will be a key advantage when considering the more sophisticated peak search algorithms introduced in Chapter 5.
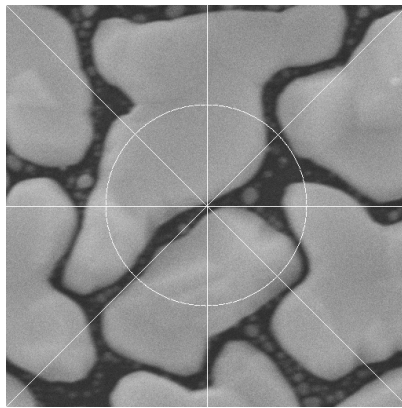
Figure 3.1: Sample Used For Domain Tests - Gold on Carbon 50,000x
White lines indicate all three 1D regions of interest

A primary concern with these and any reduced region of interest, is that the reduced region may not include a feature or edge. Therefore, a common
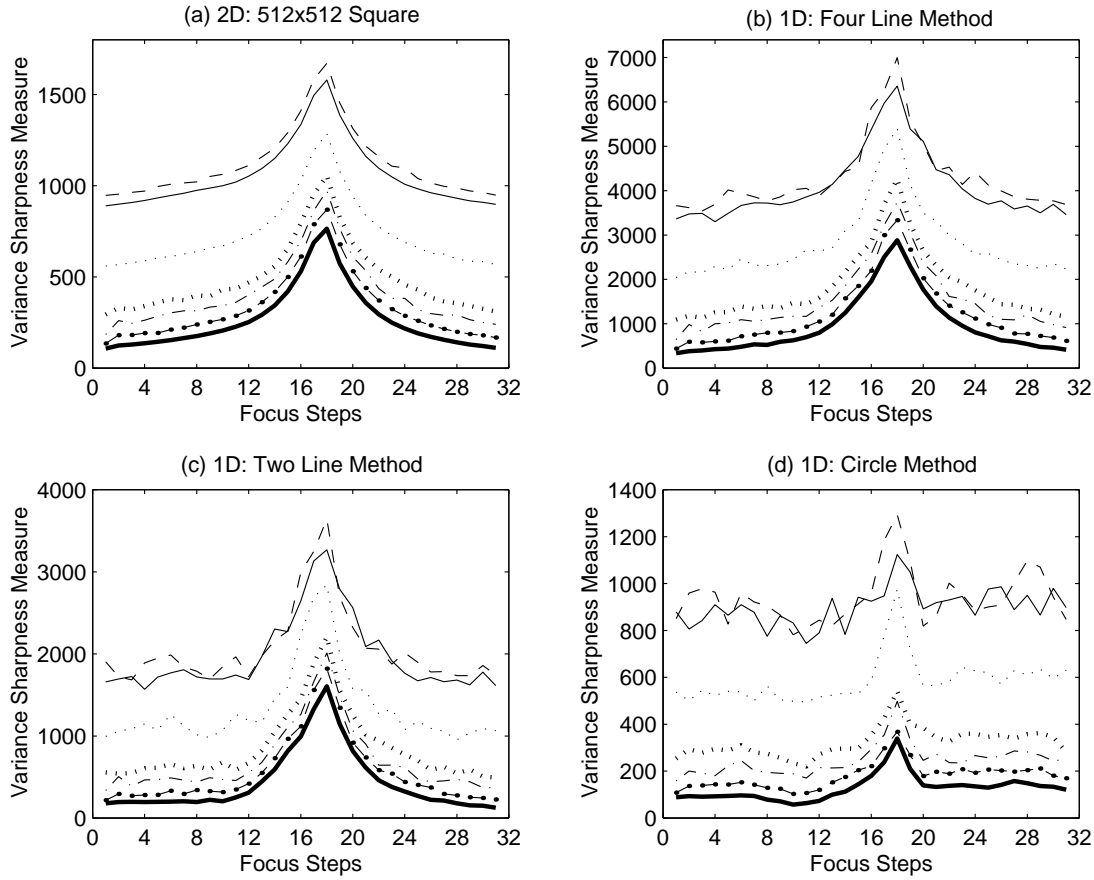
Figure 3.2: Sharpness Curves for Various Regions of Interest - Each plot includes seven increasing levels of noise reduction (dark solid line having the greatest while the thin long dashed line having the least).

assumption is that the user has positioned the specimen such that the item of interest is in the center of the screen (and thus in the center of the image). This assumption explains why the reduced 512x512 region is centered and also explains the importance of the radius used in the circle method. If the radius is too large, then the circle could miss a small feature at the center of the image. If the radius is too small then the circle could fail to cross an edge of a larger feature. Several experiments were made, and the optimum radius was found to vary for different specimens.

A reduced region of interest will theoretically improve performance in two ways: reducing the time to capture the image and reducing the time to process the image. Ideally, the SEM would just scan the region of interest, but the LEO 440 SEM only allows for rectangular reduced raster scans. This means that the circle method and the four line method (due to the 45° lines) cannot be scanned directly and must be approximated by selecting pixels from a rectangular image.

In addition, the maximum scan rate possible in the LEO 440 SEM for narrow regions (i.e. short scan lines) is less than the maximum scan rate possible for wider regions. This restriction is due to hardware constraints on the maximum speed for scan line buffering [13]. Thus scanning a single vertical line for the two line and four line methods is disproportionately slow when compared to scanning the same length line in the horizontal direction. These practical limitations severely reduce the advantage gained from 1D regions of interest.

3.2 Sub-block Method

In any image there are areas that contain a large amount of useful sharpness information such as edges, textures, and contours. There are also areas that can contain misleading or very little sharpness information such as high-noise areas or low contrast features. This implies that the optimum region of interest is strongly dependent on the specific specimen and noise conditions. Traditional feature detection methods such as thresholding based on an intensity distribution and edge detection were investigated in an attempt to determine the optimum region of interest, but these techniques had little success due to the high noise levels inherent in electron microscopy.

The sub-block method attempts to determine a near optimal region of interest by comparing the relative sharpness of smaller statically sized regions. The image is first divided into 16 128x128 pixel 'sub-blocks'. Then, using a selection method, the four blocks with the greatest sharpness information are chosen as the region of interest. The sub-block method not only reduces computation and processing time, but also increases a focusing algorithm's robustness to noise by eliminating noisy portions of the image from consideration. The following subsections examine the sub-block selection method and practical considerations when attempting to use the sub-block method to reduce the image capture size.

3.2.1 Sub-block Selection Method

One can either select sub-blocks during a focus sweep (in-sweep selection) or select sub-blocks after a preliminary coarse focus sweep (post-sweep selection). In-sweep selection requires the sharpness measure to be normalized over different sized domains, since the domain will change throughout the sweep. In-sweep selection also raises the question of when during the sweep sub-blocks should begin to be removed. If sub-blocks are removed too early then the wrong blocks could be removed, but if they are removed too late then the gain due to reducing the region of interest is small. Post-sweep selection avoids these complications,

and since most full autofocusing algorithms include a coarse focus sweep, there is little overhead.

Figure 3.3 shows the variance for each sub-block for a standard focus sweep. Notice that some sub-blocks have higher peaks than others indicating that these sub-blocks contain relatively more sharpness information. Figure 3.4(a) shows the test sample used in Figure 3.3 and the selected sub-blocks. Sub-blocks with sharper peaks correspond to edges or areas of high contrast in the image. Therefore, the method used was to select the four sub-blocks with the greatest variance peaks during the coarse focus sweep. The region of interest is then reduced to these four sub-blocks for a fine focus sweep. Figure 3.4(b) shows that the sharpness measure from just the four selected sub-blocks still peaks at the same focus step even though it only uses one fourth the number of pixels. Comparing the three curves, one can see that the curve from the unselected sub-blocks flattens out more towards the tails. Chapter 5 will show that more sophisticated peak search algorithms require the sharpness curve to be monotonically decreasing away from the maximum over a large range of focus values. Eliminating the unselected sub-blocks extends this range.
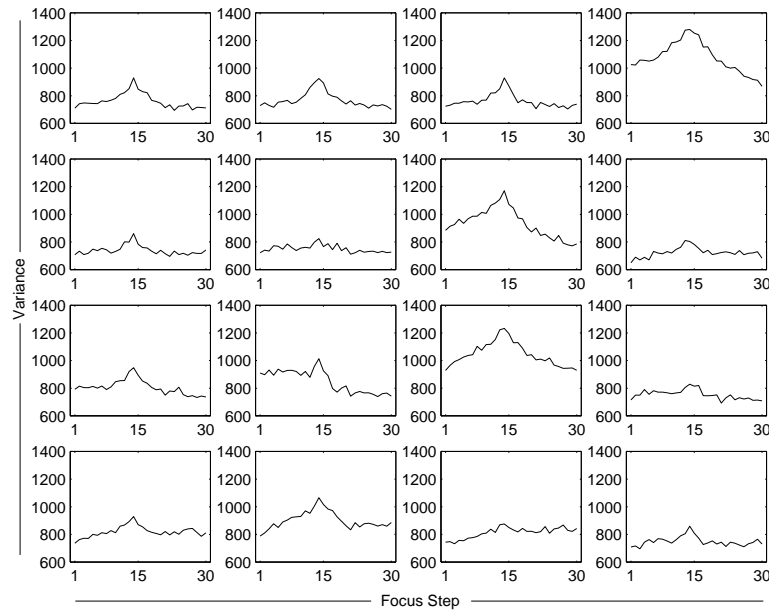
Figure 3.3: Individual Variance Curves for Each Sub-block

3.2.2 Using Sub-block Method to Reduce Capture Time

Once a sub-block selection method is used to reduce the region of interest, the time required to capture an image should also be reduced. Ideally, the SEM

Section 3.2: Sub-block Method
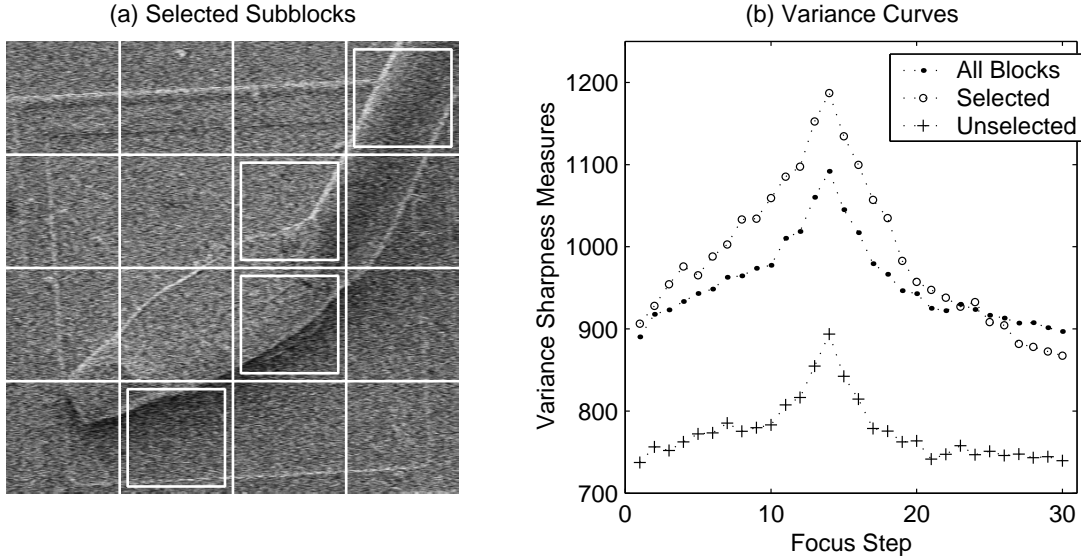
Figure 3.4: Selected Sub-blocks and Total Variance Curves

would only scan the selected sub-blocks and thus the time to capture the sub-blocks would be one fourth the time to scan the entire 512x512 image. Three practical limitations of the LEO 440 SEM reduce this gain in capture time: the SEM can only scan a single continuous rectangular region at a time, there is a required overhead for any image capture, and the maximum scan rate decreases as the width of the region to be captured decreases.

These limitations complicate the process of determining the size of the image capture regions which will obtain the selected sub-blocks. Simply capturing each 128x128 sub-block separately will always take longer than capturing the whole image. For example, capturing a full 512x512 image at the fastest possible scan rate with no frame averaging takes 67 milliseconds. Individually capturing four 128x128 sub-blocks at the fastest possible scan rate takes a total time of 220 milliseconds. Obviously, if the capture speed is to be increased, more clever methods will be required.

There are many different ways to capture any specific set of selected blocks. An optimum solution can be found by performing an exhaustive search of all possible capture combinations and then choosing the fastest one. Performing this exhaustive search is an extremely time consuming process, so the optimum capture combination would need to be calculated beforehand for every possible set of selected blocks and stored in a lookup table. Simply storing and searching such a lookup table would in itself be very expensive, so instead a non-optimal solution was adopted. This solution uses four methods to determine four different capture combinations, and then chooses the fastest one. The four methods are

described below and illustrated in Figure 3.5.

- *Full Image*: Capture the entire 512x512 pixel image

- *Single Bounding Rectangle*: Capture the smallest rectangle that still includes all selected sub-blocks

- *Multiple Bounding Rectangles*: If there is a row with no selected blocks in between two rows with selected blocks then make two captures, one for the upper portion and one for the lower portion

- *Quadrants*: If the selected blocks are only in opposite quadrants capture just those quadrants

It is difficult to quantify the overall speedup from using the sub-block method owing to the dependence of the capture time on which sub-blocks are selected. Regardless, timing analysis of the processing time shows that calculating the variance (and probably most other sharpness measures) for the four sub-blocks takes one fourth the time for the entire image. Since the overhead involved in determining the best capture size is relatively small, the sub-block method should always be faster than capturing and processing the entire image. With the LEO 440 SEM, however, it will never be four times as fast.

Even larger performance gains can be expected if the full 1024x768 image is divided into 48 sub-blocks. Another key advantage of this method is the potential for parallelism. If implemented in software, several sub-blocks could be assigned to individual processors, or if implemented in hardware, each sub-block could have its own variance calculation logic. The fundamental idea behind the sub-block method is a novel way to efficiently reduce the region of interest and is applicable to any focusing operation.

Section 3.2: Sub-block Method
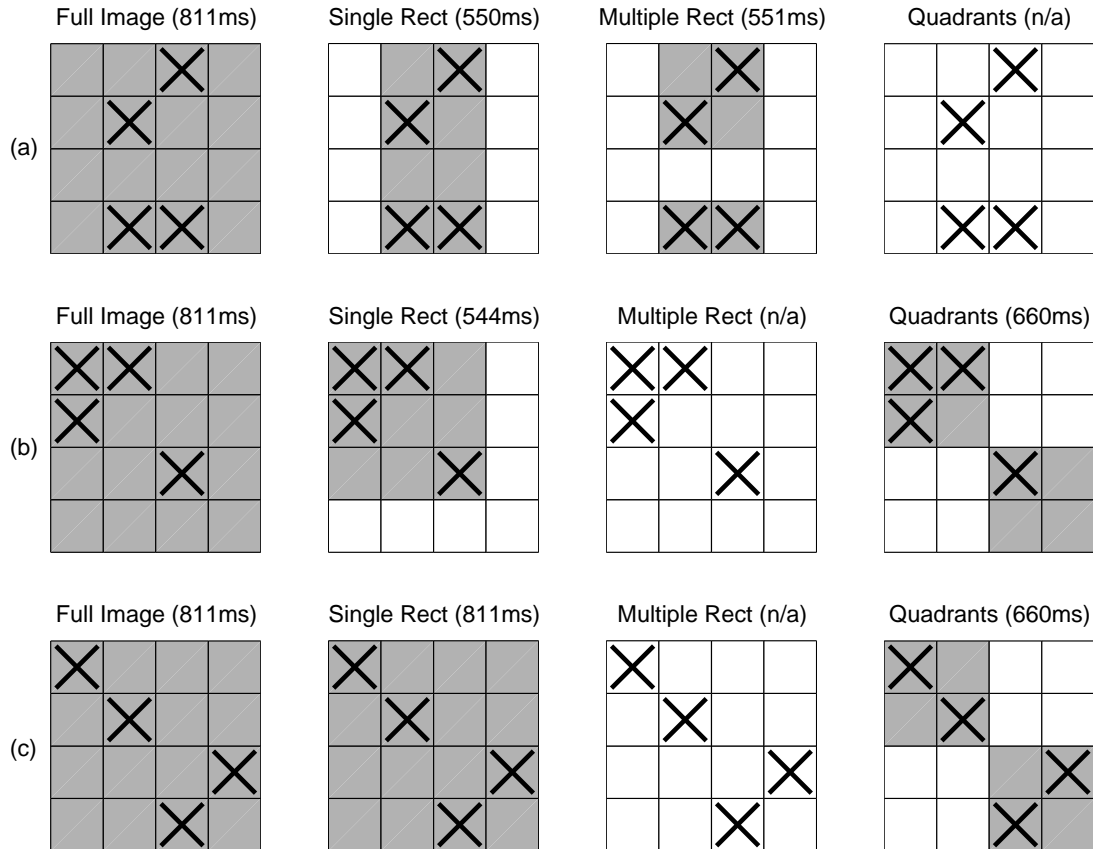
Figure 3.5: Image Capture Reduction Methods - Each of the four methods will produce a way to capture the four selected sub-blocks (marked with an X). The gray regions indicate the regions to capture. Notice that the smallest captures are not necessarily the fastest owing to the overhead required for each capture. The capture times given refer to scan rate level 2 and averaging 6 frames. Different noise reduction levels yield different results.

Chapter 4

Sharpness Measures

A great deal of research has been invested in analyzing various sharpness measures and focus functions. Work has been done on measures which rely on image gradients [12, 21], cellular logic [8], spectrum analysis [19, 23], histogram thresholding [11], image correlation [19], and image variance [6, 7]. Groen *et al.* and Firestone *et al.* offer a good review of sharpness measures for optical microscopy, although they are also readily applicable to electron microscopy [8, 11].

This chapter will identify the key properties of a good sharpness measure and will then examine four specific sharpness measures: the gradient method, the frequency domain method, the auto-correlation method, and the variance method. It was found that the variance measure yields the best balance between complexity and robustness to noise.

4.1 Evaluating Sharpness Measures

Before discussing specific sharpness measures it is important to identify key properties which make one measure better than another. The following are the basic criteria for a suitable sharpness measure [11, 25].

- *Unimodality*: Produces a single maximum
- *Accuracy*: Produces a maximum at the best focus
- *Reproducibility*: Has a sharp top
- *Monotonicity*: Is monotonically decreasing above and below focus
- *Applicability*: Works well for any reasonable specimen and conditions
- *Implementation*: Is able to be implemented easily and efficiently

Section 4.2: Gradient Method

Of particular importance are the unimodality and monotonicity properties. If these assumptions are true for a certain level of noise reduction then efficient maximum sharpness search algorithms can be used.

4.2 Gradient Method

Gradient sharpness measures are quite common in the literature largely due to the ease with which they can be implemented. Such measures were some of the first to be used in electron microscopy [12,21], and can also be found in more recent implementations and literature [3,5]. Even so, it is widely recognized that the gradient measure is particularly sensitive to noise and thus requires additional noise reduction to produce reliable results [3,7].

If $i(x,y)$ is a digitized image, then the 2D normalized gradient for $i(x,y)$ and a given integer parameter n is defined as

$$f_{grad,n} = \frac{1}{MN} \sum_{x=1}^{M-n} \sum_{y=1}^{N-n} \sqrt{[i(x+n,y) - i(x,y)]^2 + [i(x,y+n) - i(x,y)]^2} \quad (4.1)$$

The squared gradient is sometimes used to accentuate sharper edges, but this can also accentuate high frequency noise [25]. In most of the literature n is set to one, although reference is also made to the Brenner measure where $n = 2$ [3,8,25]. Large n is equivalent to a low pass filter, and thus it increases noise robustness but can also cause the measure to miss small features. Although Brenner's method is usually given as an acceptable tradeoff in the literature, it was found that $n \leq 10$ will still yield accurate results since it is unlikely the user will be viewing a specimen with no features greater than 10 pixels. Therefore, n was chosen to be ten.

Figure 4.1 demonstrates the influence of the parameter n on the gradient. The sample used was gold on carbon at 50,000x magnification, which included both coarse (100-200 pixels) and fine detail (< 100 pixels). The image was scanned at scan rate level 0 and three frames averaged. As n increases the measure becomes more robust to noise but also misses very fine detail in the specimen. Notice that for small n the gradient actually decreases through the optimum focus due to the overwhelming presence of high frequency noise.
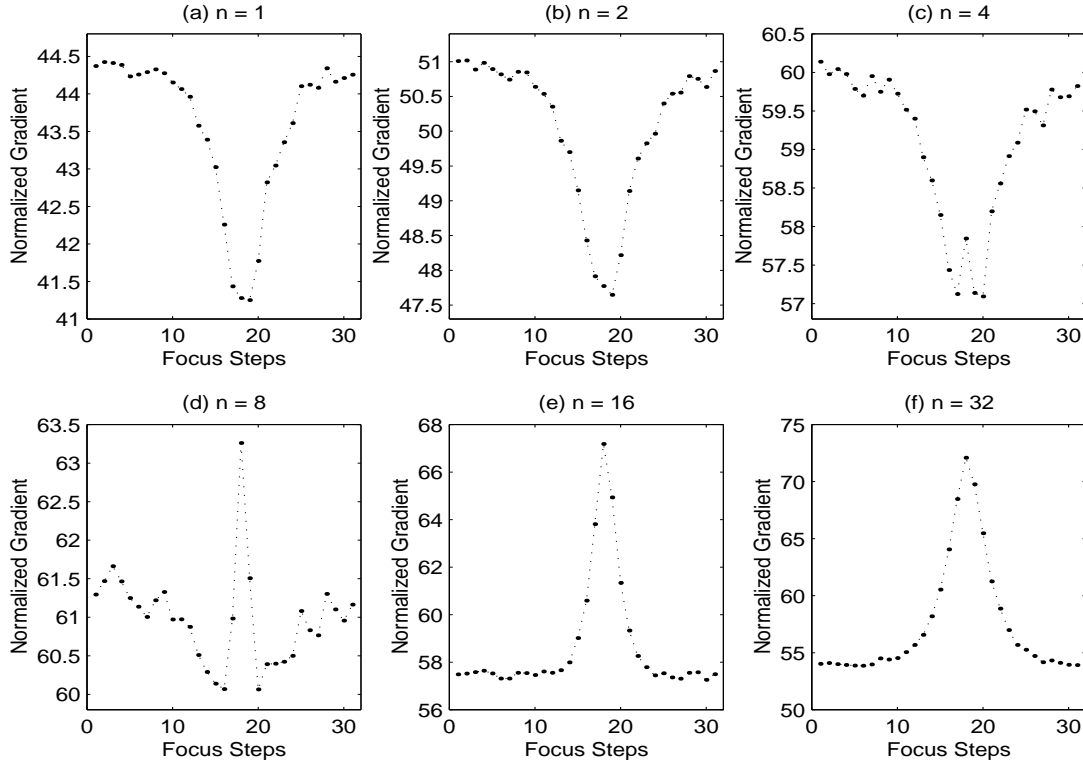
Figure 4.1: Gradient for Various n

4.3   Frequency Domain Method

Frequency domain methods work on the principle that as an image moves into focus edges become sharper and thus the high frequency components increase in magnitude. It was noted very early on that frequency domain sharpness measures should be more robust to noise than gradient methods [7], but computational limitations prevented simple calculation of the power spectrum. Performance increases in general microprocessors over the past ten years and the efficient fast Fourier transform (FFT) algorithm have made frequency domain methods a feasible alternative. Exactly how to use the frequency domain to calculate a sharpness measure varies. Some examples include calculating the number of pixels above a specified threshold in a ratio of two FFTs obtained at different focus values [19] and using parameters from the spectral 'moment of inertia' [23].

For this work, the frequency component magnitudes below a certain threshold frequency were summed. This sharpness measure is defined below, where $F(x, y)$ is the Fourier transform of the image $f(x, y)$ and Ω is the discrete threshold

Section 4.4: Auto-Correlation Method

frequency in pixels.

$$f_{freq,\Omega} = \frac{1}{4\Omega^2} \sum_{u=-\Omega}^{\Omega} \sum_{v=-\Omega}^{\Omega} |F(u,v)| \qquad (4.2)$$

Figure 4.2 is similar to Figure 4.1, but instead of performing a low pass filter in the spatial domain (using the parameter n) it performs a low pass filter in the frequency domain (using the parameter Ω). As in the spatial domain, care must be taken when choosing a threshold frequency. If it is too low then higher frequency signal data could be missed. Experiment suggested that $\Omega = 50$ seemed to provide a reasonable balance between these concerns.

Figure 4.2: Sum Of Frequency Components for Various Ω

4.4   Auto-Correlation Method

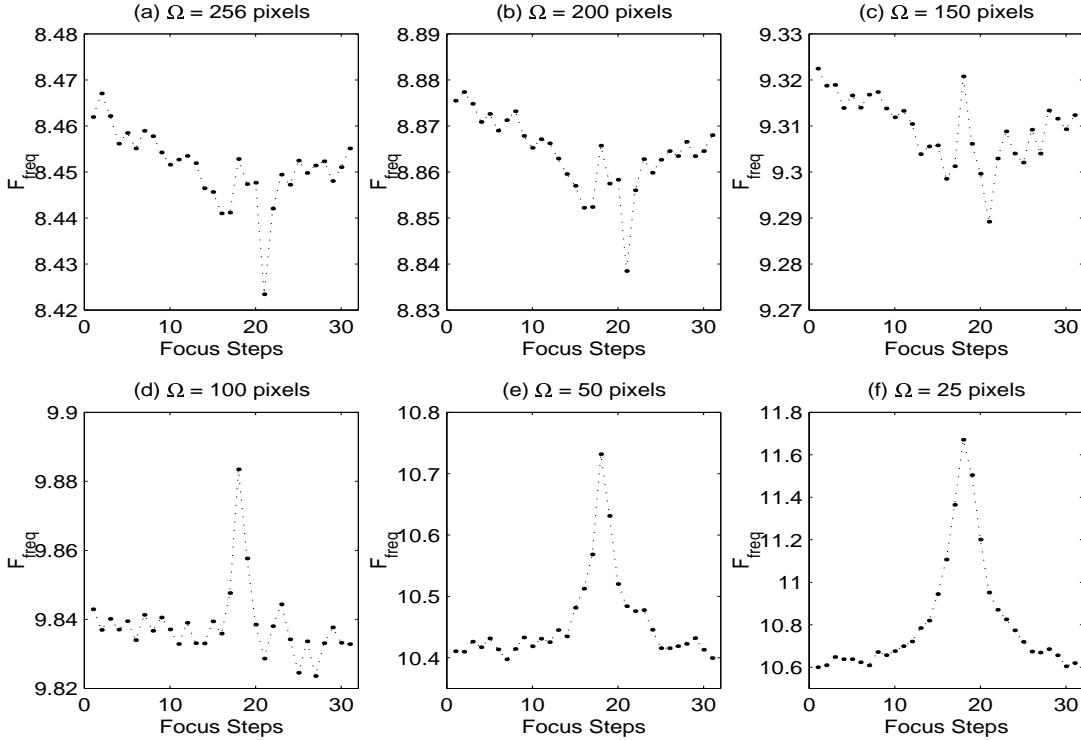Since the auto-correlation function (ACF) is the inverse Fourier transform of the power spectrum [10], it is expected that the ACF will contain sharpness information. Defocused images contain large smooth regions that result in a low broad central ACF peak. Focused images contain small highly correlated regions

that result in a tall sharp central ACF peak. The height of the ACF has been used before to measure defocus [19].

Two sharpness measures were calculated from the 2D ACF. The first uses the difference between ACF values at two locations along the x-y axes, while the second uses the area under the central ACF peak. If $a(x, y)$ is the ACF for an image, then the first sharpness measure is defined below for the given parameters n and m

$$f_{ACFdiff,n,m} = [a(0, n) - a(0, n + m)] + [a(n, 0) - a(n + m, 0)] \qquad (4.3)$$

Parameter n was chosen to be 5 in order to avoid the noise-influenced portion of the ACF near the origin (see Figure 2.1). Parameter m was chosen to be 10. The second sharpness measure is defined below for the given parameter n

$$f_{ACFsum,n} = \frac{1}{4n^2} \sum_{\substack{x=-n \\ x \neq 0}}^{n} \sum_{\substack{y=-n \\ y \neq 0}}^{n} [a(x, y) - a_{min}] \qquad (4.4)$$

where $2n$ is the size of the central square over which to sum the ACF and a_{min} is the minimum ACF value within this central square. Through experiment it was decided to set $n = 50$. The ACF can be efficiently calculated in the frequency domain using the following relationship [10]

$$A(u, v) = F(u, v)F^*(u, v) \qquad (4.5)$$

where $F(u, v)$ is the Fourier transform of image $f(x, y)$, $A(u, v)$ is the Fourier transform of the ACF, and $*$ is the the complex conjugate.

4.5 Variance Method

Image variance has been used as a sharpness measure [5,7], and its robustness to noise has been previously noted [25]. Variance should peak at the best focus, since images in focus will have greater intensity variation than blurred defocused images. Erasmus and Smith provide a theoretical explanation for the dependence of variance on the beam radius (and thus on defocus) [6]. The variance sharpness measure is defined as

$$f_{var} = \frac{1}{MN} \sum_{x=1}^{N} \sum_{y=1}^{M} [g(x, y) - \overline{g}]^2 \qquad (4.6)$$

where $\overline{g}$ is the mean intensity in the image $g(x, y)$.

It should be noted that the covariance is sometimes used instead of the variance [6]. In such cases two consecutive images are captured from the SEM, and

the covariance between them used as a sharpness measure. It is argued that such a measure is more robust to noise, but it should also be noted that the covariance requires two separate image captures. Each image capture adds overhead in addition to the time required for noise reduction. To achieve a reasonable processing time such a system would probably use lower amounts of noise reduction per frame. The same general result can be achieved by simply using the variance of one image with higher noise reduction.

4.6   Comparisons and Analysis

The sharpness measures discussed in the previous sections are all related. The gradient method uses the spatial domain to measure the amount of high frequency information, while the frequency domain method uses the power spectrum for the same reason. The ACF is just the inverse Fourier transform of the power spectrum, and the variance is the ACF at the origin (ignoring the effect of the mean intensity). Although related, each measure performs differently with respect to the properties laid out in Section 4.1.

We will first examine each method at various focus step intervals. This will primarily reveal information concerning each method's repeatability and accuracy. Ideally we desire a measure which has a sharp top even at small focus step intervals. We can also observe the sharpness measure's monotonicity at the larger stepsizes. Figure 4.3 shows each sharpness measure for three different focus stepsizes. The sample used was sublimated titanium at 3,000x magnification. Extensive noise reduction was used in an effort to better examine the measure's sharpness at the maximum. Each image was captured with a pixel averaging of 16 and a frame averaging of 25. All plots in the same column have the same vertical scale.

Figure 4.3 reveals several important points. Although f_{ACFsum} is quite wide at its base indicating strong monotonicity, it has extremely low repeatability due to the very flat curve at small stepsizes. $f_{ACFdiff}$ is slightly better, but is still poor when compared to the other three measures. The remaining three measures have relatively sharp tops at small stepsizes with f_{grad} being slightly sharper than f_{var} which is in turn sharper than f_{freq}. Also notice that f_{var} and f_{freq} have better monotonicity at the tails, as compared to f_{grad} which has very flat tails. These results were found to be typical of most specimens.

Figures 4.4 and 4.5 illustrate how each sharpness measure responds to various noise levels. An ideal sharpness measure should change little as the noise level increases, and should also maintain its unimodality and monotonicity. The sample used in Figure 4.4 was sublimated titanium while the sample used in Figure 4.5
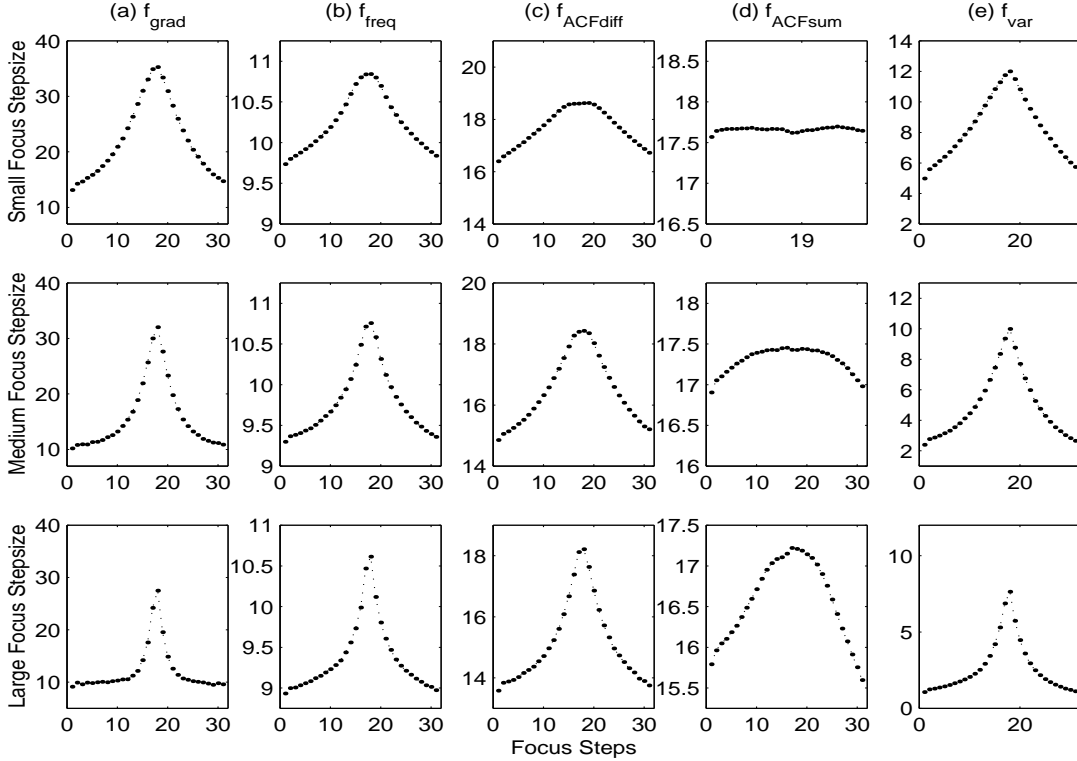
Figure 4.3: Sharpness Measures at Various Focus Stepsizes

was gold on carbon. All sharpness results were divided by the maximum value in that sweep so that all results peak at one.

Of the various sharpness measures, f_{grad} seems to be the most susceptible to noise even with a spacing of 10 pixels. $f_{ACFdiff}$ shows excellent noise robustness, except for images with bandwidth distortion (i.e. pixel averaging of 1). f_{ACFsum} response to noise is inconsistent. In 4.4(a) the measure actually increases at severe defocus and high noise levels due to the large correlation in limited bandwidth distortion. Obviously, bandwidth distortion plays an important part in the two ACF sharpness measures.

f_{freq} and f_{var} both seem to be quite robust to noise. Both remain unimodal and maintain their monotonicity near the tails as the noise level increases. A key advantage of the variance measure over the frequency method is its ease of implementation. Let N be the number of pixels in the image such that $N = 2^n$ and $n \geq 1$. Then the variance measure requires N additions to determine the mean, N subtractions, and N multiplications. The FFT requires $\frac{1}{2}Nn$ complex multiplications and Nn complex additions [10]. This is equivalent to $2Nn$ simple multiplications and $2Nn$ simple additions. In addition, the frequency domain sharpness method requires $4\Omega^2$ additions to compute the sum (where Ω is

33

Section 4.6: Comparisons and Analysis

Figure 4.4: Sharpness Measures at Various Noise Levels - Specimen #1
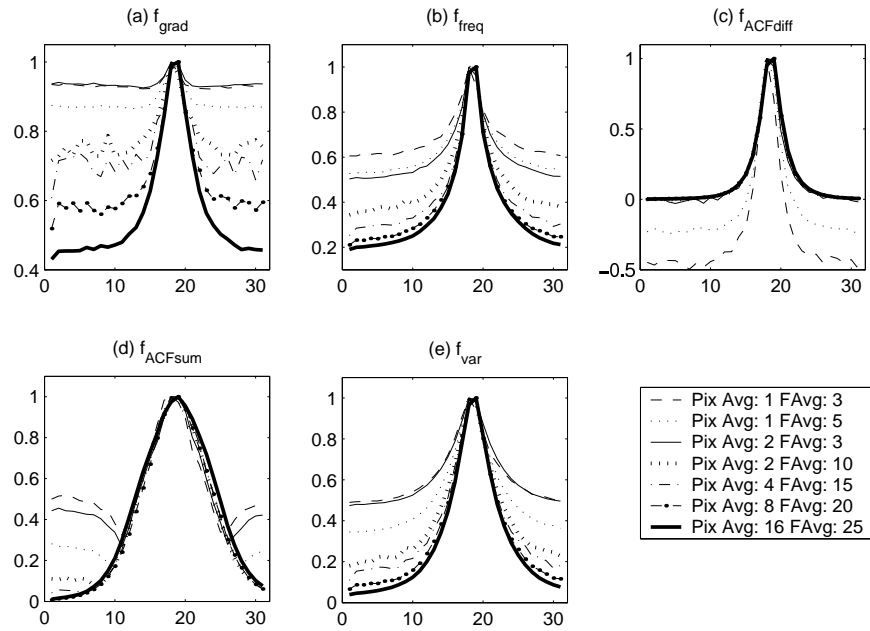
Figure 4.5: Sharpness Measures at Various Noise Levels - Specimen #2
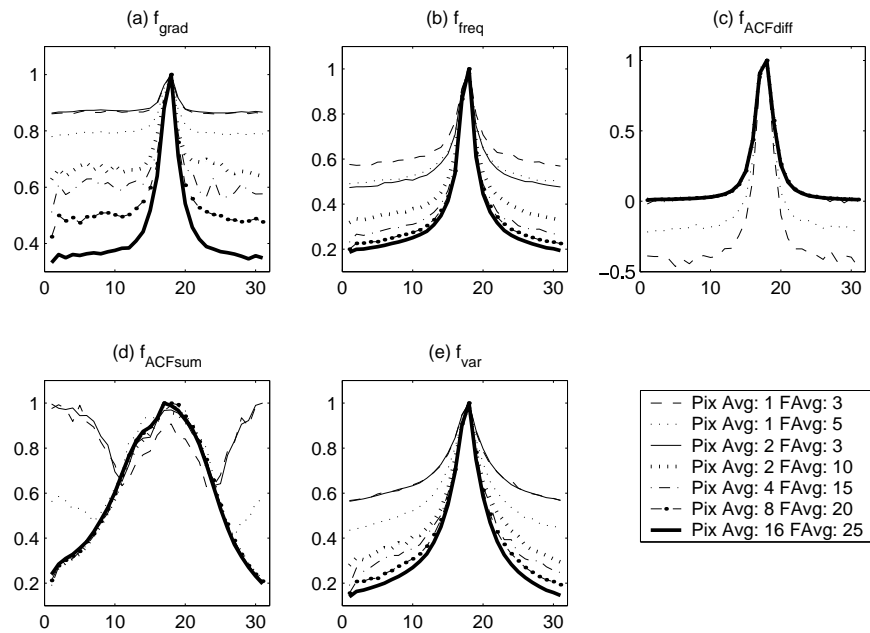
the threshold frequency). Since $4Nn + 4\Omega^2 > 3N$ the total operations for the frequency domain method outnumber the operations required for the variance method (for any n or Ω).

The following table shows the actual time required for each sharpness measure. Sharpness measures were calculated for a 512x512 image on the SEM control computer. Each sharpness measure was calculated using the MATLAB C++ libraries, and no real attempt was made to optimize these implementations. Thus these results should mainly be treated as a relative measure of performance. As expected, f_{var} is by far the fastest sharpness measure, and the ACF measures are the slowest. As seen in Equation 4.5, the ACF measures use both a Fourier transform and an inverse Fourier transform, and it is therefore reasonable that the ACF measures are more than twice as slow as the f_{freq} measure. Since the SEM control computer is relatively old, it is expected that these results can be considerably improved upon by simply using a faster computer.

Method	Time (sec)
f_{grad}	2.20
f_{freq}	1.79
$f_{ACFdiff}$	4.06
f_{ACFsum}	4.19
f_{var}	0.51

Another key advantage of the variance method is that it does not call for any parameters or thresholds. The other methods require assumptions concerning the amount of detail in an image or the size of features in order to assign reasonable parameters. The variance measure was chosen as the primary sharpness measure for this work because of its strong repeatablity, accuracy, unimodality and monotonicity properties regardless of noise, as well as its simple implementation.

Section 4.6: Comparisons and Analysis

Chapter 5

Maximum Sharpness Search Algorithms

Traditionally, the chosen sharpness measure is applied to a linearly increasing set of objective lens excitations, and the excitation corresponding to maximum sharpness is considered to be the best focus. Although it has been noted that more efficient search algorithms are possible (assuming unimodality and monotonicity) [16, 25], there seems to be little practical work on implementing such algorithms for electron microscopy. This is probably due to the complications arising from hysteresis effects and the low SNR of most SEM images. This chapter will investigate several maximum sharpness search algorithms including the traditional fixed stepsize search, the iterative search, the variable stepsize search, and the Fibonacci search. Interpolation based on a model of variance as a function of defocus will also be examined.

5.1 Preliminary Considerations

Several factors can influence the success of more sophisticated search algorithms. These include the chosen level of noise reduction, hysteresis effects, and the desired level of accuracy. More sophisticated search algorithms usually require greater noise reduction than traditional fixed stepsize sweeps. The underlying idea is that the decrease in the number of images needed due to a more sophisticated search algorithm will outweigh the increase in processing time required for greater noise reduction. This was found to be reasonably true for the algorithms investigated.

Section 5.1: Preliminary Considerations

5.1.1   Hysteresis

A current applied to an electromagnetic lens aligns magnetic regions in the ferromagnetic material of the lens and thus produces a magnetic field related to the strength of the applied current. The magnetic material in the electromagnetic lenses causes the relationship between applied current and magnetic field to be non-linear and is known as the hysteresis effect [20]. Hysteresis is of particular importance in autofocusing since it implies that the sharpness at a specific lens excitation is dependent on the history of the lens, and thus if we return to the same lens excitation at a later time in the search the sharpness may not be the same.

Figures 5.1 and 5.2 illustrate the hysteresis effect. In Figure 5.1, a set of linearly increasing focus values have been swept in the same direction three times. The focal length was set to a small value between sweeps to help minimize hysteresis effects. This is a common technique and is usually adequate [21], although more extensive cyclic approaches are possible [20]. The sharpness values correspond well between iterations except for an overall offset due to the gradual deterioration of the sample and slight drift effects between sweeps. In Figure 5.2, the same focus range has been swept in alternating directions. Obviously, the relationship between focal length and sharpness is different for each sweep owing to the history of the lens.
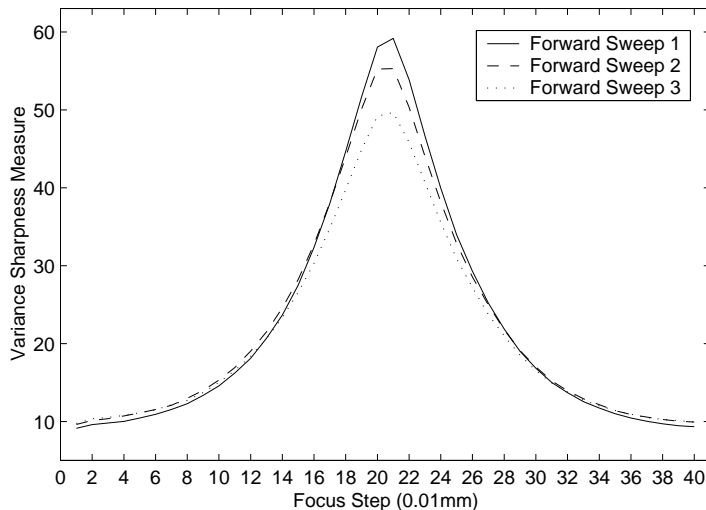

Figure 5.1: Hysteresis: Several Forward Sweeps

Hysteresis effects influence search algorithms in two ways: during the actual search and in returning to the best focus. Hysteresis is not a significant factor during a fixed stepsize search or a variable stepsize search, since these searches
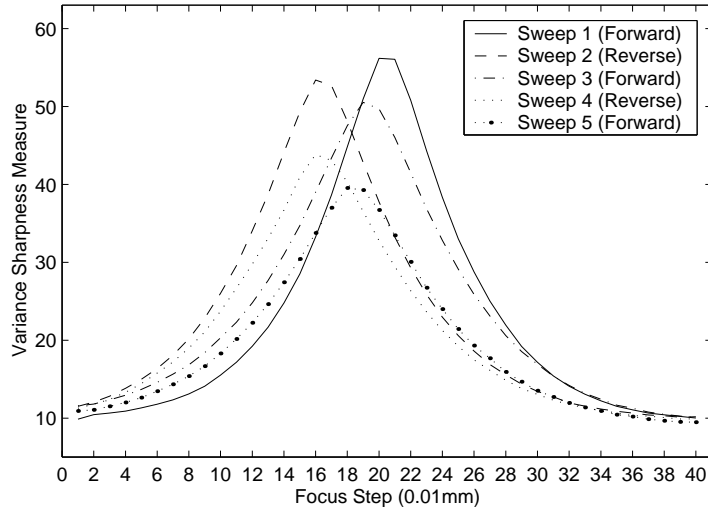
Figure 5.2: Hysteresis: Sweeps in Both Directions

only move the lens in a single direction. Hysteresis will play an important part between sweeps in the iterative search and during the Fibonacci search. More detail will be provided in the description of each search algorithm. Throughout this chapter the term 'online' will refer to running a search algorithm in real-time on the SEM. The term 'offline' will refer to running a search algorithm on a previously saved set of variance data, and thus offline searches will avoid hysteresis effects.

Almost all searches, including traditional fixed stepsize sweeps, will require returning the focus to the best focus. There is little reference in the literature to a practical method for resetting the instrument to the best focus following a focus sweep. An inefficient approach would be to clear hysteresis and then repeat the focus sweep with the exact same values, stopping at the desired best focus. This work uses a more efficient approach. It was observed that the offset due to hysteresis was relatively constant near the maximum sharpness. The following sequence of steps use this concept to compensate for hysteresis effects when returning the instrument to best focus (see Figure 5.3):

1. Perform focus search and determine best focus (f_{best1}) as well as the sharpness of the image obtained at best focus ($s_{desired}$).

2. Set the instrument to f_{best1} without any preliminary hysteresis removal efforts, and determine sharpness of corresponding image (s_{actual}).

3. If $|s_{desired} - s_{actual}| < \Theta$ where Θ is a given threshold, then finished. Otherwise determine the focus value (f_{best2}) which corresponds to s_{actual} from the initial focus sweep, using interpolation as necessary.

Section 5.1: Preliminary Considerations

4. Determine the offset $d = f_{best1} - f_{best2}$, and set the instrument to $f_{final} = f_{best1} + d$.
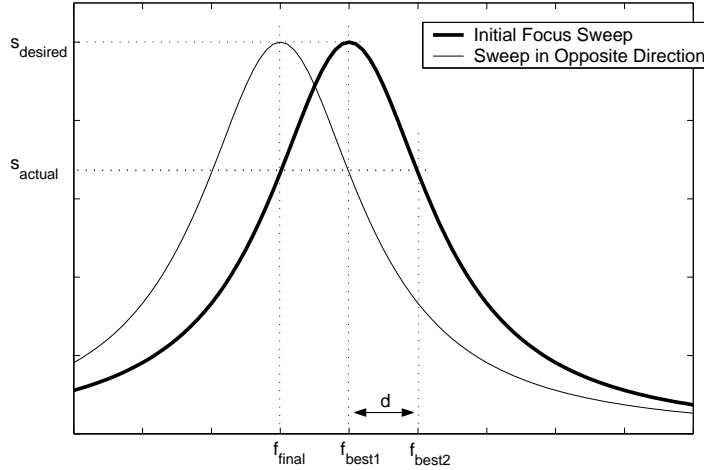
Figure 5.3: Hysteresis Adjustment Algorithm

This algorithm was tested with several of the search algorithms and worked reasonably well. If a hysteresis adjustment was deemed necessary ($|s_{desired} - s_{actual}| > \Theta$), then the image at f_{final} was almost always sharper than the image at f_{best1}.

5.1.2 Accuracy and Distance to Tails

The region of uncertainty is the focus interval in which the best focus is known to lie. A search algorithm is successful if it can narrow the region of uncertainty such that the image will appear to be equally sharp throughout this region. Or in other words, if the region of uncertainty is less than the depth of field. The accuracy of a search algorithm is simply the smallest region of uncertainty it can achieve. The distance to the tails is defined as the distance in millimeters between the best focus and where the monotonicity property breaks down. The desired accuracy and the distance to the tails play an important part in any search algorithm and are usually heavily dependent on both the specimen and the level of magnification. A general method was developed that allows these two intervals to be approximated from a given level of magnification. Figure 5.4 illustrates the symbols to be used in this method.
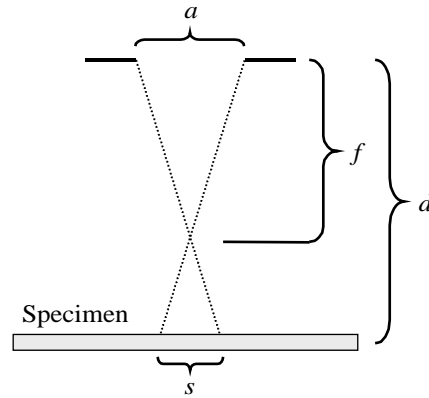
Figure 5.4: Calculating Accuracy and the Distance to Tails - a is the aperture diameter, f is the focal length, d is the actual working distance, and s is the spot size.

The focal length, working distance, aperture diameter, and spot size are related by the following equation.

$$\frac{a}{s} = \frac{f}{|d - f|} \tag{5.1}$$

The accuracy α will now be derived from Equation 5.1. Half the accuracy is the amount of defocus necessary before the spot size exceeds the pixel width p, causing the image to become slightly blurred. Substituting $\frac{\alpha}{2}$ for $|d - f|$ and p for s and then solving for α yields

$$\alpha = \frac{2pf}{a} \tag{5.2}$$

As mentioned above, the level of defocus can be written as

$$|d - f| = \frac{\alpha}{2} \tag{5.3}$$

Solving Equation 5.3 for f, substituting into Equation 5.2, and solving for α yields

$$\alpha = \frac{2pd}{a + p} \tag{5.4}$$

Equation 5.4 provides an approximation for the required accuracy at a given magnification. The working distance d is not known but can be approximated during a coarse focus sweep.

Realizing that the image will become completely blurred when the spot size exceeds the size of the largest feature in the image, we can use a similar derivation

for the distance to the tails. If k is the size of the largest feature in the image in pixels and d_{tails} is the distance to the tails then

$$d_{tails} = \frac{kpd}{a + kp} \qquad (5.5)$$

The feature size k cannot be determined without knowledge of the specimen. A satisfactory solution is to choose k such that it gives the lower limit of d_{tails}. For this work $k = 20$, and thus it is assumed that the specimen has at least one feature that is greater than 20 pixels. If this is true, then the monotonicity assumption will be valid over the interval: $best focus \pm d_{tails}$. Both Equations 5.4 and 5.5 assume a constant spot size across the specimen, a reasonable approximation for d, and beam linearity. Thus this technique is not expected to give better than order of magnitude approximations for α and d_{tails}. Nevertheless, this method was found to be a useful way to determine a reasonable level of accuracy and the distance to the tails for a given magnification.

5.2 Fixed Stepsize Search

A fixed stepsize search uses only a single sweep through the desired focus range, with a stepsize equal to one half the desired accuracy. We will use the minimum number of image captures required for a specific search algorithm at a given accuracy as a measure of its efficiency. If α is the desired accuracy, then the required number of image captures for an inclusive fixed stepsize search will be

$$N = \left\lceil \frac{2l}{\alpha} + 1 \right\rceil \qquad (5.6)$$

where l is the length of the interval to search. Fixed stepsize searches are useful for coarse focusing when the desired accuracy is low [19]. Examples of fixed stepsize sweeps can be seen in Figure 5.1.

An approach to further reduce N is to monitor the image sharpness during the sweep and stop the search once a peak in the sharpness measure has been detected. Ong's algorithm looks for changes in sharpness that exceed a given threshold, although he also notes that if this threshold is too low the sweep can stop prematurely [19]. This approach is more appropriate for sharpness measures that have a small range of monotonicity and very high accuracy. The variance sharpness measure has a much larger range of monotonicity and thus a threshold approach to peak detection is unlikely to be effective. An alternative was investigated that looks for an increasing variance value followed by two decreasing variance values. This method was easily trapped by local maxima due to noise,

so a moving average of the past three variance values was used instead. This approach worked for medium to high SNR images (see Figure 5.5).
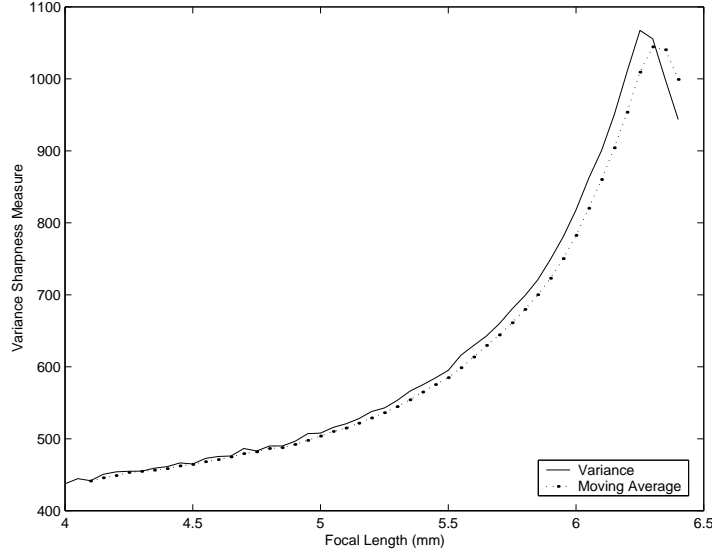
Figure 5.5: Peak Detection

5.3   Iterative Search

An iterative search is one of the most common, and it involves several sweeps with gradually smaller stepsizes [3, 19, 21].[1] After each iteration, the point of peak sharpness is determined and the region of uncertainty is narrowed. The next sweep searches over this smaller region. The number of image captures for each iteration is assumed to be constant. If the initial interval to search is l, then the accuracy for the first iteration will be

$$\alpha_1 = \frac{2}{\eta - 1}l \qquad (5.7)$$

where η is the number of image captures for each iteration. The next iteration will search over an interval of size α_1, and thus it can be seen that iteration i will search over an interval of size α_{i-1}. The accuracy of iteration i is thus

$$\alpha_i = \left(\frac{2}{\eta - 1}\right)^i l \qquad (5.8)$$

---

[1]Although magnification can be adjusted between iterations, this work assumes the magnification remains constant throughout a specific search.

Section 5.3: Iterative Search

Since η is constant, the total number of image captures N_i required at iteration i is simply ηi. Equation 5.8 can now be written as

$$\alpha_i = \left(\frac{2}{\eta - 1}\right)^{\frac{N_i}{\eta}} l \tag{5.9}$$

Solving for N_i yields the following for the total number of image captures required for the entire search at a final accuracy α

$$N = \left\lceil \eta \frac{\log\left(\frac{\alpha}{l}\right)}{\log\left(\frac{2}{\eta-1}\right)} \right\rceil \tag{5.10}$$

Note that since $i = \frac{N}{\eta}$, the number of iterations i may not be an integer. This simply means that if on any iteration $\alpha_i < \alpha$, then we set the accuracy of that sweep to α. In Figure 5.6, N is plotted as a function of η for an arbitrary α. The optimum value for η is 7.3, and thus the optimum integer value for η is 8.[2] Note that N may be slightly higher in practice than the theory indicates due to the specifics of how the implementation handles intervals that are not evenly divisible by the stepsize.
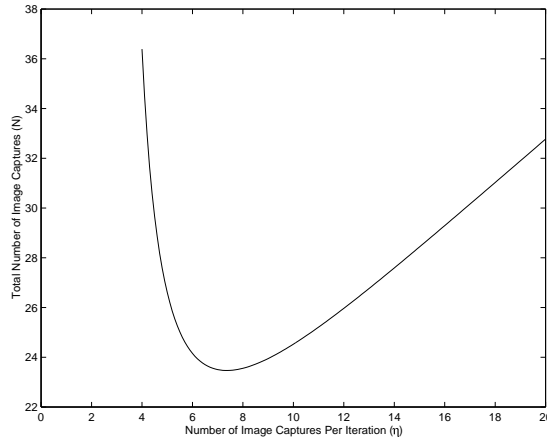
Figure 5.6: Image Captures Per Iteration vs. Total Image Captures

Figure 5.7 illustrates an online iterative focus sweep. Notice the slight discrepancy in variance measurements for the same focal length between iterations. This is due to hysteresis effects and setting the instrument to a small focal length prior to each iteration helped minimize this effect.

---

[2]Note that this differs from the optimum calculated by Beveridge and Schechter in [2] for two reasons: their derivation excludes the endpoints of each iteration and reuses overlapping data points between iterations. Hysteresis effects prevent us from using their approach.
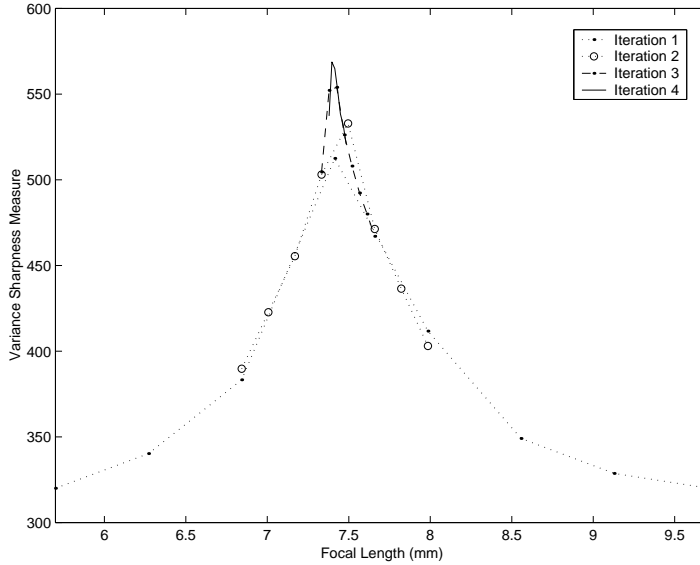
Figure 5.7: Online Iterative Focus Sweep ($\eta = 8$)

5.4 Variable Stepsize Search

A variable stepsize search is a common method used in optimization problems, but due to the noise inherent in SEM images it has yet to be explored for electron microscopy. The approach presented here attempts to decrease the stepsize as the variance increases until the stepsize achieves the desired accuracy. If the desired accuracy has not been achieved by the time a peak is detected, another fixed stepsize sweep is used over the interval of uncertainty with the desired accuracy. This approach is useful for real-time focusing where a smooth transition from a defocused image to a focused image is desired.

This section will first provide an overview of the method used for reducing the stepsize and then provide a more algorithmic description. The variable stepsize search is based on a moving average of three variance values $\overline{v}$, similar to that used for peak detection in Section 5.2. First, the change in the moving average ΔV is calculated between points 1 and 2 (see Figure 5.8). Then the variance at point 3 is calculated. If the variance is assumed to continue to increase linearly with the same slope as found between points 2 and 3, then the stepsize can be reduced to s_2 in order to keep the change in variance constant. To help increase noise robustness, the stepsize is constrained so that it never increases or becomes less than half the desired accuracy. Since the variance is *not* increasing linearly, this procedure will be repeated. In this way, the stepsize is decreased proportional to the change in variance. An algorithmic description is provided in Figure 5.9.

Section 5.4: Variable Stepsize Search
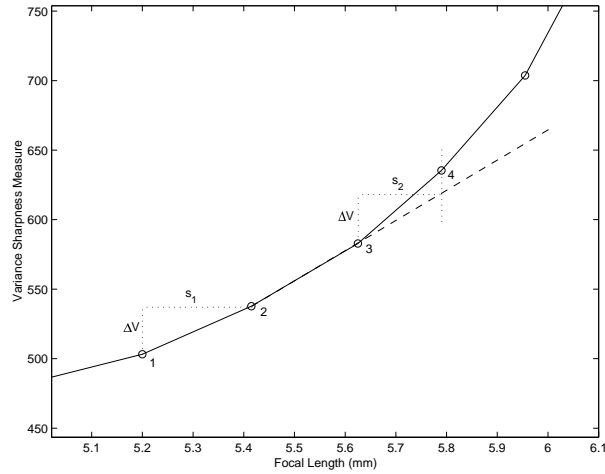
Figure 5.8: Illustration of the Variable Stepsize Algorithm

```
R   = false
f₁  = initial focus
s   = initial stepsize

capture four images at f₁, f₁ + s, f₁ + 2s, and f₁ + 3s
determine v̄₁ and v̄₂
f₃ = f₁ + 4s

for i = 3 to maximum number of steps allowed
   capture image and determine v̄ᵢ
   if not R and v̄ᵢ > v̄ᵢ₋₁ and sᵢ > α/2
      R = true
      ΔV = v̄ᵢ − v̄ᵢ₋₁
   else if R and v̄ᵢ − v̄ᵢ₋₁ > ΔV and sᵢ > α/2
      R = false
      s = ΔV (fᵢ − fᵢ₋₁)/(vᵢ − vᵢ₋₁)
   end if

   if peak has been found then break
   fᵢ₊₁ = fᵢ + sᵢ
end for

k = index of maximum variance
interval of uncertainty = f_{k−1} to f_{k+1}
```

Figure 5.9: Variable Stepsize Search Algorithm

Figure 5.10 illustrates the variable stepsize search algorithm operating at an accuracy of 0.1 mm. Notice that the change in variance stays approximately constant in (a) while the decrease in stepsize can be clearly seen in (b). The variable stepsize sweep required 15 images. Since the final interval of uncertainty is 0.2 mm and the desired accuracy is 0.1 mm, a fixed size sweep is necessary over the interval of uncertainty. Thus the total number of required image captures is 20.
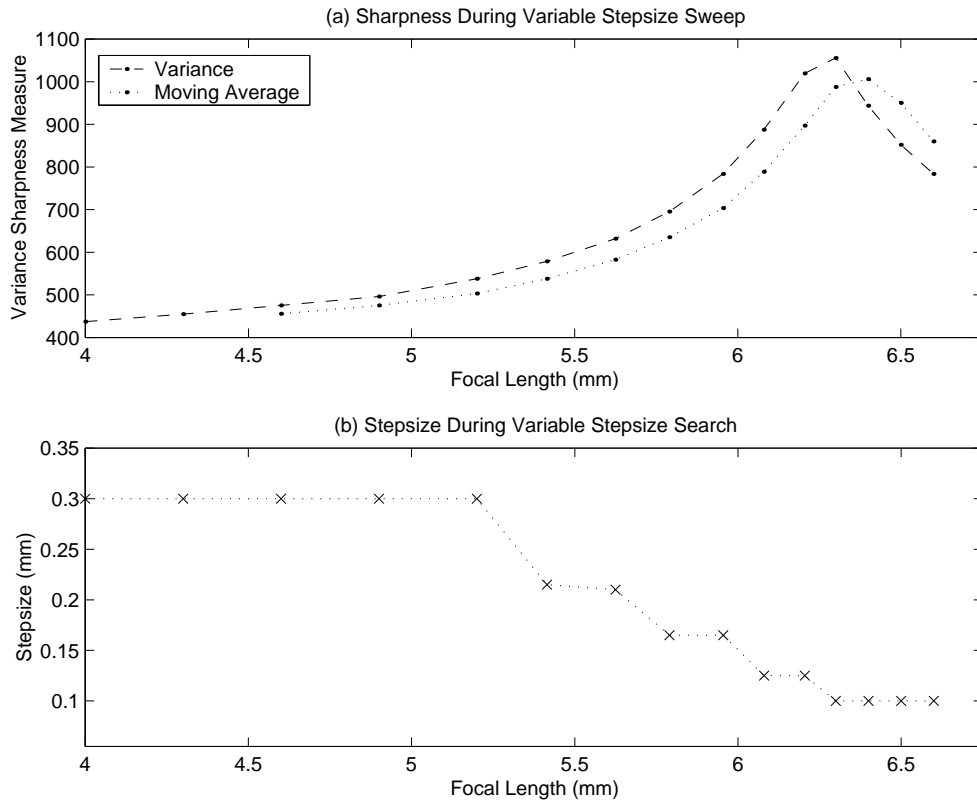
Figure 5.10: Offline Variable Stepsize Focus Sweep

A general solution for the total number of image captures is difficult to develop since the algorithm is heavily dependent on the initial stepsize and the specific variance curve. The total number of image captures N is the sum of the number of image captures required for the variable stepsize search N_v and the number of image captures required for the final fixed stepsize search N_f. Figure 5.11 shows N, N_v, and N_f for an offline variable stepsize search as a function of increasing accuracy. This variable stepsize search uses three times the desired accuracy as the initial stepsize, and the maximum number of image captures is set sufficiently high so that a peak is always found. Notice that for large desired accuracy values, N is dominated by N_f. This is because the initial stepsize is so large that the variable stepsize search quickly moves past the peak sharpness before it is able to

reduce the stepsize. As the accuracy value becomes smaller, N_v increases since the algorithm is able to reduce the stepsize before it reaches the peak. At very small accuracy values, the variable stepsize search is able to reduce the stepsize almost to the desired accuracy and so N_f is very small.

The initial stepsize is an important parameter that has considerable impact on the algorithm's effectiveness. If the initial stepsize is too small then the algorithm reduces to a fixed stepsize search with peak finding. If the initial stepsize is too large, then it is unlikely the stepsize will be small enough by the time we reach the peak. The ideal stepsize would be such that N_f is zero for the desired accuracy. It was found that three times the desired accuracy is a reasonable initial stepsize in most situations.
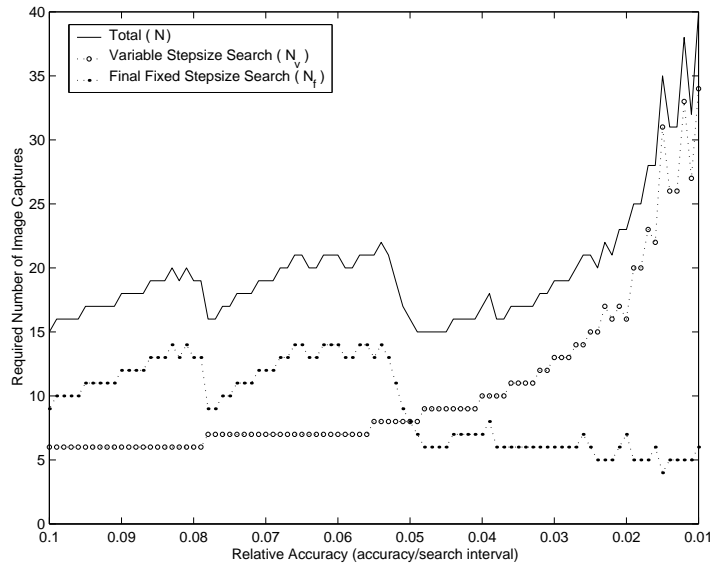
Figure 5.11: Number of Image Captures for Variable Stepsize Search

5.5   Fibonacci Search

Although the Fibonacci search has been used as a maximum sharpness search algorithm in optical microscopy [25], it seems to have been avoided in electron microscopy due to complications arising from hysteresis. This section will consider the Fibonacci search and a mechanism for handling hysteresis effects. This work will adopt the notation given in [2]. If $F_0 = 1$ and $F_1 = 1$ then for $n > 1$, the Fibonacci number F_n is defined as $F_{n-1} + F_{n-2}$.

The Fibonacci search is essentially an iterative search where $\eta = 1$, thus on

each iteration we will attempt to significantly reduce the region of uncertainty with a single image capture. Figure 5.12 illustrates a few example iterations of the Fibonacci search. First divide the search interval by half the desired accuracy, and then determine $M = F^{-1}\left(\frac{2l}{\alpha}\right)$ where $F^{-1}(x)$ is the order of the smallest Fibonacci number that is greater than or equal to x. For a search interval from a to b, determine the variance at $f_1 = a + \frac{F_{M-2}}{F_M}(b-a)$ (point 1 in Figure 5.12) and $f_2 = b - \frac{F_{M-2}}{F_M}(b-a)$ (point 2 in Figure 5.12). If point 1 is greater than point 2 then the maximum sharpness must be between a and point 2. If point 2 is greater than point 1 then the maximum sharpness must be between point 1 and b. A single additional variance measurement is now used to further reduce the region of uncertainty. The variance at point 3 (which is the same distance from b as point 2 is from point 1) is determined and compared to the previously obtained variance at point 2. By using Fibonacci numbers to partition the search interval, a single variance measurement can be compared to a previous measurement to reduce the region of uncertainty. These steps are repeated until the desired accuracy is achieved. A more comprehensive discussion of this and other optimization techniques can be found in [2]. See Figure 5.13 for an algorithmic description of the Fibonacci search.
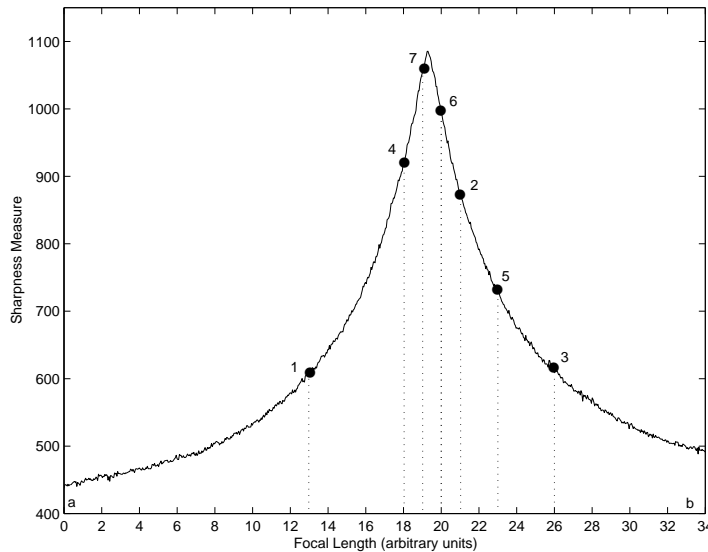
Figure 5.12: Fibonacci Search Example

Section 5.5: Fibonacci Search

```
a  = start of focus range
b  = end of focus range
α  = desired accuracy
```
$$M = F^{-1}\left(\frac{2(b-a)}{\alpha}\right)$$

$$f_1 = a + \frac{F_{M-2}}{F_M}(b-a)$$
$$f_2 = b - \frac{F_{M-2}}{F_M}(b-a)$$
```
v₁ = variance of image at f₁
v₂ = variance of image at f₂
```

```
for m = M - 1 down to 3
  if  v₁ < v₂  then
    a  = f₁
    f₁ = f₂
    v₁ = v₂
```
$$f_2 = b - \frac{F_{m-2}}{F_m}(b-a)$$
```
    v₂ = variance of image at f₂
  else
    b  = f₂
    f₂ = f₁
    v₂ = v₁
```
$$f_1 = a + \frac{F_{m-2}}{F_m}(b-a)$$
```
    v₁ = variance of image at f₁
  end if
end for
```

```
if  f₁ > f₂
  interval of uncertainty = f₁ - α/2  to  f₁ + α/2
else
  interval of uncertainty = f₂ - α/2  to  f₂ + α/2
end if
```

Figure 5.13: Fibonacci Search Algorithm [25]

Equation 5.11 gives the number of required image captures N for the Fibonacci search where $F^{-1}(x)$ is the order of the smallest Fibonacci number that is greater than or equal to x.[3]

$$N = F^{-1}\left(\frac{2l}{\alpha}\right) - 1 \tag{5.11}$$

Because the lens will be changing direction often, hysteresis effects will significantly influence the Fibonacci search. The Fibonacci search uses previous variance measurements in conjunction with the current variance measurement to reduce the region of uncertainty. Hysteresis can make using previous variance measurements unreliable. To investigate this, the image variance was measured for a set of random focal lengths. It was found that by setting the instrument to a small focal length prior to each random focal length, the monotonicity assumption could be preserved. Figure 5.14 shows a Fibonacci search and a fixed stepsize search for the same specimen and operating conditions. The Fibonacci search identifies a different best focus than a fixed stepsize search, but the final image is of similar sharpness. This will be a common approach when comparing search algorithms: we will rely on the sharpness of the best focus image instead of the absolute focal length, since two different search algorithms can find equally sharp images but at different focal lengths owing to hysteresis. An unexpected benefit of the Fibonacci search algorithm is that it is not necessary to use the hysteresis correction method described in Section 5.1.1. Once the search is finished the instrument is already at or near the best focus.

Note that the total number of required image captures given in Equation 5.11 is not completely accurate. A brief period of time will be required before each image capture to set the SEM to a small focus and then return it to the desired focal length. This overhead was found to be negligible when compared to the savings gained by using significantly less image captures than other search algorithms.

---

[3]This is slightly different from the results presented by Beveridge and Schechter in [2]. Their search algorithm uses a final dichotomous step to further reduce the region of uncertainty and thus $N = F^{-1}(\frac{l}{\alpha})$.
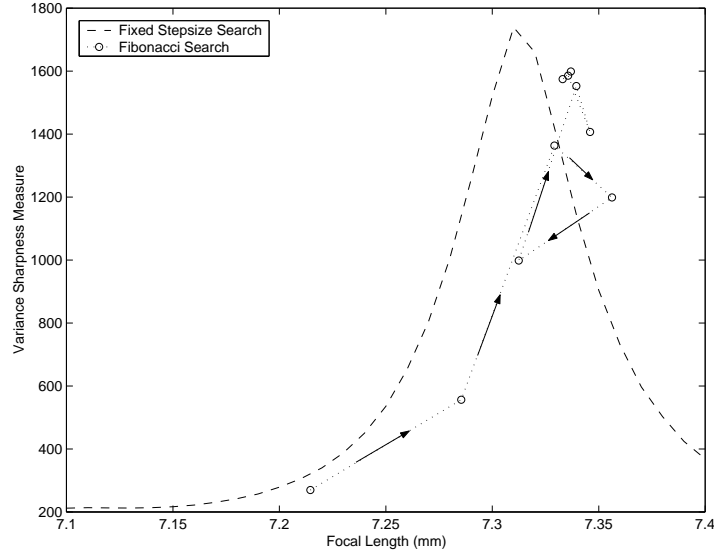
Figure 5.14: Online Fibonacci Sweep

5.6 Interpolation

Following a search algorithm, interpolation can be used to further improve the estimate of the best focus [2]. This also gives rise to the possibility of using a search algorithm with a large stepsize and then using interpolation to achieve the desired accuracy without the need for additional image captures. Quadratic interpolation was briefly considered, but most variance curves are poorly approximated by a quadratic function. This section will examine using a more appropriate approximation for interpolation.

Erasmus and Smith have developed a derivation for image variance as a function of defocus [6]. The general relationship for the non-astigmatic case is

$$v = \frac{\pi |F|^2 I^2}{\beta^2 z^2 + r^2} \tag{5.12}$$

where F is the Fourier transform of the image, I is the total beam current, β is the beam semiangle, z is the amount of defocus, and r is the spot size radius. This equation can be rewritten as a function of focal length f and several general parameters p.

$$v = \frac{1}{p_1(f - p_2)^2 + p_3} + p_4 \tag{5.13}$$

Parameter p_2 is the actual working distance and parameter p_4 is an offset due to the noise variance. This model makes three important assumptions: that the current distribution of the electron beam is gaussian, that the noise variance is constant with respect to defocus, and that the power spectrum of the specimen

is constant. The first two assumptions are acceptable for reasonable operating conditions, but the third assumption is only valid for a small subset of specimens. Even so, it was found that there was general agreement between this model and most specimens. Figure 5.15 shows the model given in Equation 5.13 fitted to the variance data from several specimens at several magnifications. Figure 5.15(c) shows that the model fits well over an interval where the monotonicity property is valid, while (d) shows that the model fits equally well over an interval with flat tails. The experimental variance data was observed to deviate from the model for specimens with a large depth of field or for specimens at low magnifications (as seen in (f)). At low magnifications, the image rotates during a focus sweep due to the electromagnetic properties of the lens system.
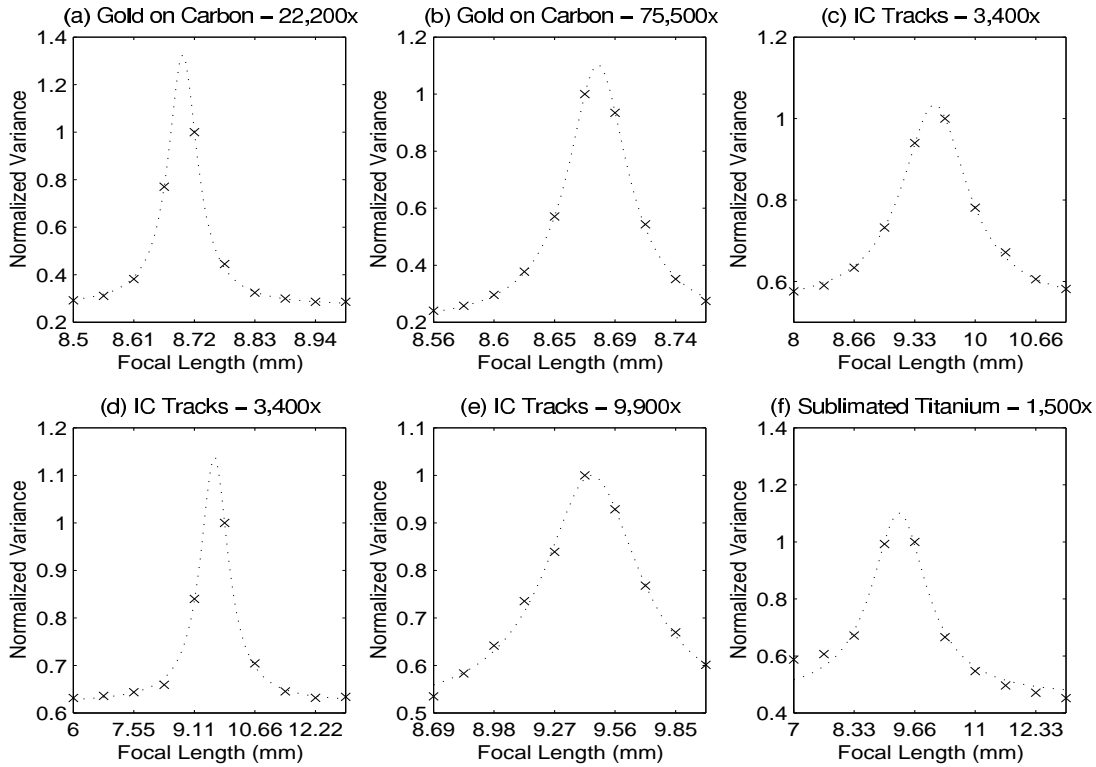
Figure 5.15: Variance Model for Various Specimens

To interpolate the best focus using this model, variance data is first gathered from a fixed stepsize sweep. Non-linear regression[4] is then used to fit the model in Equation 5.13 to the variance data, and then the best focus is simply given by parameter p_2. The dependence of this method on the number of steps in

---

[4]The Levenberg-Marquardt method was used for non-linear regression and is described in [14]. This method makes use of both a steepest-descent method and a Taylor series based method.

the fixed stepsize sweep was tested by first collecting variance data from a fixed stepsize sweep with a relative accuracy of 0.017. The model was then used offline to interpolate the best focus for various stepsizes. The results can be found in Figure 5.16.
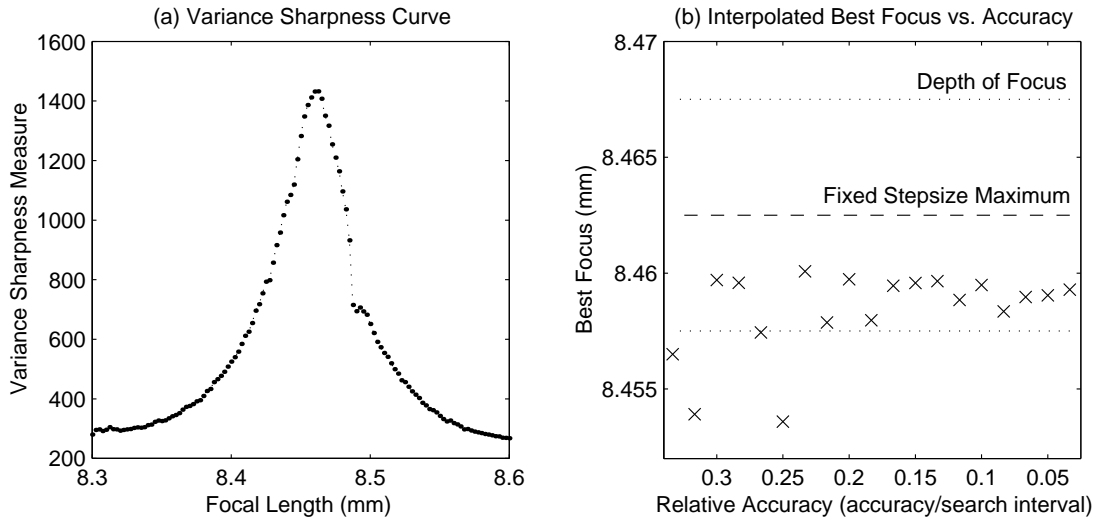
Figure 5.16: Interpolated Best Focus for Various Stepsizes

The fixed stepsize maximum in Figure 5.16(b) is the focal length associated with the maximum variance for the fixed stepsize sweep shown in (a). The depth of focus was qualitatively determined by examining the images collected from the sweep in (a). An experienced SEM operator determined the interval over which these images appeared to be equally sharp. Notice that the interpolated best focus becomes more consistent as the relative accuracy increases, and that at relative accuracies greater than 0.25 the interpolated best focus lies within the depth of focus. These results suggest that the model presented in Equation 5.13 is an effective way to interpolate the best focus from a fixed stepsize sweep. The overhead required for non-linear regression, however, will not be negligible and must also be considered.

5.7   Comparisons and Analysis

This chapter has presented four maximum sharpness search algorithms: the traditional fixed stepsize search, the iterative search, the variable stepsize search, and the Fibonacci search. The theoretical performance of these searches can be compared by examining the required number of image captures N for a given accuracy. As discussed in Section 5.4, the number of image captures required for a

variable stepsize search is dependent on specimen characteristics, magnification, and initial stepsize. Therefore Figure 5.17 shows the theoretical performance for each search, except for the variable stepsize search. Similar to Figure 5.11, the number of image captures required for two different offline variable stepsize searches are shown.
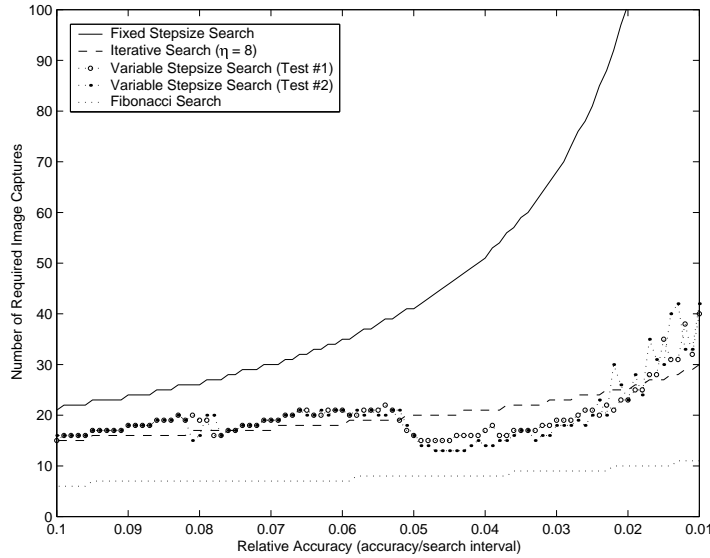
Figure 5.17: Number of Image Captures vs. Relative Accuracy

The Fibonacci search requires the minimum number of image captures, although each image capture requires a short period of time (approximately 200ms) to set the SEM to a small focal length. The traditional fixed stepsize search obviously requires the greatest number of image captures in order to achieve the same desired accuracy. To better examine the practical performance, the actual time required for each search method as a function of relative accuracy is shown in Figure 5.18.

Except for the variable stepsize search, these results are largely independent of the specimen and are only a function of desired accuracy. The interpolation search consisted of a standard fixed stepsize search at three times the desired accuracy followed by the nonlinear regression necessary to determine the best focus (see Section 5.6). Notice that the overhead of performing the nonlinear regression outweighs the reduced number of image captures. Using even less image captures for the interpolated search is a possibility, but as was seen in Figure 5.16, this can reduce the accuracy of the interpolated best focus. Figure 5.18 also shows that the time needed to set the instrument to a small focal length between each image capture in the Fibonacci search is negligible when compared to the other searches.

Section 5.7: Comparisons and Analysis

Figure 5.18: Practical Performance of Search Algorithms

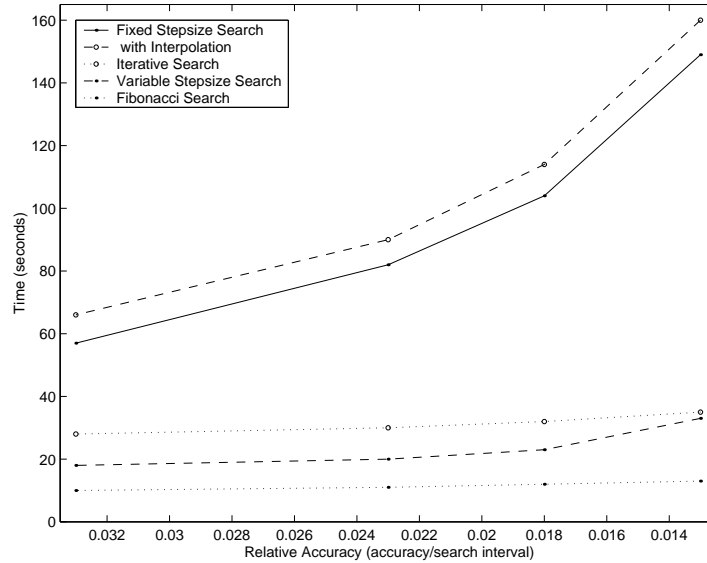Figures 5.17 and 5.18 are useful for comparing the required number of image captures and search time, but one would also like to qualitatively compare the accuracy of each search. Does each search actually select the best focus? Hysteresis effects make this question much more difficult to answer. Two different searches could identify the same best focus, but at different effective focal lengths owing to hysteresis. Thus Figure 5.19 compares the sharpness of the best focus chosen by each method. Figure 5.20 shows the specimens used in Figure 5.19.

The dashed horizontal lines in Figure 5.19 indicate the maximum sharpness found during the fixed stepsize search. Notice that the actual sharpness for the fixed stepsize search may be below this line due to hysteresis. The vertical arrows indicate where hysteresis adjustment was used to compensate for this effect (see Section 5.1.1 for details on the hysteresis adjustment algorithm, Θ was set to 50). The wide discrepancy in best focus sharpness for the titanium sample is due to the large depth of field in that sample as different search methods bring different parts of the image into focus. Notice the scale for each figure. The results in (b) and (d) are actually quite close to the maximum sharpness found during the fixed stepsize sweep. Following hysteresis adjustment, all searches generally performed well, although the variable stepsize sweep produced slightly inconsistent results.

Of the five search methods, the Fibonacci search appears to be a strong choice. It requires the fewest number of image captures, is the fastest when actually implemented, and produces sharp results. The main disadvantage of the Fibonacci search is its dependence on the monotonicity assumption. The iterative search also performs well and could be used to narrow the region of uncertainty until
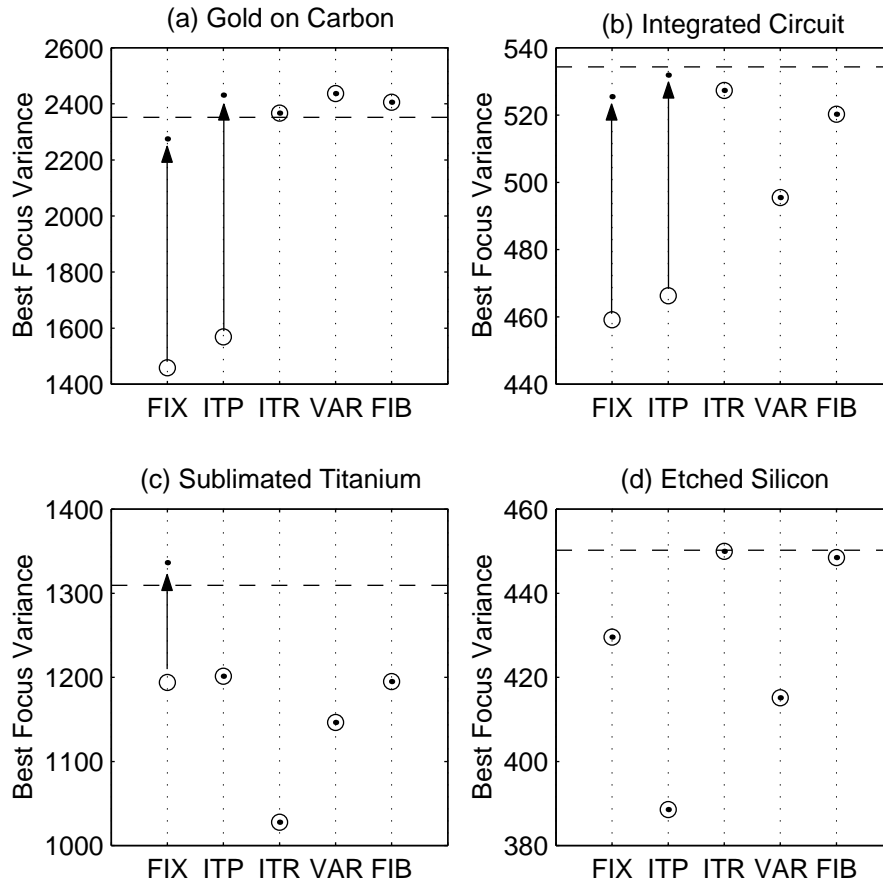
Figure 5.19: Best Focus Sharpness for Various Search Methods - The search methods used are fixed stepsize (FIX), interpolated (ITP), iterative (ITR), variable stepsize (VAR), and Fibonacci (FIB). The dashed horizontal line indicates the maximum sharpness found during the fixed stepsize sweep. The vertical arrows represent any necessary hysteresis adjustment.
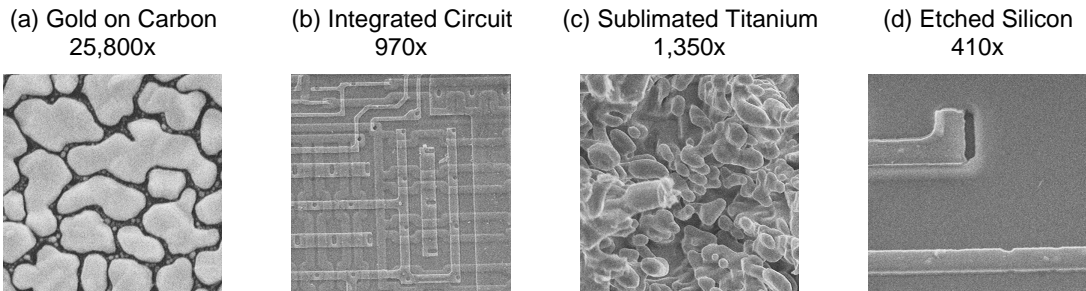
Figure 5.20: Specimens Used for Search Algorithm Tests

Section 5.7: Comparisons and Analysis

the monotonicity assumption becomes valid. It is important to note, although difficult to quantify, that the fixed stepsize and iterative searches will still work well at lower SNR levels while the variable stepsize and Fibonacci searches may not. This reveals a key concept mentioned earlier. The more sophisticated algorithms attempt to decrease the overall calculation time by using less images at a slightly higher SNR level. Qualitatively, it was found that the more sophisticated algorithms, especially the Fibonacci search, worked well even at lower SNR levels. At lower SNR levels the variable stepsize algorithm still worked, but had trouble appropriately reducing the stepsize.

Chapter 6

Additional Automated Functions

The work presented in the previous four chapters primarily concentrated on applying each area of the framework to focusing. This chapter will further apply this work to astigmatism correction, and also introduce beam alignment as an additional real-time automated operation.

6.1 Astigmatism Correction

The previous four chapters have assumed that the beam current distribution is relatively symmetric around the optical axis. Contamination in the column and especially in the spray aperture and final aperture can cause the beam to become asymmetric, resulting in a directional smear in the image. Compensation for an asymmetric beam is possible through use of the stigmators located within the optical column. Automatic astigmatism correction algorithms have been developed which use the directional information provided in the Fourier transform of an image. It has been observed that the directional smearing due to astigmatism in an under-focused image is rotated with respect to the directional smearing in an over-focused image. Several systems have been developed which rely on this information to determine the amount and direction of astigmatism correction [18, 19, 21].

An alternative approach was chosen for this work largely due to its simplicity and the relevance of the work presented in the previous four chapters. Erasmus and Smith have proposed sweeping both the x-stigmator and y-stigmator currents in turn and determining the image variance for each value [6]. The maximum of the variance curve then corresponds to the best beam shape. Figure 6.1 shows the experimental image variance as a function of stigmator values.
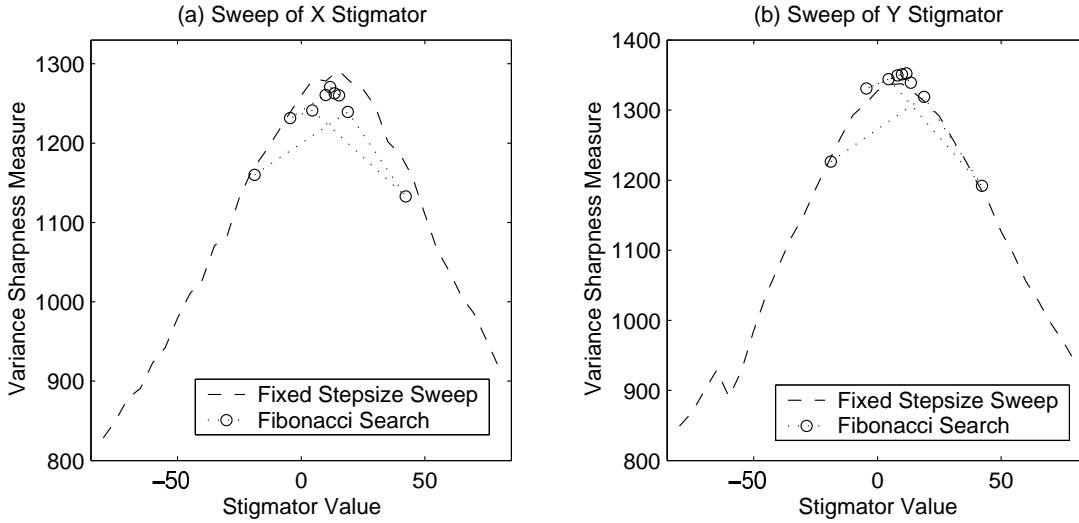
Figure 6.1: Online Astigmatism Sweeps

The framework developed in the previous chapters is readily applicable to this form of astigmatism correction algorithm. For example, since hysteresis effects are not present in the stigmators, the Fibonacci search is an ideal technique for finding the maximum variance. Figure 6.1 illustrates the use of a Fibonacci search to determine the best stigmator values. A general autofocusing and astigmatism correction algorithm would involve first finding the best focus, then determining the best x-stigmator value, and finally determining the best y-stigmator value. Depending upon the initial level of astigmatism, an additional focusing step may be necessary.

6.2   Beam Alignment

If the electron beam is not aligned with the electron-optical axis, then the number of electrons reaching the detector will be severely reduced and consequently the mean image intensity will be artificially low. Beam alignment is possible both physically and electromagnetically, although it is nearly impossible to achieve the required precision when using manual physical alignment. Therefore, most modern SEMs include additional electromagnetic coils responsible for either shifting or tilting the electron beam [3]. Alignment is usually required several times after the beam is initially turned on, and may also be necessary after changing operating conditions (particularly accelerating voltage). Gradually, the instrument stabilizes and alignment becomes less necessary. Beam alignment is an ideal candidate for real-time automation. Alignment should occur in the background without user interaction, so that the maximum number of electrons

are always reaching the detector.

Most auto-alignment algorithms perform a sweep of potential gun tilts and then determine which gun tilt results in the maximum image brightness [3]. This technique is inappropriate for real-time beam alignment since a gun tilt sweep can be very time consuming. Real-time beam alignment should ideally be both fast and transparent to the operator. For this work, beam alignment information was gathered from the emission image. Note that this technique is primarily applicable to microscopes using a LaB_6 source where the spot in the emission image is small and well defined. If the ideal spot location in the emission image is known, then one should theoretically be able to align the beam in a single step. Unfortunately switching to the emission image will obviously be noticed by the user. This raises a key concept in real-time automated operations: the need for a dual frame buffer image processing system.

The fundamental ideal behind real-time operations such as beam alignment and autofocusing is to seamlessly perform these operations whenever needed without user interaction. This is hindered, however, by the fact that an image processing auxiliary system (such as the one presented here) can only access what the operator is seeing on the display. The key to real-time operations is an image processing system that can scan two frame buffers as needed. One frame buffer is always displayed to the operator, but the second frame buffer could be available only to the image processing system. Thus one could scan the specimen, display it to the operator, switch to emission mode, scan the emission image, and then return to scanning the specimen. The operator would simply see a slight delay (the time it takes to scan the emission image) between two frames. Such a technique is not limited to real-time beam alignment, and could also allow reduced regions of interest during real-time autofocusing.

Figure 6.2(a) shows a common LaB_6 emission image. The spot location was determined by first finding the maximum brightness in the emission image in each column and in each row. Thus the 2D emission image is reduced to two 1D arrays. These arrays were smoothed using a gaussian filter, and then the maximum value found for each array. The location of the maximum was considered to be the spot location in the emission image. The goal of beam alignment is to then move the spot center to a predetermined ideal location in the emission image. This ideal location can be found through a preliminary calibration step using the gun tilt sweep method mentioned above. Calibration is only occasionally necessary, although it is definitely required if the emitter is replaced. The algorithm for moving the spot location to the ideal location in the emission image is shown in Figure 6.3.

This algorithm first determines how far the spot is from the ideal location in pixels. A conversion factor from pixels to gun tilt units is used to determine

Section 6.2: Beam Alignment

(a) Standard Emission Image

(b) Deformed Emission Image
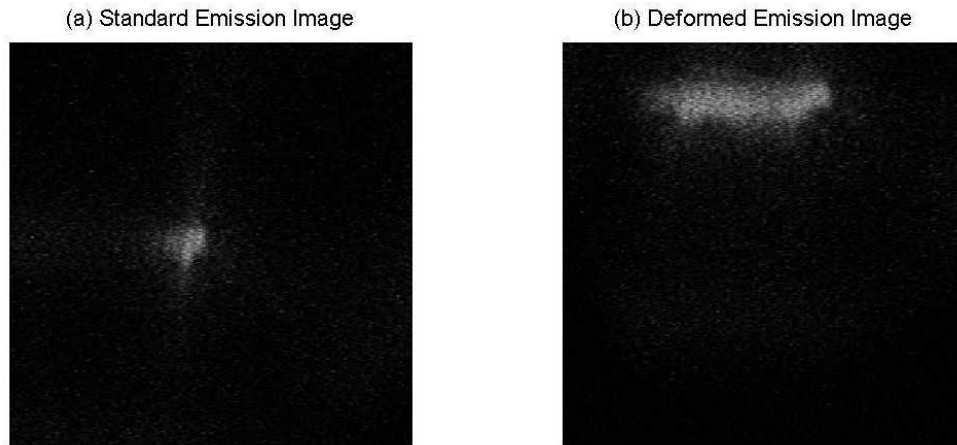
Figure 6.2: Example Emission Images

```
Θ   = threshold from ideal location
g   = current gun tilt
p_c = current spot location
p_p = previous spot location
p_i = ideal spot location
t   = initial gun tilt per pixel conversion factor

while |p_c − p_i| > Θ
  g = g + t|p_c − p_i|
  capture emission image and determine p_c
  t = t |(p_p − p_i)/(p_p − p_c)|
  p_p = p_c
end while
```

Figure 6.3: Beam Alignment Algorithm

how far to tilt the gun in order to align the beam. The gun is then tilted, a new emission image captured, and the new spot location determined. The ratio of the predicted distance the spot would move and the actual distance the spot did move is used to adjust the conversion factor for the next iteration. This process is continued until the spot location is within a predetermined threshold of the ideal spot location.

Two problems were encountered with this method of beam alignment: abnormal spot shape in the emission image and low electron yield at the center of the emission image. Figure 6.2(b) illustrates how the electromagnetic optics

can deform the spot shape at extreme gun tilts. The beam was reasonably well shaped for gun tilts near the ideal location, and therefore this problem was largely negligible. If the specimen has a low electron yield at the center of the image, then moving the beam to the center can severely darken the spot in the emission image. A potential solution is to notify the operator and perform a gun tilt sweep if the spot cannot be moved to the ideal location in the emission image after a predetermined number of iterations.

Section 6.2: Beam Alignment

Chapter 7

Implementation

The theory developed in the previous chapters provided the foundation for an autofocusing and astigmatism correction prototype implemented for the LEO 440 SEM. This chapter will discuss the development testbed used for preliminary testing, software coding, and final implementation. The actual implementation and the specific design decisions made for the fine focusing, full focusing, real-time focusing, and astigmatism correction algorithms will also be discussed.

7.1   The Development Testbed

A primary objective of this work was the establishment of a development testbed that allows for efficient testing and implementation of autofocusing and astigmatism correction algorithms. The testbed should enable a researcher to easily try new ideas and then develop these ideas into efficient implementations.

The instrument is manipulated through an ActiveX control supplied by the SEM manufacturer and known as the LEO API. This control provides a layer of abstraction between the developer and the actual instrument so that the developer can concentrate on the algorithms and not on the mechanics of instrument control. MATLAB was used for preliminary algorithm testing and development. The newer MATLAB releases support ActiveX, and should be able to use the LEO API ActiveX control. Unfortunately, a slight technical complication prevented MATLAB from being able to access certain functions within the LEO API. To solve this problem, an interface program was written that wraps around the LEO API and allows MATLAB to communicate with the instrument.

There are significant advantages to using MATLAB for preliminary testing

and development. MATLAB makes available sophisticated mathematical and image processing libraries that can allow a researcher to test new ideas quickly. MATLAB has extensive data visualization capabilities, and the graphical interface libraries can allow a developer to create an entire application completely from within MATLAB.

Once a researcher has tested a theory and developed an algorithm in MATLAB, it is then possible to port this work to a more efficient implementation. Visual C++ was chosen as the final implementation environment. Porting of MATLAB scripts was made easier by the MATLAB C++ math libraries. These libraries allow a C++ application to make use of standard MATLAB functions. Since Visual C++ applications can also act as ActiveX containers, the same LEO API control is used to communicate with the instrument. The MATLAB compiler can also aid in porting scripts to C, although this technique is limited since the compiler does not support the image processing toolbox.

A wide variety of scripts were written by the author to test the theories and ideas introduced in the previous chapters. These scripts allow for online and offline focus sweeps using various regions of interest, sharpness measures, and search algorithms. In addition, scripts were written for noise analysis, nonlinear interpolation, and beam alignment.

7.2 Prototype Description

COM technology, or more specifically ActiveX, is a powerful design strategy for developing modular and robust software. Vašina *et al.* have noted the advantages of using ActiveX to hide the complexities inherent in controlling a scanning electron microscope [22], and thus ActiveX will also be used for the implementation introduced in this section.

The prototype developed by the author includes both a graphical interface and an underlying image processing ActiveX control known as SEMimage (see Figure 7.1). SEMimage implements both lower level image processing functions as well as the higher level autofocusing and astigmatism correction algorithms. The Autofocuser graphical interface allows the user to control and monitor these algorithms. SEMimage interacts directly with the LEO API ActiveX control. The LEO32 software is the standard user interface supplied by the manufacturer. This modular approach can allow other applications to access SEMimage directly and thus make use of its automated operations. For example, SEMimage can be used easily by a performance benchmark utility or by XpertEze. XpertEze is largely written in Win-Prolog and is designed to make intelligent parameter decisions when forming an SEM image [3]. Because SEMimage is an ActiveX
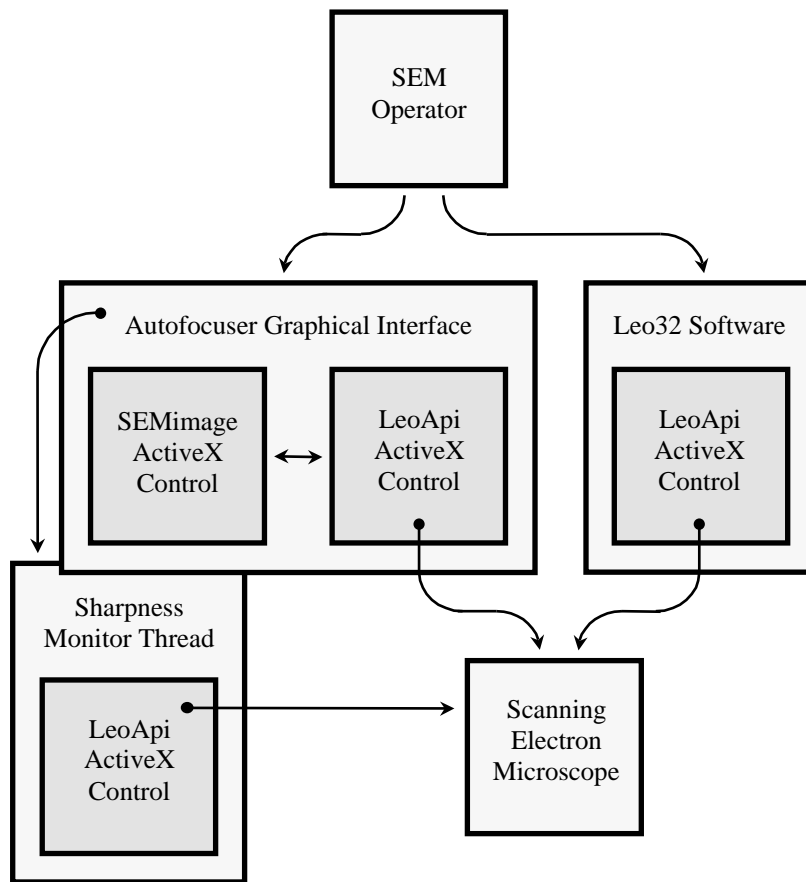
Figure 7.1: Implementation Hierarchy

Section 7.2: Prototype Description

control it can be used by any COM enabled container such as Visual C++, Visual Basic, and even MATLAB.

The implementation can monitor the image sharpness in order to determine when to perform real-time autofocusing. Continually monitoring sharpness is a computationally intensive task, and thus a multi-threaded implementation was necessary. The Autofocuser interface creates the Sharpness Monitor Thread, which then communicates directly with the SEM through its own LEO API control (see Figure 7.1). Section 7.2.3 will describe real-time autofocusing in further detail.

The graphical interface for the prototype is shown in Figure 7.2. The control buttons allow the operator to run an automated algorithm when desired, or the operator can initiate real-time operations through the Options button. Implemented real-time operations include beam alignment, fine autofocusing, and SNR measurement. The status flags indicate which real-time operations are enabled. Each of these operations are discussed in more detail below. Unless otherwise mentioned, all operations use reduced domain images of 512x512 pixels that are averaged over 10 frames at the fastest possible scan rate. The speed of all operations could be significantly improved with a faster SEM control computer[1], and possibly a smaller region of interest and lower SNR. The objective of the prototype, however, was both speed and robustness, and therefore more conservative parameters were chosen.
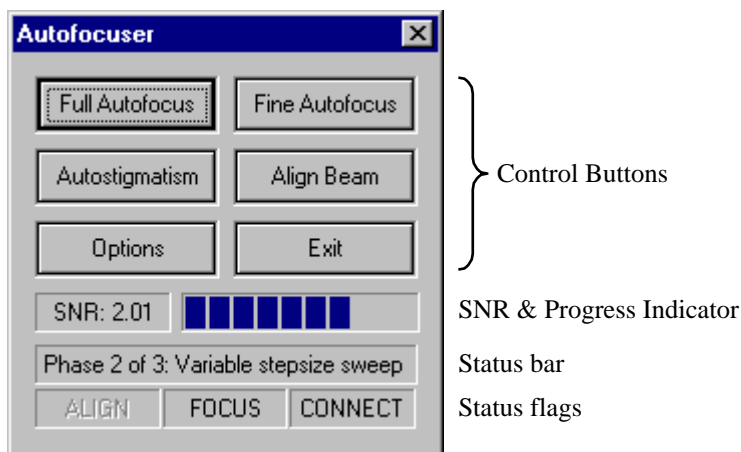

Figure 7.2: Graphical Interface for Prototype

---

[1]The LEO 440 SEM used in this work had a Pentium 200MHz processor with 48 megabytes of RAM

7.2.1   Full Autofocusing

Full autofocusing determines the best focus regardless of the initial focal length. This requires a preliminary coarse focus sweep at low magnification to determine a low accuracy region of uncertainty. SEMimage uses an accuracy of 2mm at 200x magnification for this coarse focus sweep. Following the coarse sweep, the desired accuracy and distance to tails are approximated using the method outlined in Section 5.1.2. If the distance to tails is greater than the accuracy of the coarse sweep, then the monotonicity assumption should be valid over the region of uncertainty and a Fibonacci search is used to find the best focus. Otherwise, an iterative search is used to further reduce the region of uncertainty until it is possible to use a Fibonacci search. The full autofocus algorithm is listed in Figure 7.3.

```
f_min   = lower limit for focal length
f_max   = upper limit for focal length
α_0     = accuracy for coarse focus sweep
α       = final desired accuracy

set to low magnification
f_0 = FixStepsizeSearch( f_min to f_max at α_0 )

set to original magnification
if ( d_tails > α_0 )
    f_best  = FibonacciSearch(f_0 − α_0/2 to f_0 + α_0/2 at α)
else
    f_itr   = IterativeSearch(f_0 − α_0/2 to f_0 + α_0/2 at d_tails)
    f_best  = FibonacciSearch(f_itr − d_tails/2 to f_itr + d_tails/2 at α)
end
set focal length to f_best
```

Figure 7.3: Full Autofocusing Algorithm

At very high magnifications, the stepsize for the first iteration of the iterative sweep can be so large that it completely misses any portion of the sharpness peak. To address this, the algorithm could be modified to include an additional sweep at an intermediate magnification after the coarse focus sweep. The time to perform full autofocusing varied from 30 seconds to over a minute depending on the desired magnification.

Section 7.2: Prototype Description

7.2.2   Fine Autofocusing

Fine autofocusing assumes that the initial amount of defocus is less than the distance to the tails. This means that the sharpness measure should be monotonically increasing between the initial focus and the best focus. Using this assumption, a preliminary test is made to determine whether the initial focus f_i is above or below the best focus. This test captures several images at slightly increasing focal lengths and then determines if the image variance is increasing or decreasing. SEMimage allows a user to choose between either a variable stepsize search or a Fibonacci search. The variable sweep uses an initial stepsize of three times the desired accuracy and moves in the direction indicated by the preliminary test. If the initial focal length is below focus, then the Fibonacci search uses a search interval of $f_i + 2d_{tails}$. If instead the initial focal length is above focus, then a search interval of $f_i - 2d_{tails}$ is used. The Fibonacci search worked well for specimens with features less than 100 pixels, but occasionally failed for specimens with larger features. For specimens with many large features, choosing $k = 20$ makes d_{tails} a poor approximation and thus the Fibonacci search interval is too small (see Section 5.1.2 for more on choosing k). The time for fine focusing using the variable stepsize search is usually around 30 seconds. Using the Fibonacci search decreases this time to a consistent 20 seconds.

7.2.3   Real-time Autofocusing

Real-time autofocusing enables the software to automatically determine when an image has become defocused and to then take appropriate action to move the image back into focus. The image variance was monitored in a separate thread to determine when the image becomes defocused. Once a moving average of the image variance exceeds a threshold, the sharpness is further monitored to determine when the sharpness has stabilized. The sharpness monitor thread then notifies the Autofocuser that the focus has changed and needs to be corrected. The Autofocuser uses the SEMimage component to move the instrument back into focus. A preliminary test similar to that described in the previous section determines the direction to search, and a variable stepsize search allows for a smooth transition between the defocused and focused images.

Real-time autofocusing was tested on several specimens at various magnifications and was found to be an effective method for automatically keeping the image in focus. The stage can be moved for a focused image, and the software will detect the change in sharpness and refocus the instrument. The variance threshold that determines when an image has changed sharpness is an important parameter. If this threshold is too low then the system will incorrectly initiate a

focusing operation, but if this threshold is too high then the system will fail to detect subtle changes in sharpness. In addition to defocus, image brightness and contrast can also influence the variance and cause false focusing operations. A more effective way to automatically determine when to initiate a focusing operation would be to monitor the position of the stage and other operating conditions. Unfortunately, the LEO 440 SEM available to the author uses a mechanical stage, and thus this technique could not be investigated further.

7.2.4 Astigmatism Correction

Astigmatism correction is performed by first finding the maximum sharpness for a sweep of the x-stigmator values and then repeating this procedure for the y-stigmator values. The operator can choose between a fixed stepsize search or a Fibonacci search. Astigmatism correction should be performed following a full or fine focus sweep and an additional fine focus sweep may be necessary following the correction. Astigmatism correction takes over a minute when when using a fixed stepsize search and only takes 19 seconds when using the Fibonacci search.

7.2.5 Other Real-time operations

The SEMimage component implemented two additional real-time operations: beam alignment and SNR calculation. Real-time beam alignment uses the algorithm described in Figure 6.3 at a set time interval. The time to perform beam alignment varies between 3 seconds and over 20 seconds depending on the number of required iterations. When the beam becomes stable, the component turns off real-time beam alignment. An auto-calibration function was implemented to determine the ideal spot location in in the emission image. This calibration function uses a sweep of the gun tilts to identify the emission image location which corresponds to maximum brightness in the image. The SNR is calculated using the method given in Section 2.2 and is performed at a user defined time interval.

Section 7.2: Prototype Description

Chapter 8

Conclusions

This thesis has investigated both the theoretical and practical aspects of autofocusing and astigmatism correction for scanning electron microscopy. A fundamental achievement of this work was the development of a framework for algorithm development consisting of four primary areas of concern: noise reduction, regions of interest, sharpness measures, and maximum sharpness search algorithms. Key accomplishments were made in each of these areas. Reduced domain median filters were shown to help mitigate limited bandwidth distortion more efficiently than standard domain median filters, and the gradient was integrated with a spatial high frequency filter to create a new sharpness measure. The subblock method was presented and shown to be an effective technique to reduce the region of interest without sacrificing key features in the image. The variable stepsize search was introduced as a new maximum sharpness search algorithm, and the Fibonacci search was successfully applied to electron microscopy. Finally, a new method of interpolation was presented based on a model of variance as a function of defocus.

In addition to this theoretical work, significant practical contributions were made. A testbed was established that allowed for rapid testing in MATLAB and application development in Visual C++. This testbed was then used to implement a modular autofocusing and astigmatism correction prototype for the LEO 440 SEM. The prototype performs full focusing, fine focusing, astigmatism correction, beam alignment, and SNR calculation. Perhaps the most significant aspect of the prototype is the ability to perform real-time autofocusing. The system can determine when the instrument is becoming defocused and then smoothly bring the image back into focus.

The prototype includes two parts: a graphical interface and the underlying SEMimage ActiveX control. This software architecture allows other projects to

make use of the automated functions implemented in SEMimage. More specifically, the XpertEze project will be able to use SEMimage to aid in intelligently managing an SEM. Even if the XpertEze project does not use the specific implementation produced through this work, the development testbed allows future work to be similarly integrated.

There is, however, room for improvement. A more quantitative analysis of how the subblock method impacts performance is needed, and further development of the in-sweep subblock method would be valuable. The prototype is slower than expected, and it is suspected that performance could be improved through software optimization and a more efficient technique for capturing images. Several of the ideas developed early in this work were not implemented in the final prototype including reduced domain median filtering, the subblock method, and interpolation. Finally, additional focus sweeps at intermediate levels of magnification could improve the full focusing algorithm when working at high magnifications. Combined with a faster SEM control computer these improvements could significantly enhance the current prototype.

Bibliography

[1] Andrew Armit. LEO instruments - personal communication, July 2000.

[2] G.S. Beveridge and R.S. Schechter. *Optimization: Theory and Practice.* McGraw-Hill, New York, 1970.

[3] N.H.M. Caldwell. *Knowledge-based engineering for the scanning electron microscope.* PhD thesis, University of Cambridge, November 1998.

[4] Thomas H. Cormen, Charles E. Leiserson, and Ronald L. Rivest. *Introduction to Algorithms.* The MIT Press, 1992.

[5] J.P. Davey. Autofocus algorithms 1: Effect of astigmatism. LEO Instruments - Technical Memo, September 1992.

[6] S.J. Erasmus and K.C.A. Smith. An automatic focusing and astigmatism correction system for the SEM and CTEM. *Journal of Microscopy*, 127:185–199, August 1982.

[7] Stephen Justus Erasmus. *Real-time digital image processing in electron microscopy.* PhD thesis, University of Cambridge, August 1982.

[8] Lawrence Firestone, Kitty Cook, Kevin Culp, Neil Talsania, and Jr. Kendall Preston. Comparison of autofocus methods for automated microscopy. *Cytometry*, 12(3):195–206, 1991.

[9] J. Frank and L. Al-Ali. Signal-to-noise ratio of electron micrographs obtained by cross correlation. *Nature*, 256(5516):376–379, July 1975.

[10] Rafael C. Gonzalez and Paul Wintz. *Digital Image Processing.* Addison-Wesley Publishing Company, Reading, Massachusetts, 1977.

[11] Frans C.A. Groen, Ian T. Young, and Guido Ligthart. A comparison of different focus functions for use in autofocus algorithms. *Cytometry*, 6:81–91, 1985.

BIBLIOGRAPHY

[12] D.M. Holburn and K.C.A. Smith. Topographical analysis in the SEM using an automatic focusing technique. *Journal of Microscopy*, 127(1):93–103, July 1982.

[13] David Hubbard. LEO instruments - personal communication, July 2000.

[14] Daniel Hyams. User documentation for CurveExpert v1.3. Internet: http://www.ebicom.net/~dhyams/cvxpt.htm, 1997.

[15] F.C. Nicolls, G. de Jager, and B.T. Sewell. Use of a general imaging model to achieve predictive autofocus in the scanning electron microscope. *Ultramicroscopy*, 69(1):25–37, August 1997.

[16] Frederick Nicolls. The development of a predictive autofocus algorithm using a general image formation model. Master's thesis, University of Cape Town, November 1995.

[17] C.W. Oatley. *The Scanning Electron Microscope, Part 1: The Instrument*. Cambridge University Press, Cambridge, 1972.

[18] Munehiro Ogasawara, Yuji Fukudome, Kiyoshi Hattori, Shuichi Tamamushi, Soshi Koikari, and Kazunori Onoguchi. Automatic focusing and astigmatism correction method based on fourier transform of scanning electron microscope images. *Japanese Journal of Applied Physics*, 38(2A):957–960, February 1999.

[19] Kok Hua Ong. A robust automatic focusing and astigmatism correction method for the scanning electron microscope. Master's thesis, National University of Singapore, 1998.

[20] Elizabeth M. Slater and Henry S. Slater. *Light and Electron Microscopy*. Cambridge University Press, Cambridge, 1992.

[21] W.J. Tee, K.C.A Smith, and D.M. Holburn. An automatic focusing and stigmating system for the SEM. *Journal of Physics E: Scientific Instrumentation*, 12:35–38, 1979.

[22] Jan Vašina and Radovan Vašina. COM technology based modular software for scientific instrument control. In *12th European Congress on Electron Microscopy Proceedings*, volume III: Instrumentation and Methodology, pages 1417–1418, 1998.

[23] András E. Vladár, Michael T. Postek, and Mark P. Davidson. Image sharpness measurement in scanning electron microscopy - Part II. *Scanning*, 20:24–34, 1998.

[24] Oliver C. Wells. *Scanning Electron Microscopy.* McGraw-Hill Book Company, New York, 1974.

[25] T.T.E. Yeo, S.H. Ong, Jayasooriah, and R. Sinniah. Autofocusing for tissue microscopy. *Image and Vision Computing*, 11(10):629–639, December 1993.