THE EFFECT OF NONLINEARITY IN ROBOTIC VISION SYSTEMS

by

RUPEN CHANDA, B.E.

A THESIS

IN

ELECTRICAL ENGINEERING

Submitted to the Graduate Faculty
of Texas Tech University in
Partial Fulfillment of
the Requirements for
the Degree of

MASTER OF SCIENCE

IN

ELECTRICAL ENGINEERING

Approved

August, 2000

ACKNOWLEDGEMENTS

I would like to thank all who had direct or indirect role in this project. I would like to first thank Dr. Sunanda Mitra for giving me this opportunity and support. This project would not have been completed without her advocacy. I would like to thank Dr. Micheal Eugene Parten for his continuous support throughout my master's degree. Also among the professorial staff, I would like to thank Dr. Donald Joseph Bagert for being on my committee and for helping me in completing my master's thesis.

My sincere thanks to Reginald Hunter, Member Technical Staff, Applied Materials, Inc., for providing me this opportunity and support to complete this project. My special thanks to Don T. Batson, P.E., Applied Materials, Inc., for his continuous guidance and assistance, without which this project would not have been what it is now. I would like to thank Eduardo O. Marquis and Shelly Zeigler of Applied Materials Inc., for their friendship and encouragement.

Among my friends and family, I owe special thanks to my parents for their support and patience. I would like to thank all of my friends, specially, Dmitri B. Abramov, for his moral support without which I would not been able to complete the project.

Finally, I thank everyone who has helped knowingly and unknowingly throughout my master's degree, and I wish best of luck for his or her future endeavors.

TABLE OF CONTENTS

ABSTRACT

The ultimate goal of image restoration is to improve the image in some sense. In order to reconstruct or recover an image, the specific degradation model is computed and inverse process is applied. Many types of degradations can be approximated by linear, position invariant processes. The advantage of this approach is the ability to use extensive theory of linear system theory. Nonlinear and space variant techniques, although more appropriate, introduce more complexities.

Normally, in industrial machine vision system line scan CCD camera is used for cost effectiveness instead of large-area CCD camera. But the image quality captured by a line scan camera depends on the motion stage (say, a robot or conveyer belt). If the velocity profile of the motion stage is nonlinear then the image is being distorted by the nonlinear function of motion stage. This problem can be dealt using nonlinear image restoration techniques. It is possible to remove the nonlinearity by a suitable inverse transformation model of the distortion function. But here geometrical image resampling and interpolation methods are used. This approach is more generic and cost-effective to implement in real life applications with little complexities. In geometrical resampling process output pixels are estimated by interpolation of input pixels. Bicubic interpolation of neighborhood pixel can be used for this purpose.

Cubic splines are especially practical for image pixel data interpolation because the set of equations, along with the two boundary conditions, are not only linear, but also tridiagonal. Therefore, the equations can be solved in O (N) operations (O is the number of operations) by the tridiagonal algorithm, which is faster and efficient.

LIST OF FIGURES

CHAPTER I

INTRODUCTION

In typical robotic vision systems, images of production parts moved by a robotic arm are captured while the robotic arm moves in a nonlinear manner thus introducing distortions in the captured images. The goal of this thesis is to remove such distortions induced by nonlinear robotic motion.

In industrial machine vision system based on line-scan cameras is very attractive for high-resolution applications. But if the motion stage of the application is nonlinear then the image captured by the line scan camera gets distorted by the nonlinear function of the motion stage (robot or conveyer belt).

Linear methods of image restoration have accounted for most of the practical applications of restoration to real-world problems. The advantage of this approach is that the extensive tools of linear system theory then become available for the solution of image restoration problems. Conversely, nonlinear, space variant methods usually require much more elaborate and costly computational procedures. In spite of these difficulties, nonlinear restoration methods should be considered for following reasons:

1. The capabilities of computers and numerical algorithms are ever increasing.
2. Nonlinear methods must be considered for comprehensiveness; many nonlinear problems result from a more concise and accurate description of the image restoration problem than with linear methods.

Nonlinear restoration methods of image restoration arise in several ways.

1. They may come about directly from the image formation and recording system. Common image recording systems, such as photographic film, are inherently nonlinear in response to image incident intensities. In our case, image is distorted directly when the image is being scanned by the line scan camera.

2. Nonlinear image restoration can arise by constraints. An image is formed from positive radiant energy components. There are methods, which do allow negative components in the restored solution. To produce a positive solution a constraint is required of the form $f \geq 0$. Inclusion of such a simple inequality constraints converts the basic linear solutions into nonlinear problems.

3. Finally, a nonlinear image restoration problem can arise by the way in which the problem is formulated. Formulation of the problem using descriptions--mathematical or probabilistic--that do not simplify to linear forms creates a nonlinear problem in image restoration. Typically such formulations arise from the use of optimization criteria that are more complex than quadratic or mean square criteria.

In this thesis, an application of generic approach to nonlinear image processing is described which is implemented in real-life application. Consideration is given to the general geometrical resampling process in which output pixels are estimated by interpolation of input pixels. However, here the geometrical resampling process is not at regular time interval. This generic approach can be used to solve several kinds of nonlinear image restoration problems when the image is being grabbed by a line scan

camera. With a little modification, this algorithm can be easily implemented in hardware, so that image can be corrected during the grabbing period.

As processing power increases and the cost of computing power decreases, complex and resolution-driven vision applications become more viable and cost effective. For this specific application whole pixel resolution if 1/4096 of the field of view was necessary. Cost effectiveness drove the application solution. An array of standard area camera was not preferred because of the space constraints and the software/calibration involved in handling overlapping fields of views at different scale and focus. A whole object image from one camera was deemed necessary. Large-area digital cameras with many dead pixels available, but the cost of the camera alone was $30,000 or more. Cameras with resolutions of 4000 pixels by 4000 pixels, with few defective pixels, cost in excess of $100,000. Instead, a velocity independent line scan imaging acquisition technique was implemented at a fraction of the cost of a large-area approach.

1.1 Overview of the Thesis Contents

The first chapter of this thesis gives an introduction of the thesis, overview of the thesis, and the importance and reason for nonlinear image processing.

The second chapter discusses the general scheme of the system; basic theories and characteristics of line scan camera, frame grabber, light source, and motion stage, which are typical to the particular system.

The third chapter is on nonlinear image restoration techniques. Here we are going to discuss different techniques in nonlinear image processing, linear regression and polynomial curve fit, and cubic spline interpolation. This gives an idea of the tools that

are used in implementations of the algorithms. This chapter also gives flowchart and step-by-step procedure to implement the algorithm.

The fourth chapter explains the results, constraints and critical points about the algorithm. The fifth chapter presents conclusions and suggestions for future improvements.

CHAPTER II

BASIC CONCEPT OF MACHINE VISION SYSTEMS

This chapter gives a brief introduction to all of the necessary components of a typical industrial machine vision system. This includes discussions on line scan camera, frame grabber, light source and motion stage.

2.1 Block Diagram of the System

A typical industrial machine vision system (Figure 2.1) comprises of the following parts, viz., master computer, vision card or frame grabber, line scan camera, and light source.

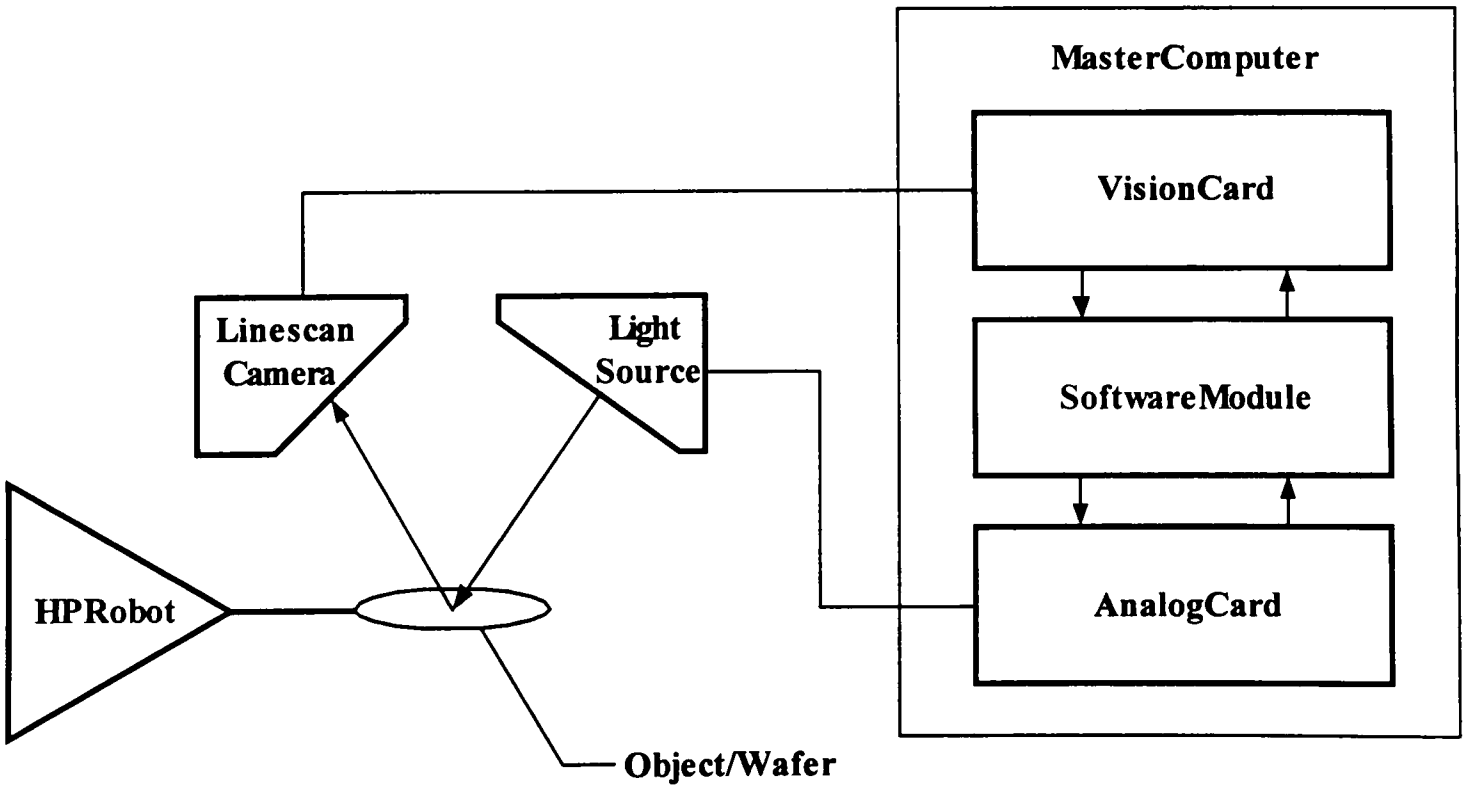

Figure 2.1: Simple Machine Vision System Block Diagram

In a simple machine vision system (Figure 2.1), the vision card/frame grabber acquires images from the line scan camera. The camera parameters are controlled either by the frame grabber or directly by the master computer via the serial communication (COM) port. The analog card is used to control the light source intensity. The vision software interacts with each card via their respective driver software.

2.2 Line Scan Camera

The line scan camera uses CCD-sensor chips with electronic exposure time control and anti-blooming features. The major components in the camera electronics include: a CCD sensor, an amplifier, and an ADC (analog-to-digital converter). The CCD sensor outputs a voltage signal when it is exposed to light. This voltage is amplified and transferred to the ADC, which converts it to a digital output signal.

The output signal of the sensor normally ranges from 0V when it has been exposed to no light to 0.7V when it has been exposed to bright light. Within that range, the sensor's characteristics are linear.

The exposure time is controlled via the external EXSYNC signal. The exposure time can be edge-controlled or level controlled, which means exposure time may be set to the full line period or be controlled by the EXSYNC signal. In these modes, a rising edge of EXSYNC triggers the readout of accumulated charges from the sensor elements to the CCD shift registers. The exposure time can also be programmed to a prefixed time period. In this case, accumulated charges are read out subsequent to the exposure time.

The accumulated charges are transported from the light-sensitive sensor elements to the CCD shift registers. The charges from even and odd pixels are processed separately in two channels (Figure 2.2). The charges then move from the two lines of shift registers

to the output amplifiers where they are converted to voltages proportional to the accumulated charges. The shift is clocked according to the camera's internal data rate.

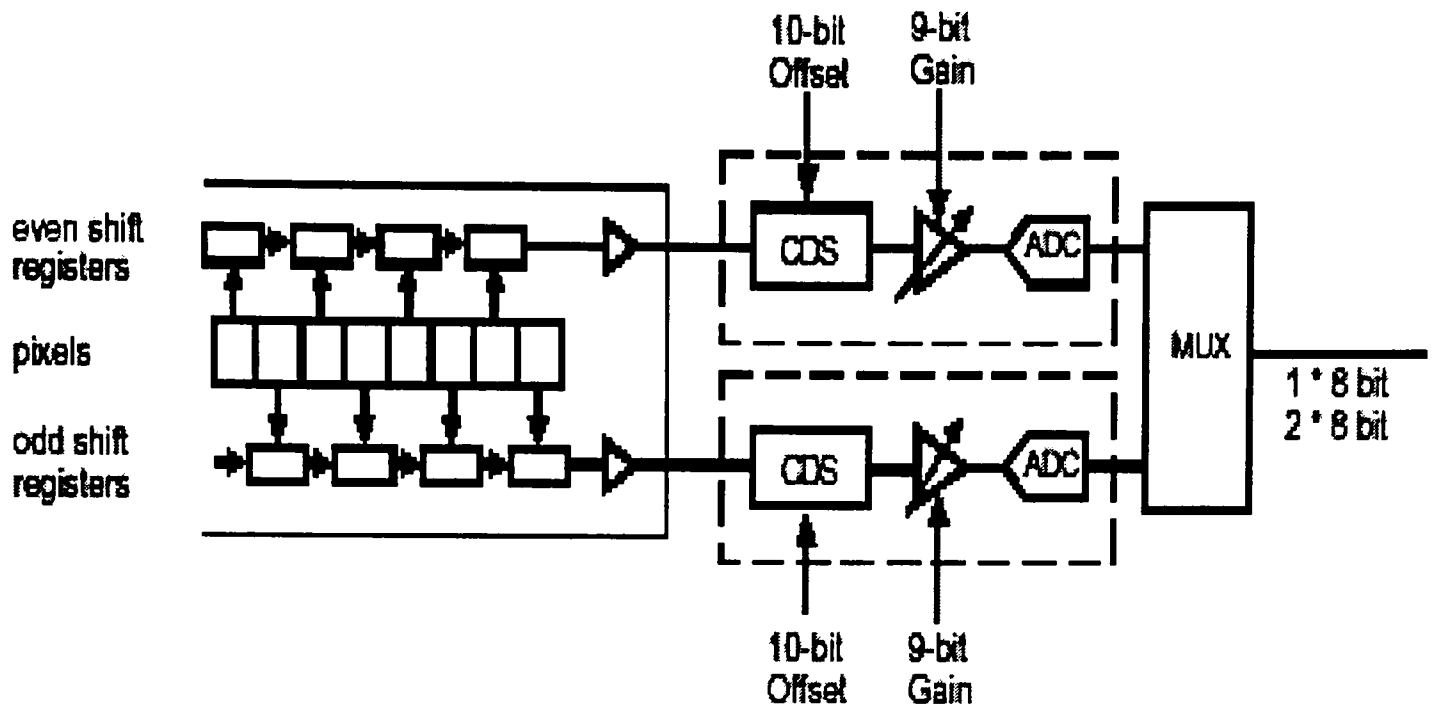

Figure 2.2: Camera data channel

The voltages are digitized and sent by the camera. The video data is transmitted either as a single (8 bit) or dual (2 * 8 bit) video data stream depending on the camera settings. All output signals use LVDS technology according to RS-644. For optimal digitization, gain and offset are programmable via the RS-232 serial port.

The camera's spectral responsivity is depicted in Figure 2.3.

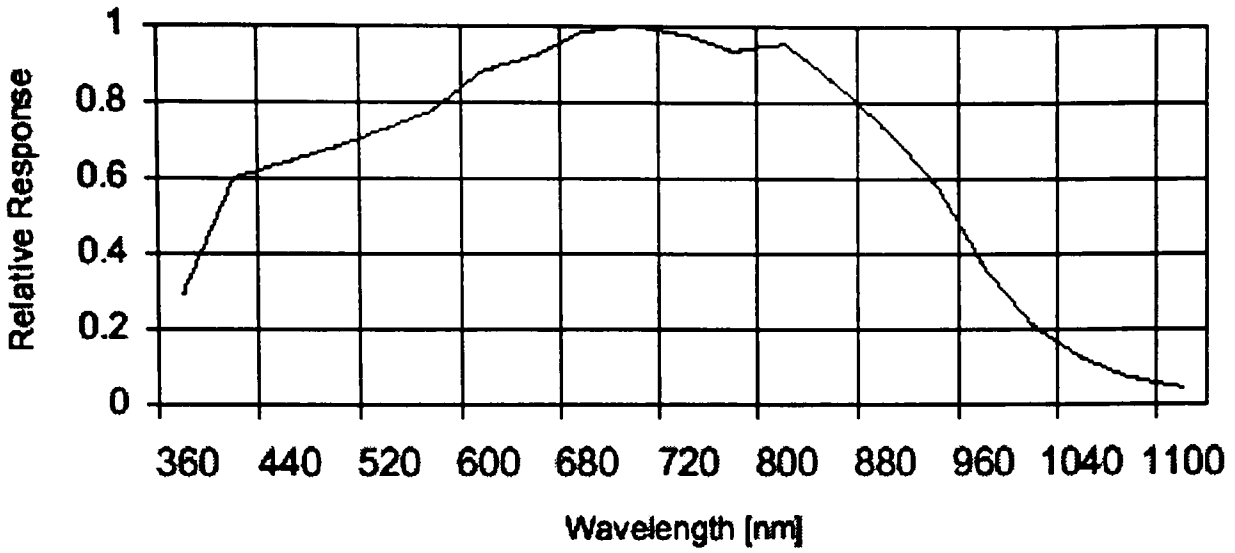

Figure 2.3: Spectral Responsivity

In free-run, no EXSYNC signal is required. The camera generates its own internally controlled sync signal and lines are transferred to the output of the camera automatically. In free-run edge-controlled mode, charge is accumulated over the full line period. The frame is read out and transferred at the end of the line period (Figure 2.4).

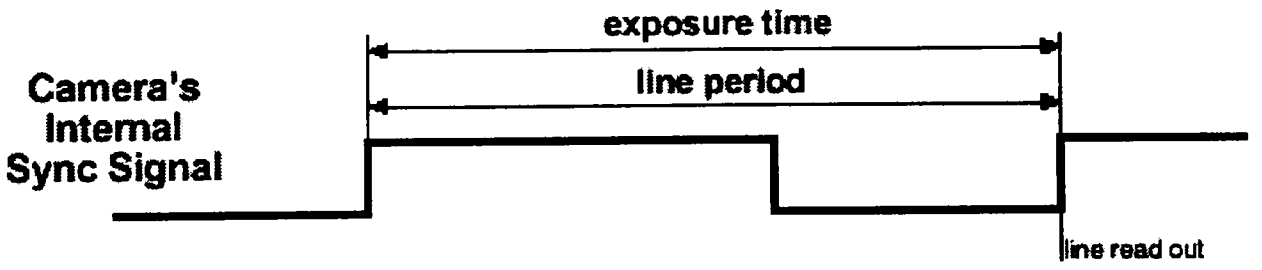

Figure 2.4: Exposure time free-run edge controlled mode

The line scan camera is programmable via RS-232 serial port. The data character format is 8N1 (8 data bits + no parity + 1 stop bit). Baud rate is 9600 bps.

2.3 Frame Grabber

Matrox Meteor-II/Digital is high performance PCI frame grabber based on Matrox Video Interface ASIC (VIA) (Figure 2.5). It is a sophisticated memory controller for managing real-time acquisition in to on-board memory with advance reformatting capabilities and, in parallel streaming image data out over the PCI bus without requiring constant host CPU intervention. It can transfer acquired images at sustained rates up to 130 MB/sec.

Matrox Meteor-II/Digital has a 32 bit interface configurable as 4 * 8 bit, 2 * 16 bit, or 1 * 32 bit, and supports RS-422 and LVDS digital signaling standards at up to 25 MHz and 40 MHz, respectively. It also supports image sizes of 64K pixels per line and 64K lines per frame.

The frame grabber features a LUT configurable as four 256 * 8-bit or two 4K * 16-bit. Also provided is a full complement of separate synchronization and control signals: pixel clock, hsync, vsync, timers, and auxiliary I/Os. A trigger input is present for synchronization acquisition to external events in a manner that is synchronous and asynchronous to the video input. Extensive buffering (4 MB) ensures reliable real-time transfer of incoming image data to host memory even under high bus latency conditions, which typically occurs in systems with concurrent image capture, display, graphics, network access, disk access, and general external I/O. The PCI interface supports image data sub-sampling from 2 to 16 (by decimation) to further reduce required PCI bus bandwidth.

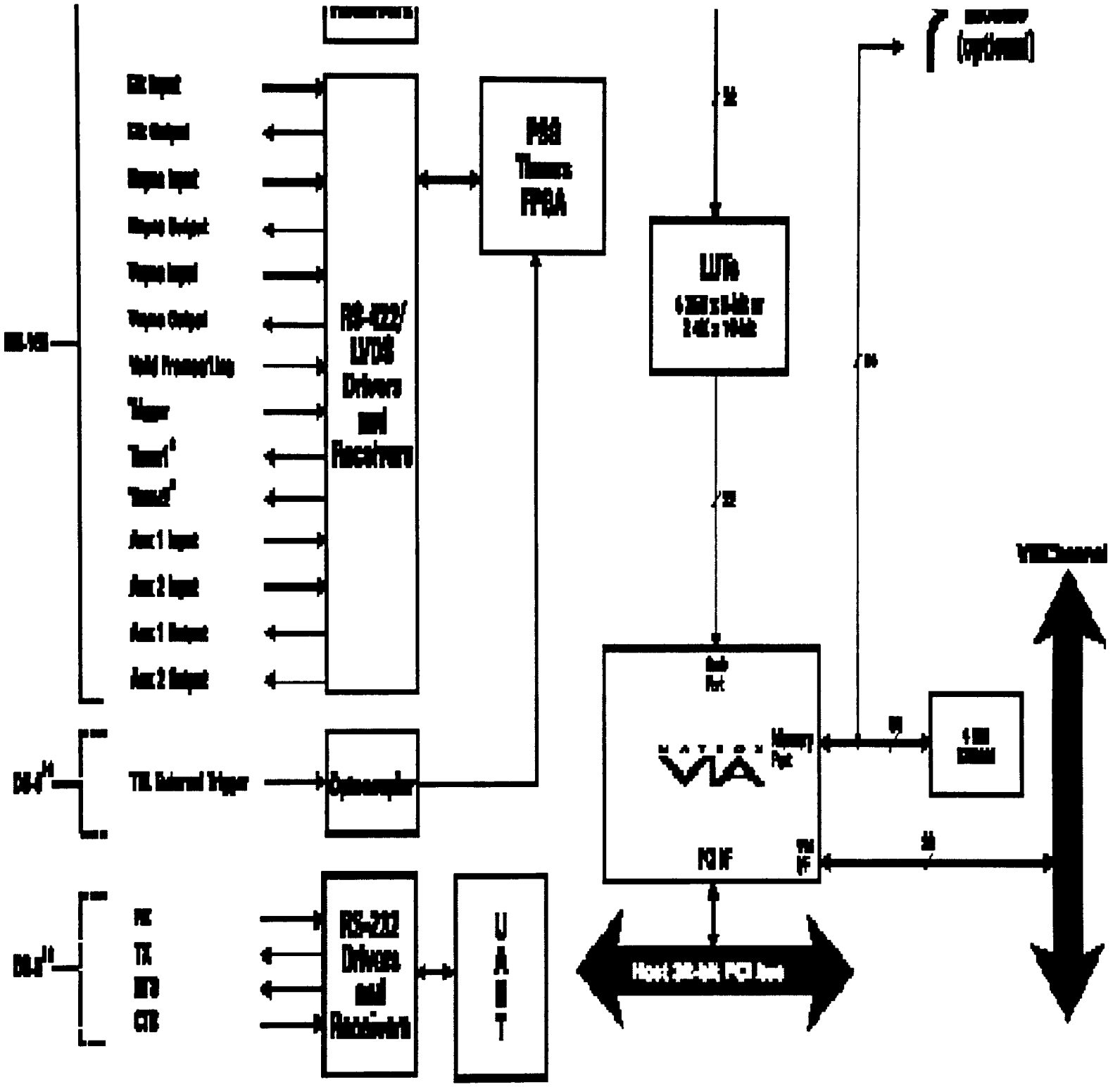

Figure 2.5: Block diagram of Matrox Meteor-II/Digital

2.4 Light Source

A CCD array integrates light, and then the gray level of each pixel is read off serially. For high-speed scanning, the active time on a line is of the order of tenths of a millisecond. The time allowed for light to be absorbed by the CCD is less than one-half the line time. Substantially more light is needed to implement this approach than is required by standard area imaging system. Because only one line is imaged at a time, only a strip coincident with the imaging area must be illuminated. The light source used to illuminate the object under inspection was tungsten halogen lamp based fiber-optic line light with a cylindrical lens to concentrate the optic's light output. For line scan acquisitions, light distribution is constant for each line acquired. The light source may be controlled manually form the front panel or remotely via the 9 pin remote interface connector on the rear of the unit. For optimum and stable lamp performance, lamp manufacturers recommend halogen lamps be operated above 80% of their rated intensity (Figure 2.6).

In the remote mode operation, the lamp is operated via an analog interface card by the master computer. The range of the external DC control voltage should be within 0.0V to 5.0V. It takes a cold lamp typically 25 minutes to stabilize within 1% or better. This stabilization period is slightly affected by the lamp voltage.

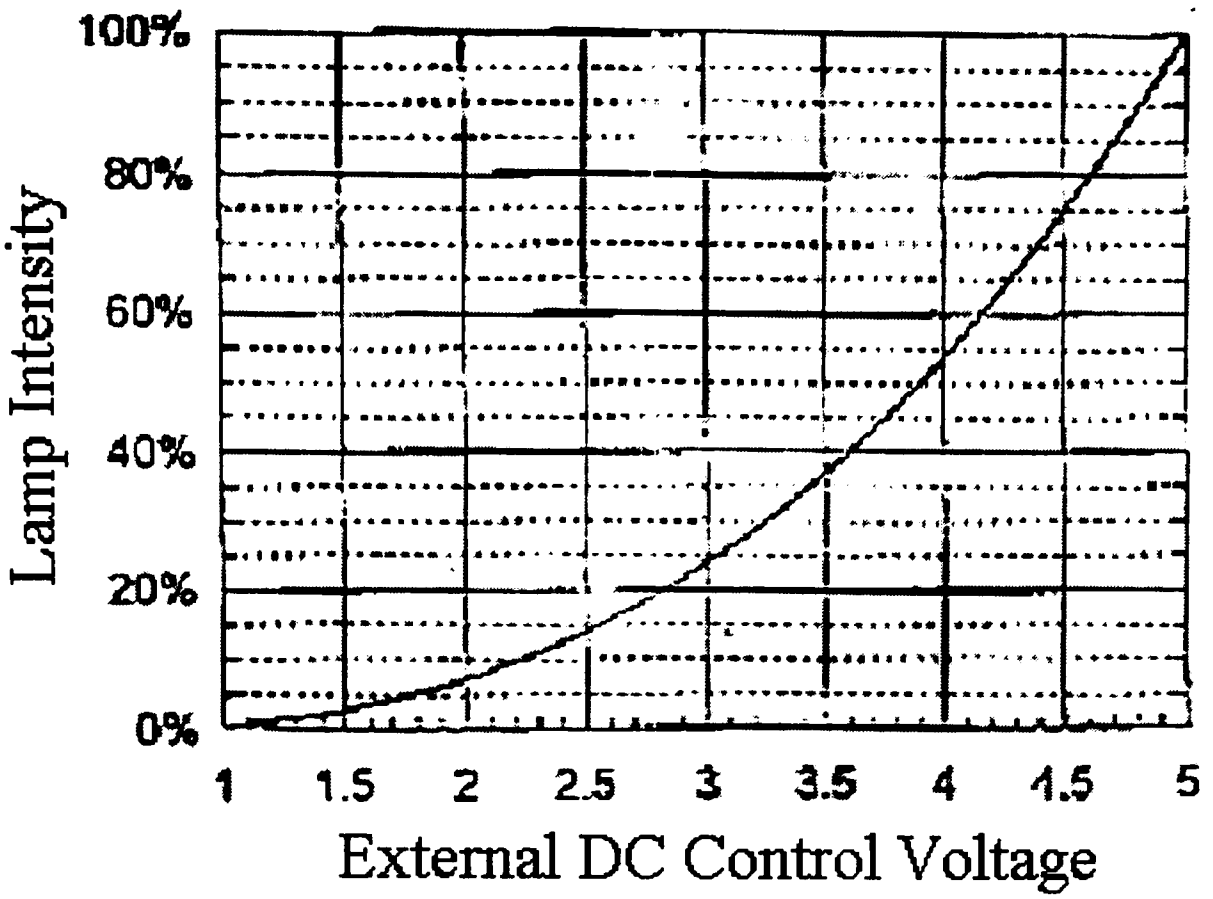

Figure 2.6: Lamp Intensity versus External DC Control Voltage

2.5 Motion Stage

The high-performance robot arm, which is used to transfer the object/wafer among the processing chambers, has nonlinear velocity profile. It is a stepper motor controlled robot arm (Figure 2.7). The position and the wing angle of the robot arm can be expressed as

$$\text{Position } X = D + A\cos\Theta + \sqrt{B^2 - (A\sin\Theta - C)^2} \tag{2.1}$$

$$\text{Wing Angle } \Theta = \tan^{-1}\left(\frac{C}{X-D}\right) + \cos^{-1}\left(\frac{A^2 + C^2 + (X-D)^2 - B^2}{2A\sqrt{C^2 + (X-D)^2}}\right) \tag{2.2}$$

For the machine of interests,

Wing length A = 10.050 inch = 255.27 mm.

Arm length B = 10.800 inch = 274.32 mm.

Wrist width C = 1.000 inch = 25.40 mm.

Blade length D = 5.100 inch = 129.54 mm.

Other mechanism constants are,

Steps per revolution = 100000.0 steps/rev.

Initial velocity = 1500.0 steps/sec.

Acceleration = 25000.0 steps/sec/sec.

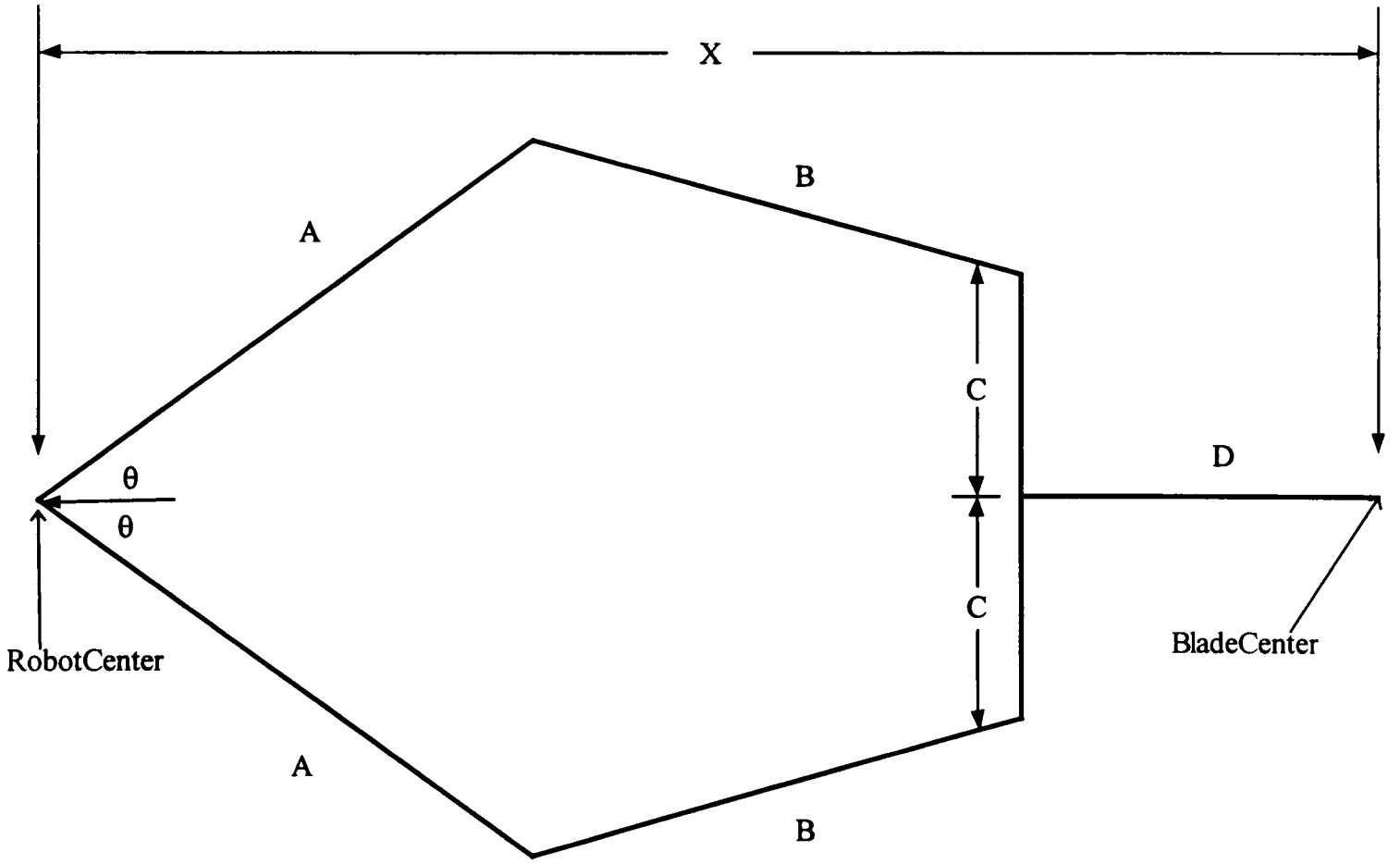

Figure 2.7: High Performance (HP) Robot

The plot of robot blade extension versus time (Figure 2.8) is derived from equation 2.1. In this case, line scan camera frame rate = 9000 lines/sec, blade acceleration = -25000steps/sec/sec. From the plot, it can be inferred that the robot gradually decelerates over time, which will cause unequal extension of the robot arm for equal time slice.

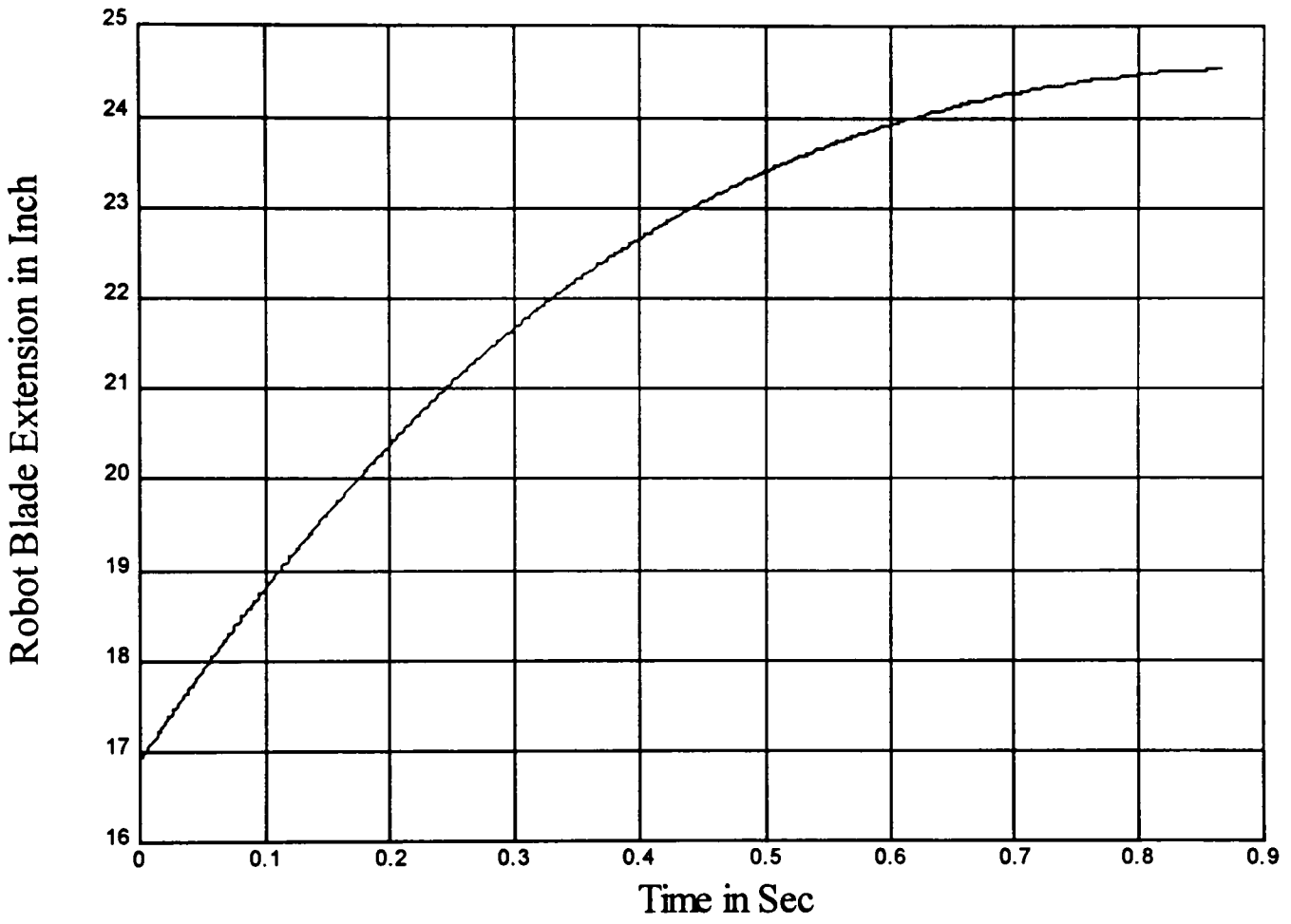

Figure 2.8: Blade Extension (Inch) versus Time (Sec)

The plot of blade movements between frames versus time is given by Figure 2.9. From this plot it can be seen that, the robot arm moves over unequal distance in between any two-line frames. The line scan camera frame rate = 9000 lines/sec, blade acceleration = -25000steps/sec/sec, remains the same.

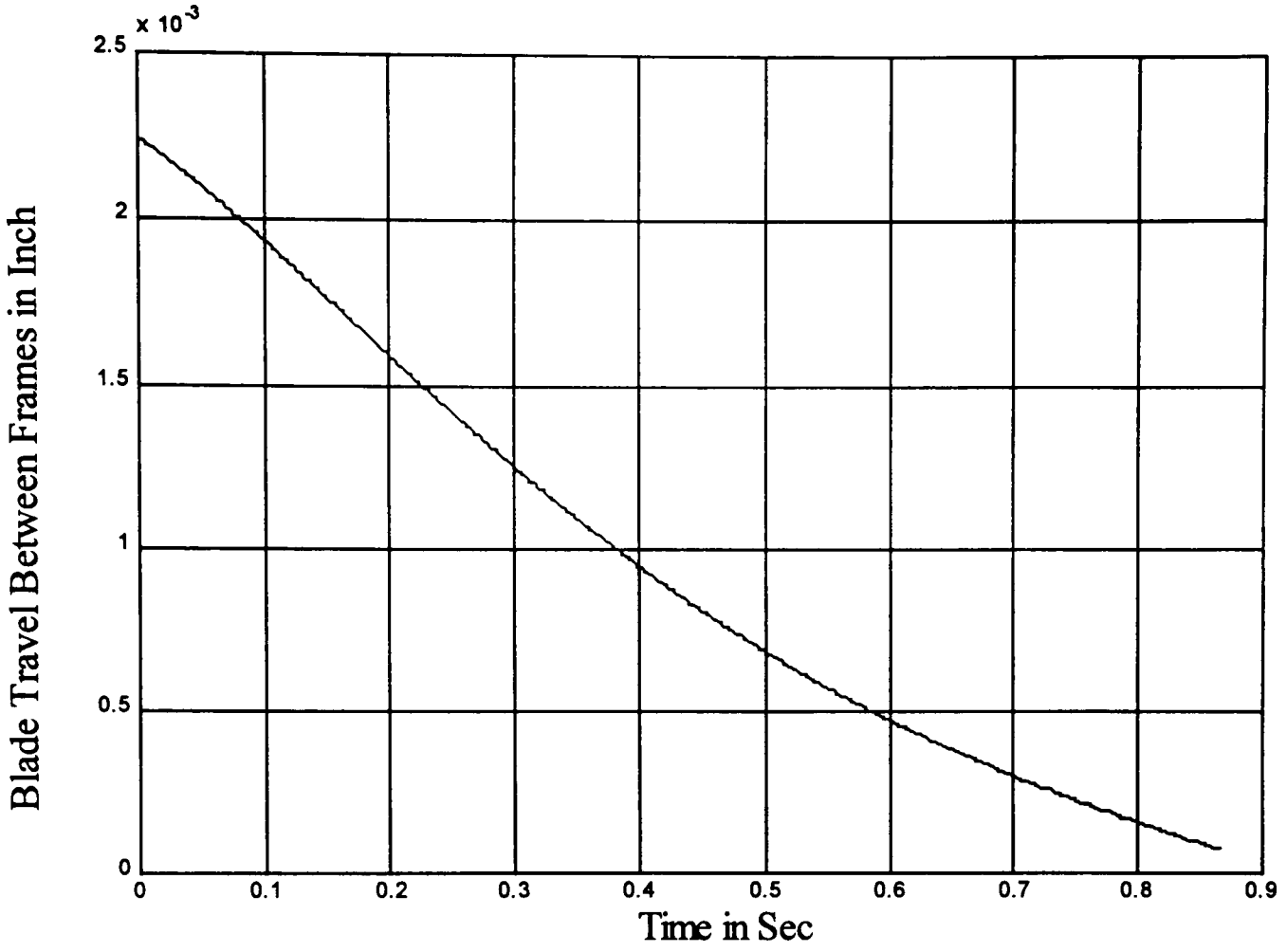

Figure 2.9: Blade Movements Between Frames (Inches) versus Time (Sec)

Since the robot blade travel between frames is larger at the beginning of the image capture time period and it gradually decreases, there are less image samples at the beginning of the image than at the end of the image. This causes the nonlinear distortion of the image.

The plot of the wing angle versus time is given by Figure 2.10, with line scan camera frame rate = 9000 lines/sec, blade acceleration = -25000steps/sec/sec.

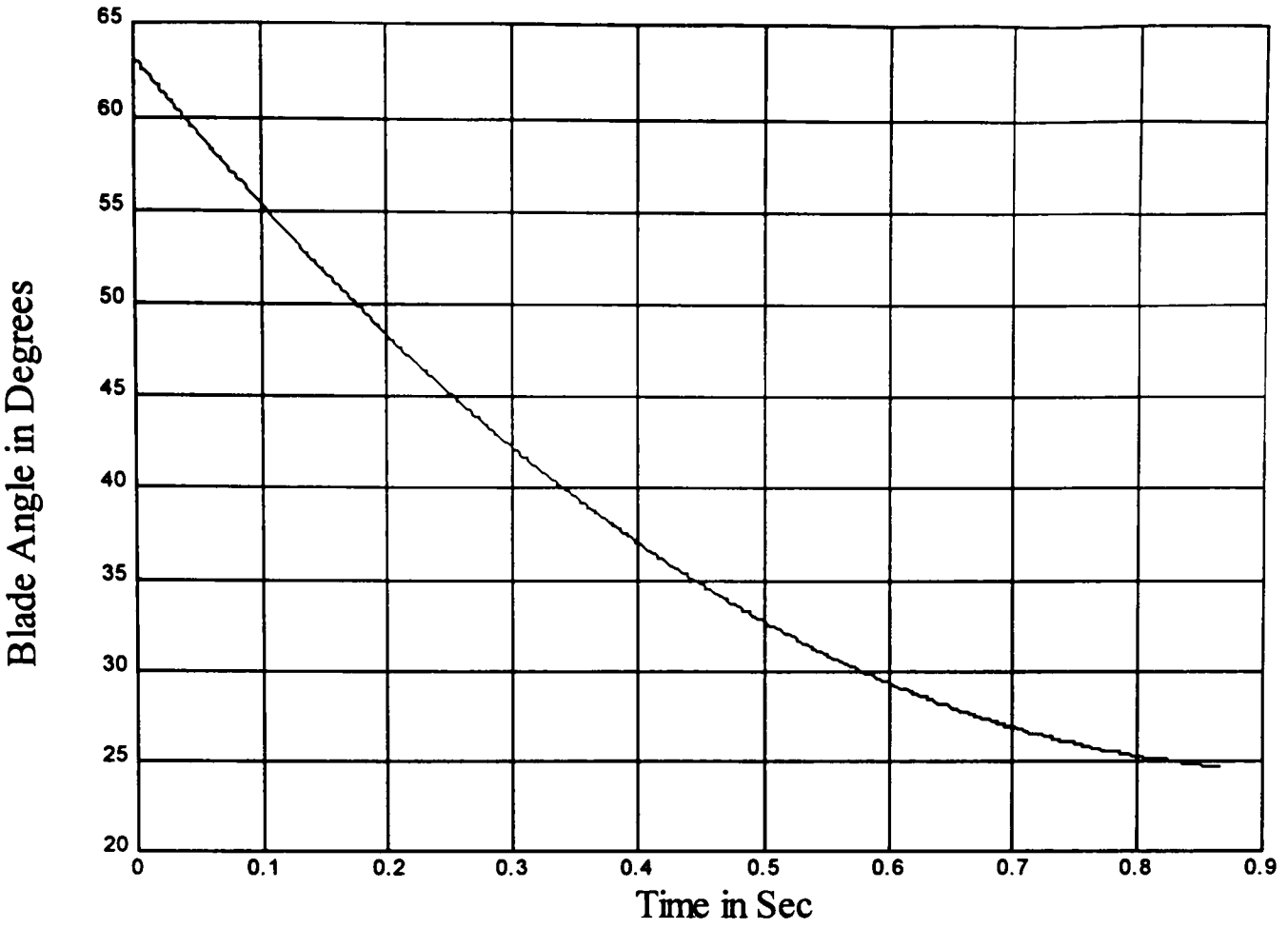

Figure 2.10: Blade Angle (Degrees) versus Time (Sec)

The line scan camera starts capturing the image of the object/wafer when the blade is extended at the position of approximately 18.05147 inch, and it stops image grab when the robot blade is at a position of approximately 23.47582 inch.

CHAPTER III

NONLINEAR IMAGE RESTORATION TECHNIQUE

Linear methods of image restoration have accounted for most of the practical applications of restoration to real-world problems. This is because the linear methods are can be easily computed in a straightforward and economical fashion. Conversely, nonlinear methods usually require much more elaborate and costly computational procedures.

3.1 Different Techniques of Nonlinear Image Processing

Several nonlinear image restoration methods are available, as listed below:

1. Optimal recursive processing: It is desired to estimate $f(x+1, y+1)$ as function of its immediate upper left neighbors and the value of the recorded image at those points. Thus it will be computed recursively in the form

$$f(x+1, y+1) = a_1 f(x+1, y) + a_2 f(x, y+1) + a_3 f(x, y) + a_4 g(x, y). \quad (3.1)$$

The problem is to find the coefficients a_i for some set of reasonable assumptions about the statistics of the image $g(x, y)$ and the initial object distribution $f(x, y)$. The recursive method is intriguing because, first, the Kalman filter, in its most general form, is a space-varying filter and thus offers the potential of treating problems where space variant point spread functions are involved, second, certain properties of image (e.g., positivity) present additional difficulties in Kalman filter design, which usually are based

upon zero-mean Gaussian statistics. Even though this method is superior, resolving the above-mentioned issues and its implementation is very complex.

2. Random-grain models: The restoration scheme based upon this objective can be stated as follows: Find the noise and object grain distributions that are most likely to have formed the observed image values; $g = [H]f + n.$ (3.2)

   Since noise and object grains are equally likely to occur in any of the N^2 sample elements, then the noise and object grain distribution is the one that can be formed from f_j and n_j in the maximum number of ways, given the image formation and recording constraint equation (3.2).

3. Bayesian methods and sensor nonlinearities.

4. Interpolation methods: The simplest form of resampling interpolation is to choose the amplitude of am output image pixel to the amplitude of the input pixel nearest to it. This process is called nearest neighbor interpolation. In this thesis, superior interpolation techniques are used for nonlinear image restoration.

3.2 Polynomial Curve fit

A set of observations can be summarized by fitting in to a "model" that depends on adjustable parameters. Sometimes the model is a convenient class of functions, such as polynomials or Gaussians, and the fit supplies the appropriate coefficients. The figure-of-merit function is conventionally arranged so that small values represent close agreement. The parameters of the model are then adjusted to achieve a minimum in the

merit function, yielding best-fit parameters. There are three important methods to fit a curve to the given data:

1. Method of Least squares,
2. Method of Group averages,
3. Method of Moments.

Among these three methods, the method of least squares is most widely used. The least square method is based on minimizing the sum of the squares of the difference between the observed data points and the values given by the approximating line.

The most common model is based on the assumption of a linear relationship between x and y of the form

$$y = a_0 + a_1 x \tag{3.3}$$

where a_0 is the intercept and a_1 is the slope of the line passing through the data points (x_1, y_1), $(x_2, y_2), \ldots$ (x_n, y_n). The values of a_0 and a_1 are determined so that the straight line passes through the data points with the least error. The calculated value at any point is given by

$$\hat{y}_i = \hat{a}_0 + \hat{a}_1 x_i . \tag{3.4}$$

The vertical deviations δ_i of the i th point from the regression is

$$\delta_i = y_i - \hat{y}_i = y_i - (\hat{a}_0 + \hat{a}_1 x_i) \tag{3.5}$$

where δ_i is the difference between the observed value y_i and the ordinate $\hat{y}_i$ of the fitting straight line at x_i. The sum of the squares of deviations is

$$S = \sum_{i=1}^{n} \delta_i^2 = \sum_{i=1}^{n} [y_i - (\hat{a}_0 + \hat{a}_1 x_i)]^2 . \tag{3.6}$$

The values of a_0 and a_1 are chosen so as to minimize S. S can be minimized by taking the partial derivatives of S with respect to a_0 and a_1 and setting the resulting equation to zero:

$$\frac{\partial S}{\partial \hat{a}_0} = -2\sum_{i=1}^{n}[y_i - (\hat{a}_0 + \hat{a}_1 x_i)] = 0 \tag{3.7}$$

$$\frac{\partial S}{\partial \hat{a}_1} = -2\sum_{i=1}^{n}[y_i - (\hat{a}_0 + \hat{a}_1 x_i)](x_i) = 0\,. \tag{3.8}$$

The solution of above simultaneous equation results in the values of a_1 and a_0. There are several indicators of how well the data can be represented by the curve fit. One useful indicator is the coefficient of determination. The coefficient of determination expresses the proportion of the total change in the dependent variable y that can be explained by the regression line. Thus

$$r^2 = \frac{\text{exp}\,lained\ \text{var}\,iation}{total\ \text{var}\,iation} = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2} \tag{3.9}$$

where $\bar{y} = \frac{1}{n}\sum y_i$. The value of r^2 ranges from 0 to 1. The closer r^2 is to 1, the better the regression equation fits the data.

The least square method is extended to fit a higher order polynomial to the given data set. An N degree regression polynomial is in the form

$$y = a_0 + a_1 x + a_2 x^2 + \ldots + a_N x^N \tag{3.10}$$

and the sum of the squares of the deviations is

$$S = \sum_{i=1}^{n}(y_i - a_0 - a_1 x_i - a_2 x_i - \ldots - a_N x_i^N)^2 \quad . \tag{3.11}$$

3.3 Cubic Spline Data Interpolation

For a given tabulated function $y_i = y(x_i), i = 1...N$, consider a particular interval, between x_j and x_{j+1}. Linear interpolation in that interval gives the interpolation formula

$$y = Ay_i + By_{i+1} \tag{3.12}$$

where

$$A \equiv \frac{x_{j+1} - x}{x_{j+1} - x_j},\ B \equiv 1 - A = \frac{x - x_j}{x_{j+1} - x_j}. \tag{3.13}$$

Since it is (piecewise) linear, equation (3.10) has zero second derivative in the interior of each interval, and an undefined, or infinite, second derivative at the abscissas x_j. The cubic spline interpolation gives an interpolation formula that is smooth in the first derivative, and continuous in the second derivative, both within an interval and at its boundaries.

So in addition to the tabulated values of y_i, there are also tabulated values for the function's second derivatives, y'', that is, a set of numbers y_i''. Then, within each interval a cubic polynomial can be added whose second derivative varies linearly from a value y_j'' on the left to a value y_{j+1}'' on the right. This will result in a desired continuous second derivative. This cubic polynomial also has zero values at x_j and x_{j+1}. So by rearranging equation (3.10)

$$y = Ay_j + By_{j+1} + Cy_j'' + Dy_{j+1}'' \tag{3.14}$$

where A and B are defined in (3.11) and

$$C \equiv \frac{1}{6}(A^3 - A)(x_{j+1} - x_j)^2,\ D \equiv \frac{1}{6}(B^3 - B)(x_{j+1} - x_j)^2. \tag{3.15}$$

The additional terms are clearly zero at the endpoints, and it can be derived that

$$\frac{d^2y}{dx^2} = Ay''_j + By''_{j+1}. \tag{3.16}$$

Typically, y''_i are not known. By requiring the first derivatives be continuous across each tabulated points x_j, $j = 2...N-1$, the following relations are found:

$$\frac{x_j - x_{j-1}}{6} y''_{j-1} + \frac{x_{j+1} - x_{j-1}}{3} y''_j + \frac{x_{j+1} - x_j}{6} y''_{j+1} = \frac{y_{j+1} - y_j}{x_{j+1} - x_j} - \frac{y_j - y_{j-1}}{x_j - x_{j-1}}. \tag{3.17}$$

This gives N – 2 linear equations for N unknowns. There are two ways to specify unique solution:

1. Set $y''_1 = y''_N = 0$ (natural spline).

2. Specify y'_1 and y'_N.

Cubic spline equations along with the two additional boundary conditions, are not only linear, but also tridiagonal. Each y''_j is coupled only to its nearest neighbors at $j \pm 1$. Therefore, the equations can be solved in O (N) operations by tridiagonal algorithm.

3.4 Image Restoration Approach for This Specific Case

In this case, since the image of the object/wafer is captured by line scan camera, the nonlinear velocity profile of the robot arm distorts the image. And this causes the image to be under-sampled at the beginning and over-sampled at the end section. The image is distorted only in Y-direction but not in the X-direction, so every column of the image has to be processed independently.

The following are the steps for nonlinear image processing via interpolation method.

1. There are two equations, which characterize the robot arm movements.

$$\text{Position } X = D + A\cos\Theta + \sqrt{B^2 - (A\sin\Theta - C)^2} \tag{3.18}$$

$$\text{Wing Angle } \Theta = \tan^{-1}\left(\frac{C}{X-D}\right) + \cos^{-1}\left(\frac{A^2 + C^2 + (X-D)^2 - B^2}{2A\sqrt{C^2 + (X-D)^2}}\right) \tag{3.19}$$

From the above equations, the following four parameters, i.e.,

- Time points at which the each line of the image was grabbed by the line scan camera;
- Position of the robot arm blade precisely at the discrete time points when each line of the image was captured;
- Robot arm blade movements in each time interval;
- Wing angle of the robot arm

are calculated.

All these parameters are stored in a vector of structure elements.

2. The image of the object/wafer was captured in linear time scale and in nonlinear distance scale (in blade extension versus time graph). In order to compensate for the nonlinearity, the linear time scale has to be converted to nonlinear time scale where each discrete time points corresponds to equal robot arm blade movements. From the geometry of the setup, the starting point of the frame is found out, which is at 18.05147-inch extension of the robot arm blade. The sampling rate of the camera is set to a fixed value of 9000 lines/second. So time taken to capture 4096 lines of the image is

$\frac{1}{9000} \times 4096 = 0.455111111$ sec. During this interval, the robot arm blade is extended to a position of 23.47582 inch.

3. This distance travel of 5.42435 inch is divided into 4096 discrete points. By using seventh-order polynomial curve fit corresponding points in the time scale is found (Figure 3.1). The seventh-order polynomial is

$$Y = -56841.28573 + 19699.7933X - 2919.154405X^2 + 239.7491075X^3 - 11.78668662X^4 + 0.346873308X^5 - 0.005658256765X^6 + 0.00003946734681X^7$$

Number of data points used = 7798

Average X = 22.1895

Average Y = 0.433278

Degree: 0

Residual sum of squares = 487.847,

Coefficient of determination, $r^2 = 2.220245 \times 10^{-16}$

Degree: 1

Residual sum of squares = 44.3261,

Coefficient of determination, $r^2 = 0.909139$

Degree: 2

Residual sum of squares = 10.6159,

Coefficient of determination, $r^2 = 0.978293$

Degree: 3

Residual sum of squares = 3.44348,

Coefficient of determination, $r^2 = 0.992941$

Degree: 4

Residual sum of squares = 1.37706,

Coefficient of determination, $r^2 = 0.997177$

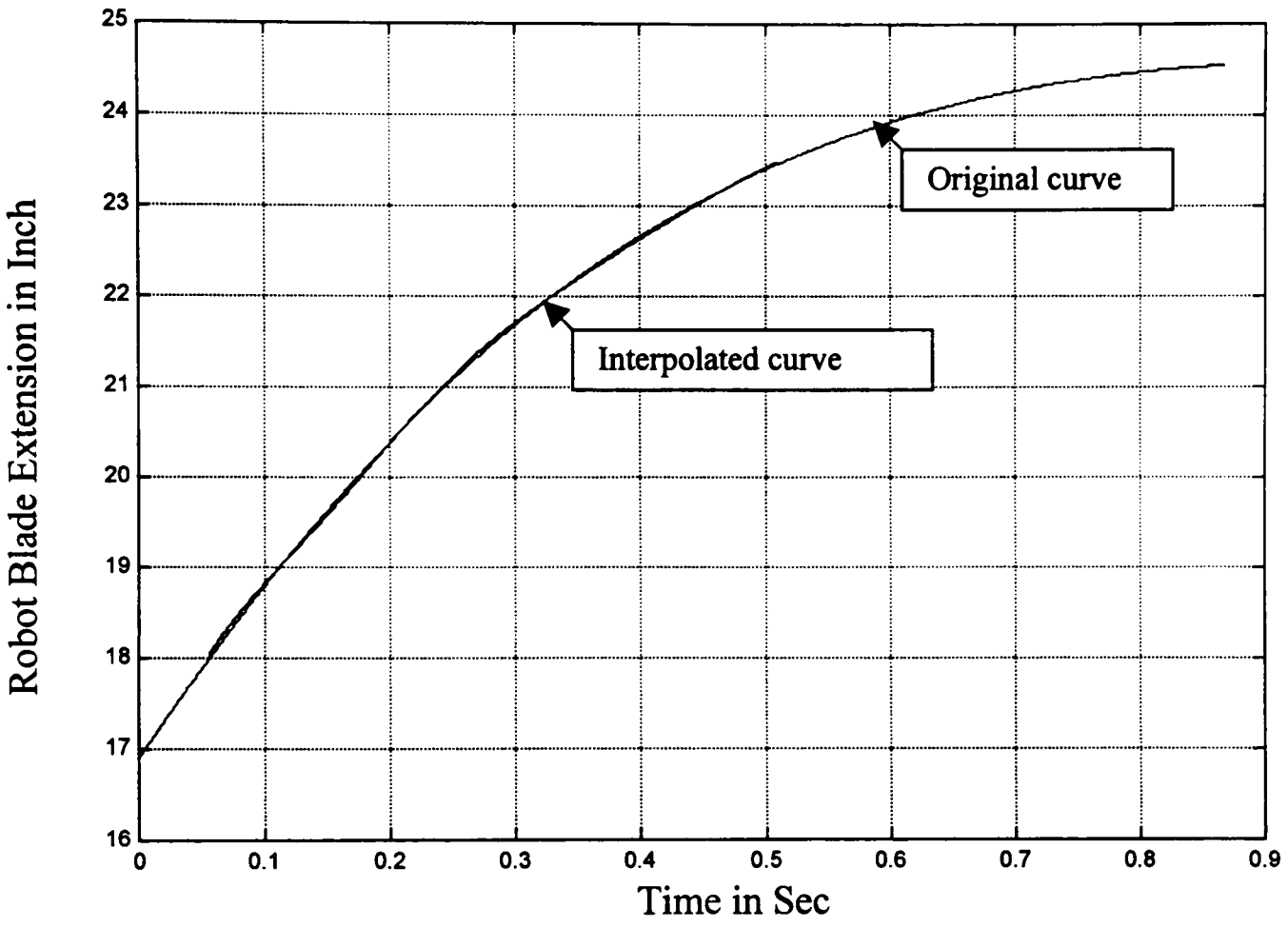

Figure 3.1: Robot Blade Extension (Inch) versus Time (Sec)

Degree: 5

Residual sum of squares = 0.629229,

Coefficient of determination, $r^2 = 0.99871$

Degree: 6

Residual sum of squares = 0.316142,

Coefficient of determination, $r^2 = 0.999352$

Degree: 7

Residual sum of squares = 0.170216,

Coefficient of determination, $r^2 = 0.999651$

All these interpolated time points are stored in a vector of structure elements.

4. Load the distorted image in an image buffer. Read each column of pixel values of the image and store all amplitude values in an array. For each column of image pixel amplitudes, compute cubic spline interpolation coefficients.
5. By using cubic spline interpolation coefficients, for each interpolated time points calculate the amplitude values for each pixel elements in the column and store that amplitude value in a image buffer.
6. Repeat this process for each column of the image. It is necessary to calculate cubic spline coefficients for each column of pixels because light illumination is not uniform across the object/wafer. By observation it is found that there is higher illumination intensity in the center of the object/wafer than that of edge of the object/wafer.
7. Flowchart of this algorithm is shown in Figure 3.2.

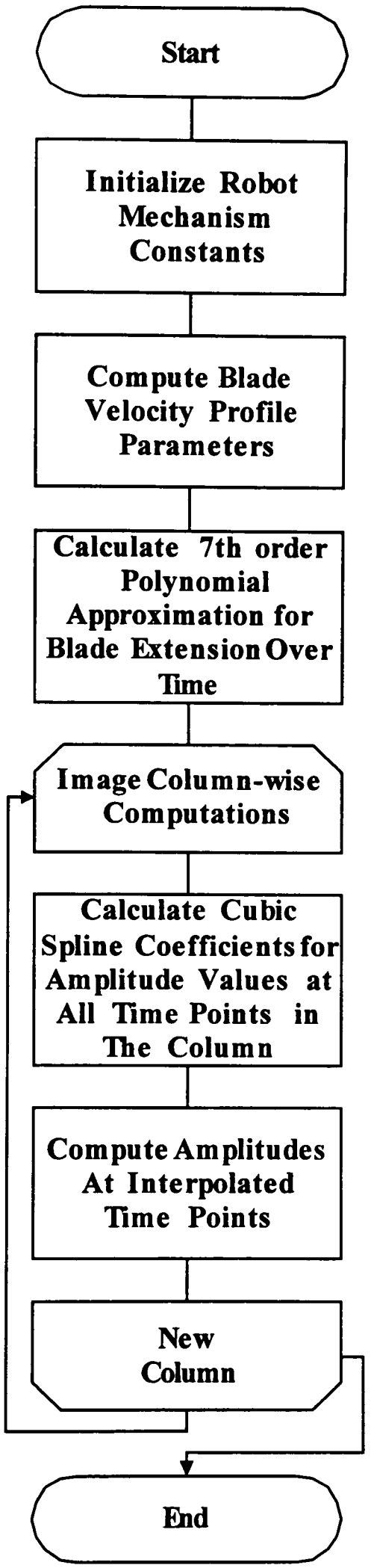

Figure 3.2: Flowchart of Nonlinear Image processing

CHAPTER IV

RESULTS

This chapter discusses the efficiency and performance of the nonlinear method of image restoration technique described in the previous chapter.

The aim of image restoration is to bring the image toward what it would have been if it had been recorded without degradation. These degradations may be the blurring that can be introduced by optical systems, image motion, and the like, as well as noise from electronic and photometric sources. In our application, degradation caused by the nonlinear motion of the robot arm carrying the object/wafer is primarily considered.

The seventh-order polynomial curve fit gives a considerably good approximation for robot arm blade extension versus time curve. For degree 7, residual sum of squares was 0.170216 and coefficient of determination, $r^2 = 0.999651$, which is very close to 1.

In our case, image data points are presorted in ascending order in time scale. This happens naturally as the beginning of each column of image data is earlier in the time scale than that of the end of each column of the image. This eliminates the requirement of computer intensive sorting algorithm. After reading the data set, cubic spline coefficients were generated successfully. Finally, the algorithm generates the amplitude value for each interpolated time points.

This algorithm corrects the image and removes the nonlinear distortion in the image very well. However, for this algorithm to work successfully, the starting time and the ending time of image grabbed have to be known precisely. Otherwise incorrect interpolated data points will be generated either at the beginning or at the end section of

the image. Since the seventh-order polynomial does not give perfect curve fit, a minute distortion remains in the middle part of the image. This can be eliminated by using higher order polynomial curve fit.

Considering the number of data points required to be computed, this algorithm performs very well. The following result is derived from IBM compatible personal computer. Computer environment specifications are:

1. 233 MHz AMD K6 processor, 64 MB RAM
2. Windows NT 4.0, with service pack 6.0.

In order to correct a gray-scale (single band) distorted image of 512 * 4096, it took approximately 40 sec.

The software implementation was done by using Microsoft Visual C++ 6.0 with service pack 3.0.

CHAPTER V

CONCLUSIONS

A large portion of digital image restoration is implemented by using linear image processing methods. Majority of available nonlinear image restoration techniques are either very difficult to implement or proprietary knowledge. The algorithm proposed here is generic in nature for an image acquired by a line scan camera. If the distortion function is known, and image grab time points are known, then this algorithm can be implemented to correct almost all nonlinear image restoration work. Here it is assumed that distortion is only in one direction.

The result of this nonlinear image restoration can be improved by using higher order polynomial curve fit. Tenth or higher order polynomial curve fit can be used for accurate result. For better computational accuracy, instead of double data type, long double data type can be used. For higher order smoothness, either bicubic spline or bicubic interpolation can be used.

The execution time of this algorithm can be reduced by approximation. By using seventh order polynomial interpolation, an interpolated time point is generated so that the distance-traveled scale (in the robot arm blade travel versus time plot) becomes linear and time scale becomes nonlinear. Now instead of computing pixel amplitude value for each interpolated time points, the interpolated time point is checked against the actual time points. If the interpolated time point has the same value as one of the actual time points, then the pixel value at that point is accepted. This procedure can drastically minimize the computation requirements of the algorithm. Furthermore, elimination of the computation

of cubic spline coefficients for every column of the image can make the algorithm faster. But in that case if there is any unequal light intensity distribution along horizontal axis, the image will not be compensated correctly.

REFERENCES

1. Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing, Addison-Wesley Longman, Inc., Reading, MA, September 1993.

2. H. C. Andrews and B. R. Hunt, Digital Image Restoration, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1977.

3. B. R. Hunt, "Super-resolution of images: Algorithms, principles, performance," Int. J. Imag. Syst. Tech., Vol. 6, pp. 297 – 304, Winter 1995,

4. William K. Pratt, Digital Image Processing, John Wiley & Sons, Inc., New York 1991.

5. David G. Sheppard, Kannan Panchapakesan, Ali Bilgin, Bobby R. Hunt, and Michael W. Marcellin, "Lapped Nonlinear Interpolative Vector Quantization and Image Super-Resolution," IEEE Transaction on Image Processing, Vol. 9, No. 2, February 2000.

6. William H. Press, Saul A. Teukolsky, William T. Vetterling and Brian P. Flannery, Numerical Recipes in C The Art of Scientific Computing, Second Edition, Cambridge University Press, New York,1992.

7. K. B. Rojiani, Programming in C with Numerical Methods for Engineers, Prentice-Hall, Inc, Upper Saddle River, New Jersey, 1996.

8. Bruce Fiala, "High-Resolution Machine Vision Inspection," Sensors, pp. 67 – 72, November 1999.

9. Fostech Technical Reference Manual, Schott-Fostech, LLC, Inc., New York, 2000.

10. Basler L2X0 Line Scan Camera Technical Reference Manual, Basler Vision Technologies, Germany, 2000.

11. Matrox Meteor – II/Digital Technical Reference Manual, Matrox Electronics Systems Ltd., Montreal, Canada, 2000.

12. Technical Reference Manual, Applied Materials Inc., Santa Clara, California, 1999.

13. George Shepherd and Scot Wingo, MFC Internals Inside the Microsoft Foundation Class Architecture, Addison Wesley Longman, Inc., Reading, MA, January, 1999.

14. Ivor Horton, Beginning Visual C++ 6, Wrox Press Ltd.30 Lincoln Road, Olton, Birmingham B27 6PA, UK, August 1998.

15. H. M. Deitel and P. J. Deitel, C++ How to Program, Second Edition, Prentice-Hall Inc., Upper Saddle River, New Jersey, 1997.

APPENDIX

Microsoft Visual C++ implementation scheme of the algorithm is given below:

1. CRobotSimulator class generates three curves (robot arm blade extension versus time, wing angle versus time, distance traveled by the blade versus time) from the robot equations and it saves all data in a vector of structures. This structure has four data members viz., time, distance traveled by the blade, distance traveled by the blade in between frames, and blade angle.
2. CDataInterPolator class implements seventh order polynomial curve fit. All the data points are saved in a vector of structures. This structure has two data members, time and distance.
3. CCubicSpline class implements the cubic spline algorithm.

Throughout the implementation scheme Hungarian notation methodology is adopted.