The Design, Implementation, and Validation of a 3D Laser Light-sectioning Scanner for Biomedical Purposes

by

Beverly D Bradley

A Thesis submitted to

the Faculty of Graduate Studies and Research

in partial fulfilment of

the requirements for the degree of

Master of Applied Science

in

Biomedical Engineering

Ottawa-Carleton Institute for Biomedical Engineering

Department of Systems and Computer Engineering

Carleton University

Ottawa, Ontario, Canada

September 2008

Abstract

Three-dimensional scanning is quickly becoming a valuable imaging modality in the healthcare field for a variety of applications requiring the external shape features of anatomical parts and other biological objects. It presents many clear benefits over existing and traditional medical imaging modalities, however the availability of biomedical-specific scanners and the cost of existing commercial models prohibit the widespread use of 3D scanning in a medical setting. The potential exists to expand the use of 3D imaging even further, by continuing to develop simpler, more cost effective scanning systems.

In this thesis, a simple, low cost, 3D scanning system is presented, which employs the laser light-sectioning technique for data acquisition. A complete description of the system's calibration, image acquisition, image processing, data manipulation, and object reconstruction processes is provided. Specific focus is placed in the areas of calibration and data manipulation. For calibration, it is shown that the Direct Linear Transformation least squares algorithm is better than a purely analytical technique for finding the transformation required to calibrate images when input calibration data are noisy. For data manipulation, Elliptical Fourier Descriptors are shown to be an appropriate and beneficial means of fitting a smooth curve through the data collected from the scanner. The proposed design has been implemented in two prototypes. Promising results are shown which demonstrate the validity of the proposed design. Current system limitations are discussed with directions for future work.

To my family

Acknowledgments

I would like to gratefully acknowledge my supervisors, Dr. Adrian Chan and Dr. John Hayes, for their guidance and support throughout my graduate career and during the completion of this thesis. Their enthusiasm for this research and genuine encouragement along the way has made this experience an enjoyable one. I am also obliged to NSERC, the Department of Systems and Computer Engineering, the Faculty of Graduate Studies, my supervisors, and the family of Allan Buchanan for the generous financial support which has been provided to me.

I would like to thank Jason for his infinite patience and support throughout this past year. Thanks to Lindsay, a loving sister and patient roommate, whose unwavering dedication to her own studies has been an inspiration as I aimed to completed mine. Finally, I dedicate this thesis to my loving parents, grandparents, and Grams, who have privileged me with unconditional support, encouragement, and love throughout this experience, and in each of my life's endeavours. I extend my deepest appreciation to them.

List of Symbols

α angle between the camera and the laser plane (i.e., xy-plane)

ρ_i independent homogeneous scale factor for i^{th} point mapping

σ^2 variance

A_i DLT matrix equations for i^{th} point mapping

A_0, a_n, b_n coefficients of the Fourier series expansion of $x(t)$

a area of the polygon formed by a set of point coordinates

$\mathbf{b}$ array of known calibration coordinates

C_0, c_n, d_n coefficients of the Fourier series expansion of $y(t)$

C_f correction factor obtained from calibration cylinder

D_a, D_e actual and measured diameter of the calibration cylinder

H homogeneous transformation matrix

HN calculated harmonic number for a planar section

$\mathbf{h}$ array of H matrix parameters

$\tilde{\mathbf{h}}$ 1×8 array of the first 8 elements of $\mathbf{h}$, the column vector of H matrix parameters

$I(x,y)$ the intensity value at location (x,y) in an image

K number of points in an arc

M number of views captured about an object

m number of calibration point pairs used to find H

N number of harmonics kept for the EFD approximation

N_{max} maximum number of harmonics kept for the EFD approximation

n	number of planar sections
PA	perimeter/area ratio for a planar section
$\mathbf{p}_i$	homogeneous coordinates of a point in the image plane
$\mathbf{p}_i'$	homogeneous coordinates of a point in the world reference frame
p	power factor for center of gravity formula
pm	perimeter of the polygon formed by a set of point coordinates
R	rotation angle between each view
RF	roundness factor for a planar section
T	period of $x(t)$ and $y(t)$ (i.e., time required to trace the entire contour at a constant speed)
$V(t)$	2D continuous closed contour as a function of time
$\mathbf{x}_i,\ \mathbf{x}_i'$	general 2D homogeneous projective point coordinates
$x(t),\ y(t)$	projections of $V(t)$ on the x- and y-axes
$x_N(t),\ y_N(t)$	truncated approximations of the Elliptical Fourier Functions
y_c	one-dimensional intensity center of gravity in an image
$y_s,\ y_e$	start and end indices for the column of pixels which exceed the threshold

List of Acronyms

AGWN   Additive White Gaussian Noise

ANOVA   Analysis of Variance

BPA   Ball-pivoting Algorithm

CAESAR   Civilian American and European Surface Anthropometry Resource

CAD   Computer Aided Design

CCD   Charge Coupled Device

CMM   Coordinate Measuring Machine

COG   Centre of Gravity

CT   Computed Tomography

DLT   Direct Linear Transform

EFD   Elliptical Fourier Descriptor

EFF   Elliptical Fourier Function

LCD   Liquid Crystal Display

LED   Light Emitting Diode

LS   Least Squares

MLT   Multistripe Laser Technology

MRI   Magnetic Resonance Imaging

NBV   Next Best View

NI   National Instruments

OA   Osteoarthritis

.obj   Wavefront Object File Format

.png	Portable Network Graphics File Format
RGB	Red Green Blue Colour Space
RMS	Root Mean Squared
SCSI	Small Computer System Interface
SPSS	Statistical Package for the Social Sciences Software
USD	United States Dollar

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Introduction to 3D Scanning and its Biomedical Applications

For many years, three-dimensional (3D) scanning systems for acquiring the external shape features of arbitrary objects have been widely used for industrial applications such as reverse engineering, part inspection, and automated design [1], [2]. Over the past few years, dramatic decreases in the cost of 3D scanning equipment have led to its increased use for many other applications, including rapid prototyping and 3D printing, development of realistic computer graphics in the video gaming market [1], e-commerce, various applications in the apparel industry [3], and more recently, in the biomedical field. Biomedical applications of 3D scanning include anatomical parts reconstruction [4], orthodontic treatment planning [5], cranial deformation research [6], cartilage morphology studies [7], anthropometric data collection [8], and various forms of surgery (plastic and maxillo-facial for example [9]). Additionally, "this technology enables full customization of manufactured parts for individuals by scanning and generating a geometric model of a human body and its respective parts and then designing customized one-of-a-kind products. The application of this technology can be

especially useful for the handicapped population in need of aids, such as wheelchairs, customized orthopedic shoes, special braces, and so on" [2].

There exists a variety of different techniques for acquiring 3D models of objects, all with a wide range of hardware costs, and differing levels of achievable accuracy and detail in the captured geometric models. Stereoscopic image-based techniques [10], structured light [11,12] and laser light-sectioning methods [13,14], time-of-flight range finders, and shape-from-silhouette algorithms [15], are all methods which have been studied in recent years. With advances in camera technology and computing power, it is becoming more and more possible to create such instrumentation both economically, and for specific applications.

That being said, the majority of commercial 3D scanning systems have been developed with industrial applications in mind, and are not tailored specifically to a biomedical environment. There is currently a missing link between 3D scanning technology and biomedical-specific scanners, despite the emergence of applications where 3D models of anatomical parts are proving invaluable for enhancing patient care. Therefore, the potential exists to increase the accessibility of 3D scanning technology in healthcare by continuing to develop simpler, more cost effective systems for acquiring the external shape features of biological objects. This will help expand the use of 3D models in biomedicine even further.

1.2   Thesis Objectives

The overall objective for this research is the development of a simple 3D scanning system for biomedical purposes, with the underlying goals of the system being cost effectiveness and versatility. We decided to focus upon the accepted technique of laser light-sectioning for this system. In this thesis, our objectives were to explore multiple design options at different stages of the 3D scanning process and to validate

the chosen techniques experimentally, specifically in the areas of calibration and data processing. It was also an objective to demonstrate the execution of the chosen design options by implementing a prototype system and testing this system on multiple test objects, including an object representative of a biological object.

1.3 Thesis Contributions

1.3.1 Major contributions

The following is a list of thesis contributions with brief descriptions of each.

1. **An overall system design for a simple, low-cost, versatile 3D laser scanning system using the laser light-sectioning technique**

   This thesis presents the overall system design for a simple and low-cost 3D laser scanning system. Different methods of accomplishing various stages of the scanning process have been examined and their advantages and disadvantages are discussed. Based on the accepted laser light-sectioning technique, the chosen design is both economical and versatile, taking into account several design considerations for biomedical applications.

   This contribution was presented at the IEEE International Instrumentation and Measurement Conference, $I^2MTC'08$ (Victoria, BC) in May 2008 [16].

   a. Design implementation, validation, and testing

   In this thesis, the feasibility of the proposed design is demonstrated with two prototype iterations; the first validating the calibration and data processing techniques and the second offering more automation and precision in the

data collection process. Implemented systems were tested with numerous test objects, including an object representative of a biological entity. Measures and procedures have been set for the quantitative assessment of system performance. Quantitative and qualitative results are presented. Overall, both prototypes validate aspects of the calibration, image acquisition, image processing, and object reconstruction capacity of the chosen design.

2. **The identification of the best calibration technique between two viable methods**

In this thesis, it is shown that compared to a purely analytical technique, the Direct Linear Transformation (DLT) algorithm using least squares (LS) is the best technique for finding a homogeneous transformation matrix, H, which is used to remove camera perspective distortion in the system. The two techniques were evaluated according to their performance using noisy data. Simulation data were created to mimic the image calibration for the proposed scanning system. A 3-way analysis of variance (ANOVA) test was used to identify the effects of calibration method, noise level, and point location within the image on transformation accuracy. Although studies exist in which calibration methods have been evaluated for accuracy, to our knowledge, no study has looked at the interaction between all three of these factors. Overall, the DLT LS method proved to be more robust when input data are noisy and is thus recommended as the best algorithm to be used in this scenario.

Results from this work were presented at the 31st Annual Canadian Medical and Biological Engineering Conference, CMBEC31 (Montréal, PQ) in June 2008 [17].

3. **Demonstrated use of Elliptical Fourier Descriptors (EFDs) as a suitable method for planar section reconstruction when using the laser-light sectioning method**

   a. **EFDs for noise reduction (results for real and simulated data)**

   b. **EFDs for creating seamless planar section contours (results for real data)**

Ideally, the point coordinate data collected would correspond exactly to the position of the object's surface profile in 3D space; however, due to noise in the measurement data, this is not the case. The effect of this noise on the point coordinate data should be minimized. Furthermore, a complete and continuous planar section of the object's profile must be constructed from multiple segments of data since the scanner collects data from multiple views about the object in order to give a 360° view. This thesis presents work demonstrating how EFDs can be used for reducing noise in the captured data, as well as for reconstructing the data segments into a complete continuous planar section contour. Through analyses on both simulated and real data it is shown that EFDs are a useful data processing tool for the proposed scanner. To our knowledge, EFDs have not been used for a 3D biomedical scanner of this type.

Results from this work were presented at the IEEE International Workshop on Medical Measurements and Applications, MeMeA '08 (Ottawa, ON) in May 2008 [18].

1.3.2 Minor contributions

Although not the main focus of this research, the following is a list of minor thesis contributions with brief descriptions of each.

1. **A consolidated literature review**

   This thesis presents a comprehensive literature review of the state of the art in 3D scanning technology, with a specific emphasis on laser light-sectioning systems. Additionally, a review of recent biomedical research involving commercial 3D laser scanners is presented, which lead to the compilation of a list of design considerations and recommendations for the further development of biomedical-specific laser scanners.

2. **Labview application for data collection**

   A Labview (National Instruments) application was developed for data collection. Images are captured and scanner mechanics are controlled via a single user-friendly interface, facilitating future work on the scanner apparatus.

1.4 Organization of Thesis Dissertation

The remainder of this dissertation is organized as follows. Chapter 2 presents a high-level review of the literature in the field of 3D scanning, specifically in the area of laser light-sectioning. Chapters 3 to 7 present different aspects of the 3D laser light-sectioning scanning system. A system flow diagram of these aspects is shown in Figure 1.1, which also illustrates the organization of these chapters.

Chapter 3 presents an overview of the 3D laser light-sectioning technique; a general system design is presented with relevant literature which lead to a more refined system design. This refined design is Contribution 1. Chapter 4 discusses the calibration of the system. Contribution 2 is contained in Chapter 4. Chapter 5 presents image
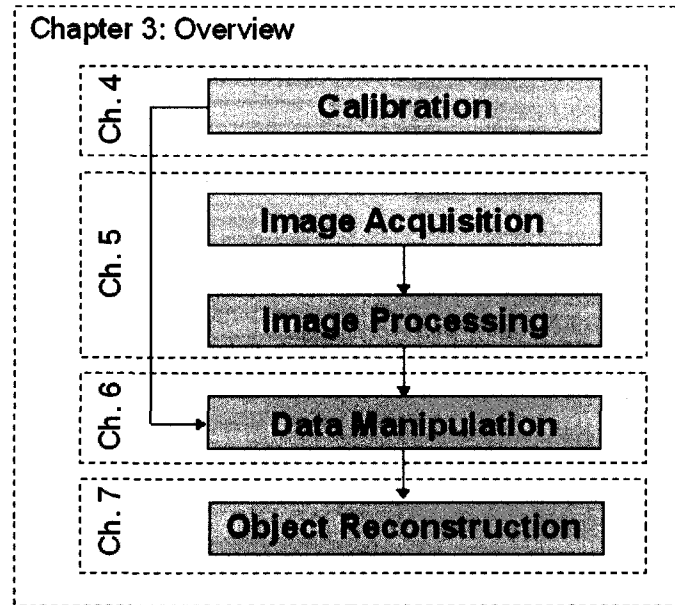
Figure 1.1: Diagram of chapter organization based on system flow diagram.

acquisition and image processing combined. Chapter 6 describes the processes used for data manipulation. Contribution 3 on Elliptical Fourier Descriptors is contained in Chapter 6. Object reconstruction is presented in Chapter 7.

With the overall system described completely, Chapter 8 presents results of the proof of concept experiments conducted for the proposed system which demonstrate the validity of the chosen approach. These results constitute Contribution 1a. Conclusions and recommendations for future work are discussed in Chapter 9.

Chapter 2

Background and Literature Review

This chapter provides a high-level overview of 3D scanning technology and current literature on the topic. Section 2.1 describes the basic principles behind the many different technologies that currently exist for acquiring 3D object surface data. In Section 2.2, a more detailed background on the state of the art in laser light-sectioning scanners is provided, the particular technique chosen for this thesis. To provide insight into the importance of 3D imaging technology in medicine, Section 2.3 provides examples of where 3D scanning has proven to be useful in biomedical applications, many of which use the laser light-sectioning scanners specifically highlighted in Section 2.2. Finally, the motivation for this thesis is presented in Section 2.4.

2.1   3D Scanning in the Literature

Many different techniques for acquiring 3D models of objects currently exist, all with a wide range of hardware costs, and differing levels of achievable accuracy and detail in the captured geometric models. Figure 2.1 summarizes these techniques and their respective classifications. High-level reviews of the different 3D model acquisition techniques and the processing of range scanner output data into efficient numerical representations of objects can be found in [1, 2, 19].
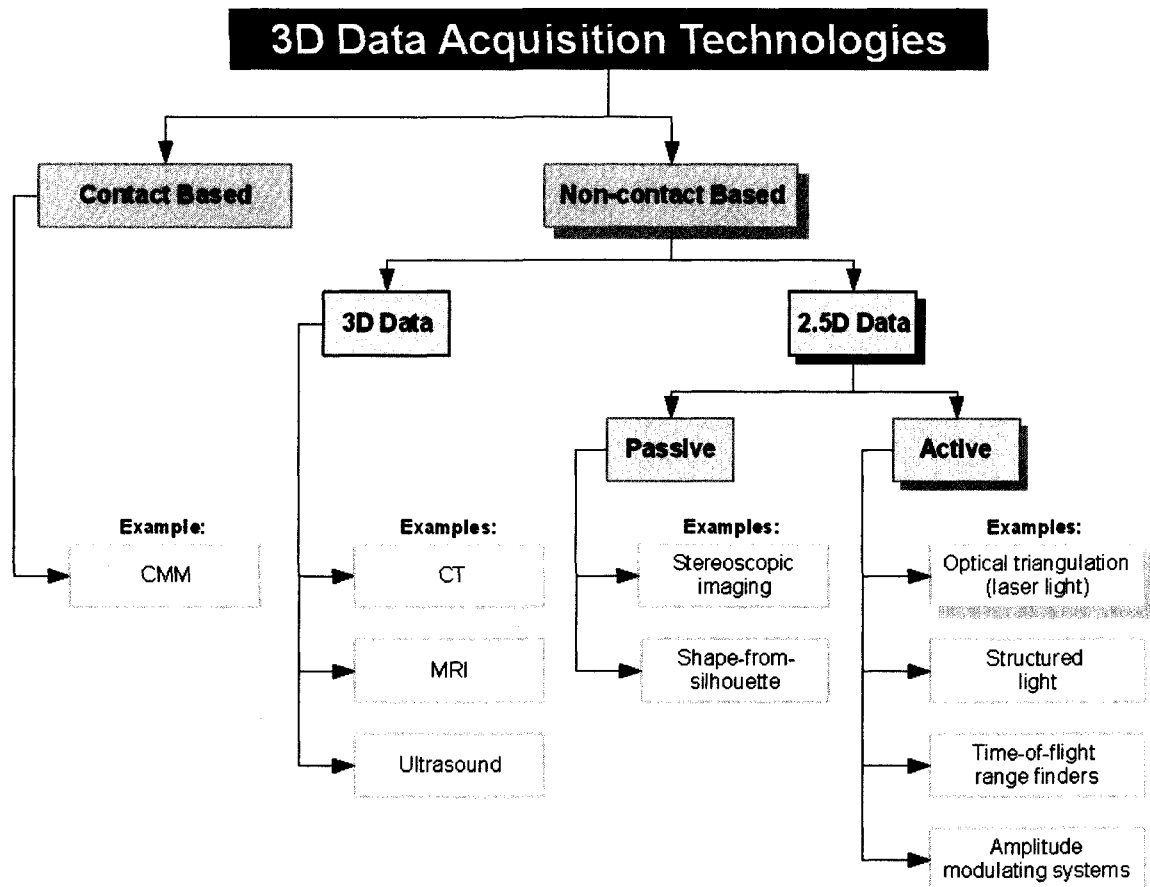
Figure 2.1: Diagram of 3D data acquisition technologies and their classifications.

At the most general level, these different technologies are classified into two main measurement categories: contact based and non-contact based [2]. As the names imply, contact based measurement systems require physical contact with the object of interest, and non-contact based systems do not.

Contact-based 3D measurement systems tend to be very precise, but are quite slow, expensive, and not easily automated [2]. An example of a contact-based system is a Coordinate Measuring Machine (CMM) in which the coordinates of a probe are recorded when it is physically brought in contact with the object; this tends to be a very manual process [2].

In the non-contact realm of 3D scanning, a further division is based on the use

of 3 or 2.5 dimensional sensing devices; that is, some technologies are capable of capturing true 3D information, whereas others are considered to provide data in "2.5 dimensions", meaning 3D information is inferred from 2D data, such as images.

2.1.1   Non-contact 3D systems

Examples of 3D sensing devices are found mostly in medicine; Computed Tomography (CT), ultrasound, and Magnetic Resonance Imaging (MRI) [2] are a few examples. CT measures the attenuation of radiation passing through the body, based on the density of the tissue or part of the body being imaged [20]. The result of a CT scan is a series of images that depict 'slices' of the part of the body being imaged. The major disadvantage of CT is the fact that it uses ionizing radiation, which can cause tissue damage. Ultrasound imaging uses sound as the external source of energy which penetrates the medium. Measured characteristics of the reflected sound waves reveal details about the inner structure of the medium [20]. MRI uses magnetic energy to excite selected atomic nuclei within the object such as hydrogen protons, which, during the relaxation process, provide electromagnetic signals which can be used to form an image [20]. MRI images have very high-resolution and provide excellent soft-tissue contrast, however scanning time tends to be long (between 3 to 10 minutes) and is therefore susceptible to patient motion [20].

In general, 3D sensing devices are *extremely* expensive to purchase and operate (in the range of millions of dollars) [2]. Comparatively, 2.5D based measurement systems have become more ubiquitous due to their speed in acquiring data, their non-invasive nature, and their relatively lower cost, which is more in the range of tens to hundreds of thousands of dollars [2]. In contrast to CT, MRI, and ultrasound, 2.5D systems can only yield external features of objects, not internal ones. These sensing devices can further be categorized as being either active or passive, each of which is explained in more detail in the following sections.

11

2.1.2 Non-contact 2.5D passive systems

Passive 2.5D devices simply record light that already exists in the scene [19]; that is, they do not emit any kind of radiation themselves but instead rely on detecting ambient or natural light reflecting off the object [2]. These systems employ at least one camera, but often involve a multi-camera setup [2]. Passive methods tend to be very inexpensive, because in most cases, the instrumentation required is very simple; just digital cameras without the need for particular optical hardware. In general, the acquisition times for passive imaging systems are well below one second, which is one of this method's main advantages [6, 13]. Another advantage, due to their imaged-based nature, is the fact that they can simultaneously capture color and texture information, making them suitable for applications requiring realistic 3D visualizations, such as the creation of avatars for game development [13]. They also have a larger viewing angle compared to active systems [13]. On the other hand, "the quality of these digitizers remains below that of active digitizers" [13]; lengthy post-processing time, lower geometric accuracy, design complexity, and performance when scanning topologically complex objects remain issues for passive techniques [6,13,19]. Examples of non-contact passive 2.5D techniques are stereoscopic imaging and shape-from-silhouette imaging.

Stereoscopic imaging

Stereoscopic imaging (or stereophotogrammetry) refers to the special case of passive measurement where *two* cameras (located slightly apart) are used. By simulating human stereoscopic vision, 3D distances to the surface of the object are recovered from the images by means of triangulation [5,19]. Stereoscopic systems are considered to be potentially one of the best solutions for video-rate 3D data capture due to the speed at which such a system can acquire data [13].

Shape-from-silhouette

Another passive technique is called the shape-from-silhouette method [15]. This technique involves placing an object in front of a plain background and imaging it from different angles. Silhouettes of the object's outline against the background can be extracted from the images and substituted for actual geometry. A major restriction of this method is that only smooth convex objects can be reliably digitized [13].

2.1.3   Non-contact 2.5D active systems

In order to measure 3D surface information, active scanners *actively* emit some kind of radiation or light, sometimes using special patterns, and then detect the light's reflection off of the object's surface using an appropriate sensor [2]. Different devices use different illumination sources, such as coherent light (laser) or incoherent light (white light), with varying wavelengths or bandwidths [2]. Active systems usually provide more accurate results than passive systems, with less computational cost. Examples of non-contact active 2.5D techniques are optical triangulation, structured light, time-of-flight range finding, and amplitude modulation.

Optical triangulation

"One of the most common forms of active range sensing is optical triangulation" [19]. This technique uses laser light as the illumination source. The fundamental principle behind optical triangulation is as follows. A focused beam of light illuminates a tiny spot on the surface of an object. The reflected light beam is collected by a sensor and the location of the centre of the spot is computed. Tracing a line of sight through the location of the spot until it intersects the illumination beam will reveal the point on the surface of the object from which the light is being reflected, as shown in Figure 2.2. The spot location, the camera's line-of-sight, and the laser beam form a triangle.

With certain aspects of the geometry known, namely, the spot location on the sensor u, the triangulation distance b, the angle between the laser beam and the camera α, and camera's focal length f, the distance to the point on the surface can be computed using trigonometry.

Laser beam   **u**   **Sensor**

b   **f**

x   **Lens**

α

$$x = \frac{bu}{f\cot\alpha - u}$$

$$z = \frac{bf}{f\cot\alpha - u}$$

z

(x,z)   **Target**
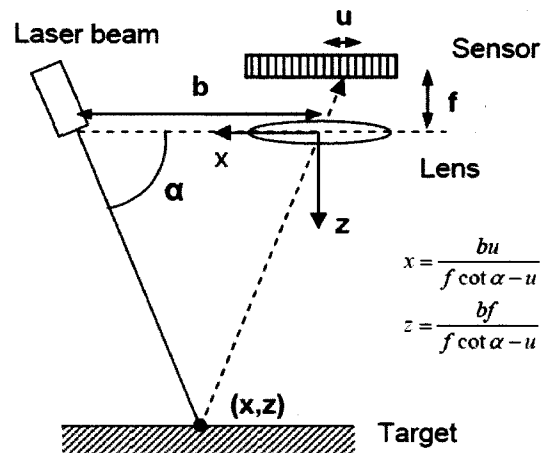
Figure 2.2: Diagram illustrating triangulation (adapted from [21]).

Many variations of this basic setup have been developed over the years to facilitate the scanning of an entire object's surface. The laser spot can be scanned over the entire surface of the object, either manually or with mirrors [19]. Another common approach is to use special optical filters to fan the laser beam into a plane which then appears as a stripe across the object. In this *laser light-sectioning* approach, the sensor used is a 2D matrix, typically a charge coupled device (CCD) camera. The resulting image provides a profile of the object at the location of the laser stripe. This is the approach used in this thesis; a more detailed literature review in this area of 3D scanning is provided in Section 2.2.

There are many reasons for choosing laser as the source of illumination for a 3D scanning device. Compared to incandescent or fluorescent light, "lasers can be focused tightly over very long distances and they have an extremely narrow radiation spectrum" [19]. The ability to project very narrow stripes allows for better system

resolution, and with a depth of field considerably larger than that of a typical sensor camera, they do not have to be focused during normal operation [13]. Their narrow spectrum makes them less susceptible to interference from ambient illumination in the surrounding environment [19]. This also means the laser can be easily filtered out from ambient light using a narrow-bandpass filter, allowing laser systems to be used in normal daylight conditions [13]. Furthermore, "diode lasers are compact and inexpensive" [13] making them a natural choice for a 3D scanning system. One fundamental limitation of using laser technology is the requirement of having an adequate clear view between the surface points being scanned and both the laser source and the sensor [1].

Structured light

A common variation to the optical triangulation approach is to project many points or stripes of light at once to capture as much shape information as possible in one image [19]. These structured light systems function according to the same basic principles of triangulation, except now a 2D pattern of stripes or dots (i.e. a grid or a line stripe pattern) is projected onto the object of interest. The deformation of the pattern on the object is captured by a CCD camera and used to calculate distances [9, 13]. The patterns are typically projected with one or more LCD projectors. The main advantage of structured light systems is their acquisition speed because many dots or stripes are captured at once; a fast acquisition time can reduce distortion from object motion.

One challenge with this approach, however, is that since the system projects multiple patterns, it is difficult to identify individual elements. Occlusions may make it difficult to determine which imaged stripe center corresponds to which illuminating stripe [19] therefore complex image processing software must be able to distinguish between two different lines [13]. Solutions to this challenge currently being researched

include a technique called Multistripe Laser Technology (MLT) [11, 12], and techniques which use different colours to disambiguate the stripes [12, 13]. Sensitivity to ambient light remains an issue, and certain assumptions have to be made about the object's surface smoothness in order to use these techniques [13, 19].

Examples of commercial developers of active structured light scanning systems are: Inspeck (Quebec, Canada) which develops Talbot-effect-based digitizers that use white light projectors instead of laser light [13] and Steinbichler Optotechnik (Neubeuern, Germany) which makes the Tricolite system, another white light stripe projection system using an LCD projector and CCD camera [9]. 3DMD (Atlanta, GA) [10] is an example of a company which combines structured light techniques with stereoscopic imaging (see Section 2.1.2) for their 3D scanners, which are made specifically for biomedical purposes, for example, visualization, measurement, and surgical planning for the face, head, torso, and body. Their approach is to use an *active* stereo technique whereby a unique random light pattern is projected onto the surface and used as the foundation for triangulating the geometry in 3D from images.

Time-of-flight range finding

Time-of-flight devices also use laser light, but in this case, the laser is used as a range finder. The scanner emits a focused pulse of laser light towards the object and times how long it takes for it to return to the sensor. Since the speed of light is known, the time it takes for the light to return is used to calculate the round trip distance of the light pulse; the distance to the object surface can then be determined [19]. The scanner scans its entire field of view to get the distance profile of an object. Time-of-flight range finders are capable of measuring very long distances, however their accuracy is limited by how accurately the speed of the light pulse can actually be measured.

Amplitude modulation

An amplitude modulating 3D scanning system employs a continuously operating laser beam with its power modulated sinusoidally over time. When the laser beam reflects off the object's surface and returns to the sensor, it still has sinusoidal power variation however, it is now out of phase with the emitted laser beam. The roundtrip distance to the object surface can be computed by calculating the phase difference between the emitted and reflected power signals. [19]

2.1.4 3D Models and their applications

The data returned by these 3D scanning technologies can come in different forms depending on the technology used. Most 2.5D systems return what is called a *range image*, which is similar to a conventional camera image, except rather than each pixel storing a colour or intensity value, they store a depth value [19]. Multiple range images are then merged in order to create a complete 3D surface. Data from 2.5D scanners can also be just a set of (x, y, z) points in 3D space, referred to as a *point cloud*. The product of merging multiple range images is also a point cloud. A 3D model is created by connecting these data points with their nearest neighbors to form triangular or polygonal faces [19].

Once a full geometric model is constructed from the range data, it may be used for a number of purposes. Industrial applications include part inspection, reverse engineering, and rapid prototyping [1,2]. Reverse engineering refers to the process of creating Computer Aided Design (CAD) models from an existing part so that it can be copied and re-manufactured [2]. Rapid prototyping and 3D printing refers to making high resolution hardcopies of objects using technologies such as stereolithography [19]. 3D scanning technology is also having a major impact on the entertainment industry. These scanners are instrumental in the development of realistic computer graphics in

the video gaming market [1], not only by facilitating the capture of existing objects but also by allowing sculptors to work in their preferred medium (such as clay) while still being able to bring their sculptures into the computer [19].

The applications of 3D models most relevant for this thesis are those recently emerging in the biomedical field. These will be discussed in more detail in Section 2.3 in the context of laser light-sectioning scanning in particular.

2.2   Active Scanning: The Laser Light-sectioning Technique

The 3D scanning method chosen for this research, the laser light-sectioning technique, is a form of active optical triangulation in which the laser, fitted with line optics, creates a horizontal light plane. The theory behind optical triangulation was presented in Section 2.1.3, however to briefly explain how it is adapted to the laser light-sectioning case, the technique involves projecting a plane of laser light onto the object and then capturing images of reflected light where the object's surface profile intersects the laser plane. By determining the geometric parameters of the scanner setup, either explicitly or through a calibration process, the (x, y, z) coordinates of the illuminated points lying along the object's surface can be computed in a trigonometric fashion [13]. Technical details of how this technique is implemented for the 3D scanner presented in this thesis will be provided in Chapter 3. The remainder of this section will provide a background on the state of the art in the area of laser light-sectioning.

A review of many different 3D laser light-sectioning digitizers is provided in [14], with comparisons given for accuracy and speed. Most of the scanners presented are research initiatives under development, although a few commercial systems are

mentioned. The author proposes a classification scheme for these laser light-sectioning scanning systems. They describe *raw acquisition* systems as those where no encoding techniques have been applied; that is, one image is captured by a frame grabber for each new position increment of the scanning mechanics [14]. The scanner presented in this thesis would fall under the category of raw acquisition. Raw acquisition scanners evaluated in [14] include those presented in [22] and [23] and will be discussed more in Chapter 4 within the context of calibration. The other classifications given for laser light-sectioning methods are *space-encoded* (the projection of successive binary patterns onto the scene) and *smart sensors* (with on-chip range computation). Most scanners discussed hereafter will fall under raw acquisition systems or smart sensors.

There are many commercial 3D scanners available which employ the laser light-sectioning technique. A few of these scanners will be highlighted here; many of these scanners have been chosen specifically because they have been used for various biomedical purposes, both in practice and in research. Table 2.1 provides a summary of these scanners, highlighting their cost and, if applicable, examples found in the literature of their use in biomedical applications. These applications will be discussed in more detail in Section 2.3.

The first three of these scanners appeared in a recent review of current commercial 3D digitizers [13], which gave special consideration to active laser triangulation based systems. One popular 3D scanning company is Konica Minolta Inc. (Osaka, Japan) [29] who produces both the Vivid 700 and 910 laser light-sectioning scanners. These scanners, which sell for about $60,000 USD are "mobile and handy" [25] with applications in a variety of fields, ranging from industrial engineering to the survey of works of art. These systems offer the flexibility of connecting several scanners by means of a Small Computer System Interface (SCSI) link in order to record multiple range scans simultaneously. This of course increases the operating costs significantly. As shown in Table 2.1 these scanners have been particularly popular in the biomedical

Table 2.1: Summary of Commercial 3D Scanners.

Company	Scanner	Type of technology	Intended applications	Biomedical uses	Cost(USD)
Konica Minolta Inc	Vivid 700 and 910	Laser light stripe	Various (not biomedical specific)	Face (orthodontics [24]; plastic and craniofacial surgery [9, 20, 25, 26]; biometrics [27])	$60,000
Cyberware	WB4, WBX	Laser light stripe	Whole body	Whole body (anthropometrics [8]; apparel [3])	$150,000-$350,000
Metris	ModelMaker	Laser light stripe with digitizing arm	Various (not biomedical specific)	None found	unknown
Polhemus	FastSCAN	Handheld laser light stripe	Various (incl. biomedical)	Face (biometrics [27]; facial deformation research [6]), whole body (apparel [3])	$21,50-$27,500
Shape-Grabber	Ai310, Ai810	Laser light stripe	Various (not biomedical specific)	Cartilage research [7]	$99,000-$199,000
Next-Engine	Desktop 3D Scanner	Multistripe Laser Technology	Various (incl. biomedical)	None found	$3,500-$5,000
3D Digital Corp	EScan	Laser light stripe	Various (incl. biomedical)	None found	$7,400
Vitronic	Vitus	Laser light stripe	Biomedical (whole body, head, foot)	Whole body (anthropometrics [8]; apparel [3])	unknown
Creaform	ERGOScan	Handheld laser light stripe	Biomedical	Prosthetic design [28]	$40,000
3DMD	3dMDface, cranial, torso, body	Active stereophotogrammetry	Biomedical	Face (biometrics [27])	unknown

field, despite the fact that they are not developed specifically for this purpose.

Another popular 3D scanning company described by [13] is Cyberware (Monterey, CA) [30]. Although their systems can capture photorealistic images of a wide range of objects, they are probably most noted for their fullsize human body scanners. Cyberware systems are "complex, not portable, and prohibitively expensive for many applications" [13]. Depending on the application, the exceptional quality offered by these systems might not outweigh the need for a simple, economical and portable solution.

The last commercial laser scanning company evaluated by [13] is Metris (Leuven, Belgium) [31] which produces ModelMaker, a laser-stripe scanner head that can be attached to a mechanical digitizing arm. An object is scanned by manually moving the laser head around the object. Although the system is straightforward to use, acquisition time is in the order of 10 minutes due to the manual manipulation required [13]. The ModelMaker system offers submillimeter accuracy, and the option of using a rotational platform alleviates the dimensional restrictions of the mechanical arm [13].

Other 3D scanning companies which have been researched include Polhemus (Colchester, VT), ShapeGrabber (Ottawa, ON), NextEngine (Santa Monica, CA), 3D Digital Corp (Sandy Hook, CT), Vitronic (Wiesbaden, Germany), and Creaform (Lévis, PQ).

Polhemus [32] develops handheld laser-stripe scanners for a wide variety of applications. Their light-weight and portable FastSCAN device provides real-time visual feedback for straightforward monitoring and control of the scan process. One caveat with handheld scanners is that they do require some training and skill in order to use them effectively, otherwise the scans may turn out incomplete [27]. ShapeGrabber [33] develops industrial laser scanners which are mostly used to inspect complex parts for design or quality control purposes, or to produce computer models of objects for reverse engineering. Accuracy of these systems therefore needs to be extremely

high, which of course is reflected in their cost. Their two main models are $99,000 USD and $199,000 USD which include a computer, software, training, and maintenance. The licence for the software alone is $16,000 USD, a good portion of the entire cost. Some biomedical applications are reported on their website [33], such as the inspection, modeling, and manufacturing of prosthetics and dental components.

Two of the most economical scanners found to date are developed by NextEngine [34] and 3D Digital Corp [35]. The NextEngine Desktop 3D Scanner system, designed for small to medium sized objects, uses proprietary Multistripe Laser Triangulation (see Section 2.1.3). Their system sells for $2,500 USD and offers an accuracy of about ±0.4mm. The accompanying software for the scanner costs an additional $1,000 to $2,500 USD. 3D Digital Corp.'s EScan system, a lightweight, accurate, and quick solution for scanning medium sized objects, sells for about $7,400 USD. Both of these systems sit on a table or tripod and therefore only capture a frontal 3D profile; multiple scans have to be merged to form a complete 360° model.

Vitronic [36], originally an industrial machine vision company, produces the Vitus series of scanners for scanning the body, head, and foot. These scanners are designed more for high volume scanning applications where accuracy is not as high of a priority; scanner resolution is in the range of 1 - 5mm. Examples of applications for these scanners would be in a fitting room for taking measurements for custom clothing or uniforms or for creating models of the human body for ergonomic design purposes.

Lastly, Creaform [37] is a producer of handheld laser scanners for a variety of purposes. One such model, the ERGOScan, is made specifically for biomedical purposes such as orthotic and prosthetic design, plastic surgery planning, and cranial scanning. This user-friendly, highly-portable device offers accuracies of up to ±0.5mm at a cost of about $40,000 USD. Note that 3DMD (see Section 2.1.3) has also been included in Table 2.1; although their technology is not laser-based, they produce a variety of scanners specifically made for biomedical purposes and are included here

for comparison.

2.3 Biomedical Applications of 3D Laser Light-sectioning Scanners

There are numerous applications in medicine where a 3D model of the external features of an anatomical part is beneficial. Traditionally in medicine, CT and MRI technology have been used for recording and visualizing different anatomical structures in 3D, however these tools are expensive to operate, provide more detailed information than would be required for a 3D surface geometric model (i.e., internal features), and involve elaborate and time consuming procedures. Furthermore, CT exposes the patient to harmful radiation and is therefore not safe for use in routine procedures. "Examination methods that can record 3D changes of body surface structures precisely, rapidly, and without side effects or major inconvenience for the patient would be of great benefit" [25]. Hence, the shift towards 3D scanning technology for acquiring 3D models of anatomical structures.

In general, 3D laser scanning provides a less invasive, non-ionizing, economical, and fast alternative to other imaging modalities. Although there is a wide range of costs for various laser systems, they are still generally less expensive and more flexible than full 3D imaging systems (i.e. CT, MRI), while obtaining much more densely sampled shape data than completely passive systems (i.e. stereoscopic imaging) [1]. Compared to traditional methods, laser scanning systems also offer the advantage of relatively faster acquisition times. 3D models from laser scanners are also a valuable communication tool; realistic visualizations are not only useful for consultations between physicians but also facilitate communicating with patients about certain aspects of their condition and/or planned treatment [5].

The most common applications of 3D models in practice are in the area of or-thodontics and surgery planning. These and a few other examples are summarized here to highlight the *breadth* of research and diagnostics that are now being conducted with the use of 3D scanning techniques. If applicable, the specific scanner used (as shown in Table 2.1) is also described.

Orthodontics and oral maxillofacial surgery planning

Having computerized 3D models of different facial structures is a huge benefit for many orthodontic and surgical applications. A good review of applications of 3D imaging in orthodontics, as well as in oral and maxillofacial surgery can be found in [5]. Some applications include: "pre- and post-orthodontic assessment of dentoskeletal relationships and facial aesthetics, auditing orthodontic outcomes with regard to soft and hard tissues, and 3D treatment and surgical planning" [5]. Computerized 3D models of different facial structures can also assist in the fabrication of custom-made archwires and other orthodontic devices.

Schwenzer *et al.* [9] performed a study to investigate the accuracy and precision of two commercial 3D measurement systems, one employing a white light technique and the other using laser, in order to evaluate their appropriateness in biomedical applications. The systems tested were the Vivid910 laser scanner from Konica Mi-nolta and a Tricolite white light scanner from Steinbichler Optotechnik (Neubeuern, Germany). "Both systems are unobjectionable with regard to health" [9] however, when using the Vivid910 scanner, for a complete facial image covering 270°, up to four views would be required which adds to acquisition and registration time. See [9] for other advantages and disadvantages of these scanners in biomedical applications.

Kusnoto *et al.* [24] assessed the accuracy of generating 3D models of both a dental model and a plaster facial model using the Vivid700 3D surface laser scanner (also by

Konica Minolta). Measurements taken from the dental model of two dental parameters, intermolar width and palatal vault depth, were shown to be accurate to 0.2 mm and 0.7 mm respectively [24]. These findings will be useful for comparison with the proposed scanner in this thesis, as will be discussed in Chapter 8.

For applications in plastic and reconstructive surgery, Kovacs *et al.* [25] evaluated the precision and accuracy of the 3D assessment of a complex artificial surface replicating the human face, again with a Vivid910 3D laser scanner. The goal was to provide recommendations concerning possible clinical applications of the scanner [25]. Due to the fact that Minolta scanners are not specifically designed for biomedical applications, they looked at factors related to the recording procedure and the processing of scanner data on final results including lighting, alignment of the scanner and object, the examiner, and the software used to convert measurements into virtual images [25]. They also performed a follow-up study which looked at the same factors for real human faces [26]. Overall, "accuracy proved to be good enough to satisfy requirements for numerous clinical applications" however the fact that the device cannot capture a 360° view in one session was deemed a disadvantage [25]. Although multiple scanners can be linked for this purpose, they suggest it might not be widely adopted in practice because of the considerable cost of purchasing and operating more than one scanner [25]. One case where multiple Vivid910 scanners *have* been used is in a research project by Rana *et at.* [20] who are working on developing a craniofacial surgical planner using input from CT, 3D laser scanners, and photogrammetry. They have used two linked Minolta Vivid910 scanners for this system. Through this multimodal approach, they hope to be able to reconstruct human hard and soft tissue along with anthropometric landmarks for a complete surgical planning aid. The role of the laser scanners in this case is to capture facial skin surface soft-tissue data. Two scanners are used in order to get a more complete model of the face, however even with this setup, they do not get a full 360° model.

Anatomical parts reconstruction for reverse engineering and prosthetics

A recent article in Mechanical Engineering profiled the reverse engineering applications of 3D scanning in the biomedical field. 3D scanners can be used "to create the prosthetic twin to a body part lost to cancer or trauma, such as a hand or an ear" via rapid prototyping processes [28]. The ERGOScan handheld device, by Creaform (Levis, Quebec), was "recently used to create a natural looking prosthesis for a patient who lost the left half of her nose to skin cancer" [28]. A prosthesis was created using stereolithography based on a model of the remaining half of her nose, and the 3D model was used in the planning of her rhinoplasty operation. The FastSCAN device (by Polhemus) has been used for fitting patients with prosthetic and orthotic devices in partnership with Hanger Orthopedic Inc (Bethesda, MD) [38].

A research group in Italy, Tognola *et al.* [4,39–41], is developing a laser scanner for anatomical parts reconstruction as well. The culmination of this work is a method for acquiring a 3D point cloud of an anatomical object's surface as well as a novel surface reconstruction algorithm. Their system employs a HE-NE laser, and the MaxReflex motion capture system (two CCD cameras and a real-time video processor) from Qualysis (Gothenburg, Sweden). This equipment setup was chosen because it is common in biomedical laboratories designed for motion capture and gait analysis where the 3D position of passive reflective markers would be captured rather than active laser reflections. Surface scanning in this case is performed by manually moving a laser beam all over the object surface while the two digital cameras and video processor measure the 3D position of the dot. This system will also be compared to ours in Chapter 8.

Face scanning for facial deformation research and biometrics

3D scanners can provide models for studying asymmetrical facial structures, functional and facial soft tissue movements, and differential facial growth, for monitoring progressive skeletal pathologies more objectively [13, 24]. A study, which used the Polhemus FastSCAN scanner, has shown that facial dysmorphogenesis, or facial deformations, "can serve as an accessible and informative index of brain dysmorphogenesis in neurological and psychiatric disorders of early developmental origin" [6], such as Down's syndrome (a chromosomal abnormality), as well as in less readily recognizable conditions such as velocardiofacial syndrome and even in schizophrenia [6]. Other applications of 3D laser imaging to face and brain studies include: facial anthropology, facial growth, fetal alcohol syndrome, clinical malocclusion, and sexual dimorphism [6].

Biometrics is also an area of interest when it comes to face scanning. Boehnen *et al.* [27] studied the accuracy of 3D scanning technologies in a face scanning scenario by reviewing several 3D scanners, many of which are general purpose scanners not designed specifically for faces, particularly the Vivid910 from Konica Minolta, and the handheld FastSCAN from Polhemus (which included a special attachment for scanning the face even though it is a general-purpose device). A 3DMD scanner was also used for this work. The most accurate scanner reviewed (in every category) was the Vivid910, followed closely by the 3DMD and Polhemus scanners. The difference in accuracy between the Minolta and 3DMD was small. The advantage of the 3DMD system is that it requires only milliseconds to take its scan, which is especially important with live moving subjects, but the sampling is much coarser. While the Polhemus was "not as accurate as the 3DMD and Minolta scanners, its portability opens other potential uses" [27]. Using this type of handheld scanner does require some skill to use [27].

Cartilage research

A study presented in [7] provides a method to accurately assess articular cartilage morphology using 3D laser scanning technology, potentially a very valuable tool in the study of osteoarthritis (OA), a degenerative disease of cartilage in articulating joints. They used the ShapeGrabber (Ottawa, ON) laser scanner for acquiring 3D models of articular cartilage. Being able to accurately assess different aspects of cartilage morphology, such as total volume and average thickness as well as more localized parameters such as local thickness and surface curvature are of significant interest in OA research for studying disease progression of OA and assessing the effectiveness of treatments.

Whole body scanning for anthropometry

CAESAR, which stands for Civilian American and European Surface Anthropometry Resource, was the first large scale 3D anthropometry survey project conducted using 3D scanners. About 6,000 civilians in the USA, the Netherlands and Italy, were scanned to create this anthropometric database. The Cyberware [30] and Vitronic [36] full-body scanners were used. 3D scans of entire human bodies contain much more information about the human body shape than the traditional anthropometric measurements and provide opportunities to extract new measurements [8].

Whole body scanning for apparel

There is significant potential for the use of 3D models in the apparel industry, particularly in cases where custom-made garments are required on a large scale (i.e. military) [3]. A review of whole body scanners can be found in [3], where the specific goal was to compare specifications of different systems in order to provide some direction for further research into the use of these technologies for the apparel and

custom fitting industry. Both light-based (white and LED) and laser light-sectioning systems were explored; the laser-based systems included WB4 and WBX by Cyberware, FastSCAN by Polhemus, and Vitus by Vitronic. The main disadvantage of 3D scanning technology, compared to traditional anthropometry or tape measurement methods, is the cost of the technology [3], about $150,000 to $350,000 for whole body scanners.

Until 3D scanning began to be employed, this was a time consuming and laborious task. For example, a 1988 anthropometric survey of US Army personnel required four hours physically to landmark, measure and record the data of one subject [3]. This technology "has the potential for obtaining an unlimited number of linear and nonlinear measurements of human bodies" [3] which would be much more precise and reproducible than measurements obtained through a more physical measurement process. The US Army now uses the Cyberware WBX system to develop and analyse body shapes for armour coverage and for other military uniform clothing [3].

2.3.1 Challenges of using industrial scanners in biomedical applications

"In order to apply surface scanners to medicine, one usually has to resort to commercially produced systems originally designed for industrial use" [9]. Despite the fact that "these devices can be considered as the state of the art of scanning technology for their remarkable accuracy, resolution, and velocity, they may present some limitations" [39] when adopted for biomedical applications. The special characteristics of vital tissues include many aspects rarely encountered in artificial objects (e.g., great variations in texture, tension, transparency, and reflectivity) that have to be investigated as possible sources of error in the clinical use of the scanning technique [25]. Also, measurement errors due to patient movement, even just from breathing, are

inevitable, meaning the accuracy levels that these scanners are capable of attaining (sometimes at a micron level) are not realistically achieved in a medical scenario [27]. Furthermore, the "variety of scanner systems and types available is a major obstacle to giving general recommendations concerning the use and benefits of scanners in medicine" [25].

This is why there has been so much work done to evaluate certain commercial scanners in order to determine their precision and accuracy in defined medical applications (see [3, 9, 24, 25] as discussed above). Many company websites advertise the effectiveness of their scanners for biomedical purposes by presenting case studies, much like the ones summarized above, of where their scanner has been shown to be useful for a particular scenario. Only very recently have scanners emerged on the market which are catered specifically to healthcare and biomedical research institutions (i.e. 3DMD, Creaform, Vitronic).

2.4 Motivation for this Research

Although many of the studies presented in Section 2.3 have concluded that certain available commercial scanners are useful for biomedical applications, and the advantages of using the laser-light sectioning technique are apparent, there are still many aspects of this technology that can be improved upon in order to promote its use for biomedical purposes even further. This is the underlying motivation for this research. Based on the literature reviewed, a list of design considerations for a biomedical-specific 3D scanner has been compiled and is presented in Appendix A. However, as it has not been the main focus of this thesis to address *all* of these issues, only those particular aspects of laser light-sectioning technology that we hope to improve on with this research are listed here:

1. cost effectiveness;

2. simplicity of design;

3. automation of data acquisition steps where possible;

4. reliable and robust processing of data;

As shown in Table 2.1, currently available commercial 3D scanning systems employing the laser light-sectioning technique have been found to cost anywhere from $3,500 to $350,000 USD. Since cost is often a major roadblock in many healthcare institutions, this was a key motivation for this research. By including more complexity in software algorithm development, the use of simple, inexpensive, off-the-shelf components lends itself to a more economical design with a straightforward setup. Automation of the acquisition process was also a key motivator, to help reduce the need for user intervention. Finally, since the complexity (and cost) of many commercial scanners increases in order to produce highly accurate data, and since noise in measurement data is inevitable in many biomedical scenarios, the final motivation was to explore reliable and robust data processing options that would address noise in the measurement data and which would be particularly suited for biological objects.

Chapter 3

System Overview

In this chapter, an overview of a general system design for a laser light-sectioning scanner is presented. A review of relevant literature describing design possibilities for each of the various system stages (calibration, image acquisition, image processing, data manipulation, and object reconstruction) is provided in Section 3.1. More emphasis is placed on those areas where there is significant focus for this thesis. This background research provides the basis for a more refined laser light-sectioning system design, which is presented in Section 3.2. The various system stages (calibration, image acquisition, image processing, data manipulation, and object reconstruction) for the proposed system are described.

3.1 General System Design Overview

3.1.1 General hardware setup

A typical system employing the laser light-sectioning technique would have the following setup (Figure 3.1). A laser fitted with line optics creates a horizontal light plane (i.e. parallel to the xy-plane), the trace of which is visible when projected onto the object of interest. The image of this trace is recorded with a CCD camera, which

is inclined at a fixed angle α, relative to the xy-plane. The goal of each measurement is to obtain one segment of one planar section of the object. Multiple planar segments are obtained by rotating the laser and camera about the object, yielding a complete planar cross-section. Multiple 360° planar sections are stacked upon a vertical axis giving a reconstruction of the object.
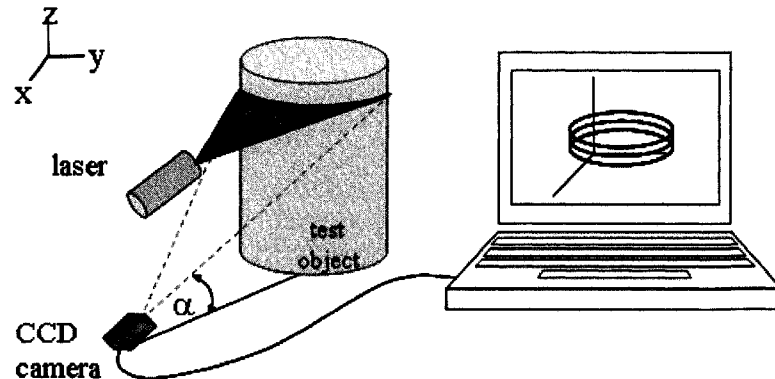
Figure 3.1: Laser light-sectioning system diagram

The main design decision to be made in terms of the hardware setup is the angle at which to place the camera. The system's measuring range and resolution will be determined by the triangulation angle between the plane of the laser line and the optical axis of the camera [42]. The larger this angle, the larger the observed lateral displacement of the line, which means objects with finer detail can be scanned (i.e., resolution is high). For objects with large variations in surface topography, a smaller triangulation angle must be used to increase the measurement range; the tradeoff in this case is poorer resolution. [42]

3.1.2 General image acquisition and data processing steps

The general image acquisition and data processing steps for obtaining a reconstruction of the exterior features of an object are summarized in Figure 3.2 and listed below:

1. **Calibration:** the system must first undergo a calibration procedure to determine a transformation matrix H which will compensate for the projective distortion of the camera.

2. **Image Acquisition:** M views of the laser trace on the object are then captured, separated by rotations of $R = \frac{360^\circ}{M}$ about the vertical z-axis. This process is repeated for n planar cross-sections in vertical increments along the z-axis.

3. **Image Processing:** images undergo processing in order to isolate the laser trace segments within the image, and determine a set of point coordinates to represent the trace.

4. **Data Manipulation:** laser trace point coordinates are transformed using the transformation matrix obtained from the calibration procedure. Then the segments are registered into a common coordinate system by rotating them to their respective orientation and stacking them along the vertical z-axis. Curve fitting is applied to the segments of each planar section to form entire 360° outlines of the external profile of the object, resulting in a 3D point cloud representing the object's surface.

5. **Object Reconstruction:** a 3D surface mesh is applied the point cloud in order reconstruct the final 3D model.

These general data acquisition and processing steps are each described in more detail below with relevant references to the literature.

Review of calibration techniques

For the general light-stripe case shown in Figure 3.1, the result of projecting a plane of laser light onto an object is an illuminated stripe where the plane intersects the object's surface. This is captured by a camera in a 2D image. Since the camera
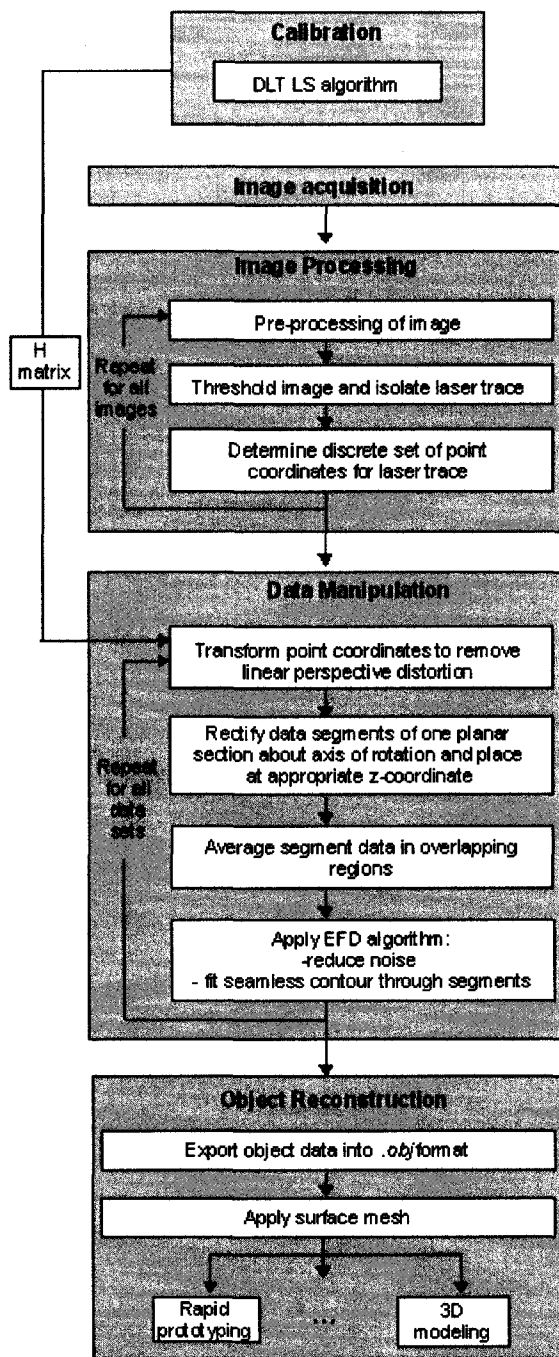
Figure 3.2: Flow diagram of general image acquisition and data processing steps for the laser light-sectioning technique.

used to capture images is located at a fixed angle α to the laser plane, the images contain perspective distortion. A geometric transformation can remove the linear part of this distortion. A calibration procedure is required to find the transformation matrix which will eliminate the linear portion of the camera perspective distortion for points lying along the imaged laser traces (in the xy-plane). Therefore, the basic goal of calibration is to find the geometric relationship between the illuminated pixel coordinates and the world coordinates of the corresponding object's surface points [22].

This section provides an overview of some of the work being done in the area of camera calibration, specifically for applications involving 3D scanning using laser light-sectioning. The most standard case of calibration is where the world points lie on a 3D surface. As a result, there is quite a bit of literature on this non-coplanar camera calibration [43]. Not as much attention has been given to coplanar camera calibration, where the world points lie on a 2D plane, because it tends to be the less frequent but more complex case [44]. The case of laser light-sectioning is one example of where coplanar calibration techniques can apply.

Chatterjee *et al.* [44] present a review and analysis of several different algorithms for coplanar camera calibration, some of which use linear methods (meaning the parameters are computed by solving a linear system of equations) and others which use non-linear methods. Non-linear approaches provide optimal results, where 'optimality' refers to certain orthonormality constraints which the extrinsic camera parameters must satisfy. Linear methods are computationally efficient but lack accuracy in the camera parameter estimates producing sub-optimal results. If one is concerned with finding optimal solutions of *all* camera parameters, then linear methods are non-ideal because these methods cannot explicitly compute all calibration parameters [44]. Salvi *et al.* [43] also present a detailed review of some common calibration algorithms, both linear and non-linear. They also conclude that the advantage of

linear techniques is the simplicity of the models leading to simple and rapid calibration whereas non-linear optimization would be required to get accurate and optimal estimates of individual camera parameters [43].

As a further simplification to the coplanar case, Chen and Kak [22] propose that when a camera is used in conjunction with a light stripe (the laser light-sectioning case), it is possible to obtain the transformation relationship without having to worry about the low-level details such as the precise locations of the camera and/or laser, nor the intrinsic camera parameters. They show that if the fundamental theorem of two dimensional projectivity is applied, "each image point can uniquely determine the world coordinates of the corresponding object point" [22] without the need to calculate the sensor's position or intrinsic parameters explicitly. The transformation from image coordinates to world coordinates involves a set of linear equations which can be solved analytically or by using least squares [23]. This approach is beneficial since the exact camera and laser position are difficult (or near impossible) to measure manually and intrinsic parameters require very complex methods to estimate with sufficient accuracy (as explained above) [23]. In fact, many strongly discourage manual pose measurements since they can induce very significant inaccuracies in the shape measurements [14]. Note that this approach is only valid under the condition that it be possible to use the *pin-hole model* for the camera [22]. Since a camera is essentially a mapping between the 3D world and a 2D image, a camera *model* is used to describe the mathematical relationship between the coordinates of a 3D point and its projection onto the image plane of the camera. The common pin-hole model treats any camera as an ideal pin-hole camera; that is, a camera where the camera aperture is described as a point and no lenses are used to focus light [45]. Although the model does not include geometric distortions, it is considered to be a reasonable description of how a camera depicts a 3D scene. It is used in many computer vision applications and its validity increases with camera quality.

Building on the work of Chen and Kak, Reid [23] proposed a projective calibration method for their laser light-sectioning range system for a robotic vehicle, which models a projectivity between two planes; the plane of the laser stripe and the plane of the camera's detector array [23]. Control point pairs are acquired from orthogonal objects placed within the scene. This simple, elegant and accurate calibration technique is applicable to many other light-stripe sensors [23]. [46] also uses the projectivity ideas of Chen and Kak for a laser light-sectioning system, whereby the system is first calibrated by placing a calibration grid in the scene to compute the transformation before an object is imaged.

Again, all of these linear methods involve solving for parameters with systems of linear equations. One drawback of linear techniques is that they are not capable of non-linear lens distortion modeling [43] and the parameters obtained do not correspond to the physical parameters of the camera itself. However, Chen and Kak have shown that these distortions only become important for object points that are far away from the camera lens (usually farther than two feet) [22] which is beyond the distances considered for the proposed scanner in this thesis. If needed, lens non-linearities may be taken into account by a variety of techniques, some of which have been summarized by [43] and [44].

As mentioned, these linear systems can be solved both analytically, or with least squares methods. The method proposed in [46] uses an analytical approach, whereas that in [23] uses least squares. One method of finding a least squares solution is the Direct Linear Transformation algorithm presented in [45]. This algorithm has been used in many other calibration and registration scenarios. For example, in [47] two uncalibrated CCD cameras and the DLT algorithm are used to obtain the relative relationship between two neighboring assembly devices. Also, [48] presents an efficient approach for medical image registration; alignment between two medical images with various forms of geometric distortions is achieved by using the DLT algorithm for

parameter estimation of a projective transformation matrix.

One problem with using the DLT approach directly on a set of candidate control points is the fact the DLT algorithm is a least-squares estimation method and therefore performs poorly in the presence of outliers [48]. Far outliers can have an enormous effect on the estimated transformation model because the error is squared [48]. In the case of calibration for a laser light-sectioning system, outliers would result from mismatched control pairs and would arise more frequently when the control pairs are determined by automated processes.

As a final note on calibration, a precise calibration method can allow for lower quality and cost components to be used while still achieving high-quality scans. As long as the scanner's mechanical stability is maintained, then the imperfections in alignment will be quantified and compensated for through the calibration procedure [13]. Also, keeping in mind that calibration complexity would increase costs, the design can be simplified by using a more manual calibration process. In the specific case of biomedical applications, the calibration procedure should ideally be simple, so that any user can perform it [13].

Review of image acquisition techniques

Different laser light-sectioning systems use different approaches for image acquisition. The object can be rotated on a turntable in fixed degree increments while the laser and camera remain fixed [33]. Alternatively, the laser and camera head can be rotated about the object, while still remaining fixed relative to each other. Some systems sweep the plane of laser light across the object using mirrors controlled precisely by servo or stepper motors [23, 29, 30, 34, 35]. Others require that the laser plane be swept over the object manually using a hand-held device or a mechanical arm which tracks its location [31, 32, 37]. Handheld systems are portable and flexible, but many incorporate a magnetic tracker used as a 3D frame of reference so their accuracy

may decrease in environments containing metal objects or electromagnetic fields due to interference with the tracking system [4]. Some measurement systems aim to incorporate the Next Best View (NBV) problem, which is to determine the next best position of the scanner in its workspace such that some previously unseen portion(s) of the viewing volume will be captured given the range data already acquired [2].

Regardless of the technique used, all of these areas of research are a result of the simple fact that 3D laser light-sectioning scanners cannot scan through an object, and so to acquire a complete surface description of an object images of where the laser plane intersects the object must be taken from different vantage points such that the entire 360° view of the object is captured.

In the specific case of biomedical applications, the image acquisition technique should involve as much automation as possible so as to lessen the requirement for an expert user, as in the case of the manually swept scanners (i.e., [31,32]). Acquisition time should also be as rapid as possible to help reduce human body movement artefact when taking measurements [3].

Review of image processing techniques

The goal of the image processing stage is to determine a discrete set of point coordinates to represent the laser trace. Unfortunately, the line in the image is in most cases diffuse and spread out over many pixels [49]. Figure 3.3 shows an image of a test object with the laser trace projected onto it and Figure 3.4 shows what the trace would look like under ideal lighting conditions for image capture. As shown in Figure 3.4, although the laser trace is very thin, it can span multiple pixels; determining a set of coordinates that best traverse this trace can thus result in quantization error.

The first step is segmentation, which is the identification of the laser trace object within the image. The image is segmented by applying a threshold in order to isolate
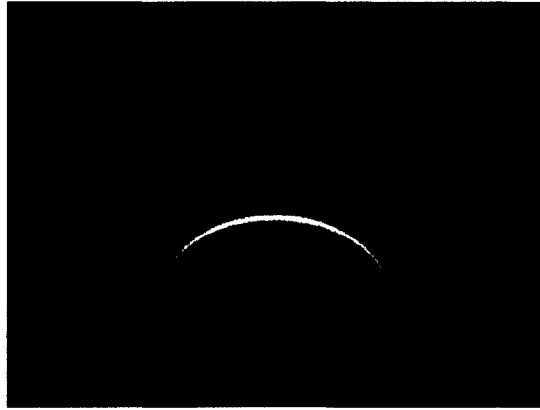
Figure 3.3: Sample object with laser trace.

the laser trace from the rest of the image. For images with ideal lighting, good contrast, and little noise, "it is sufficient to apply a simple threshold value to extract the object from the image" [49]. Otherwise, images acquired must first undergo preprocessing in order to condition the images for effective segmentation. This could involve background removal techniques or contrast adjustments. Segmentation can result in an image where everything below the threshold is set to one intensity value and everything above is set to another (i.e., a binary black and white image). Segmentation can also mean that only those pixels above the threshold are considered for further processing. This can be accomplished with line walking algorithms [49] whereby an area of interest is computed by walking a pre-defined search column along the image, scanning the next column for the maximum value pixel, centering the search column on that pixel and retaining those above and below it. This continues until no maximum value above a given threshold is found [49].

Once the method of segmentation has been chosen, the last step of the image processing stage involves determining a discrete set of point coordinates to represent the laser trace. Much research has already been done in this area. A good review of algorithms which can be used for determining point coordinates from laser trace images captured with a laser light-sectioning system is provided in [49]. Some
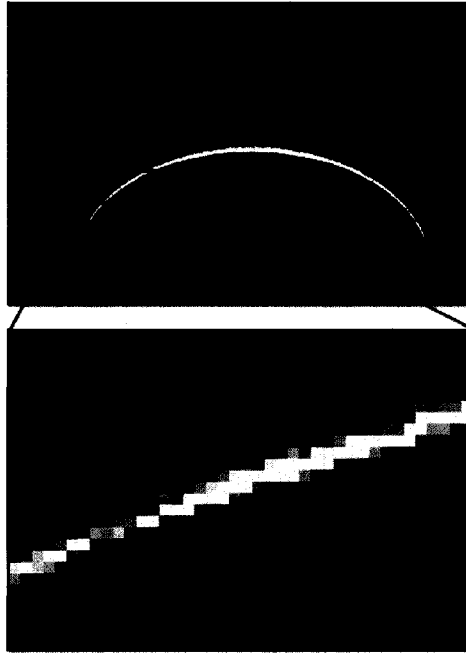
Figure 3.4: Diagram showing laser trace thickness at a pixel level.

examples of techniques for this step include line thinning algorithms [50, 51], peak detection [49,52,53], and center of gravity techniques [13,49]. Preferably, the method chosen will find the position of the centre of the laser trace with subpixel accuracy [13]. The simplest form of peak detection detects the maxima along each vertical line within the image. However, "simply calculating the maximum intensity along a cross section of the laser line will not result in reliable detection of the centre of the line" [49] due to certain interference with the laser and the object surface which can lead to artificially bright points. Centre of gravity techniques can achieve subpixel accuracy because by measuring the pixel signal intensities and then calculating the center of gravity coordinate, the result will not necessarily lie on a specific pixel. With a line thinning approach, the trace is usually thinned to a line one pixel thick, but the choice of that one pixel can sometimes result in a bias, as will be discussed further in Chapter 5.

Review of data manipulation techniques

The transformation matrix found during the calibration process is used to transform all sets of point coordinates, removing the linear portion of the projective distortion. As described above, many views are necessary to acquire the object's shape from all sides. The next stage of data manipulation is to place all the data sets into the same coordinate system in order to merge all of the scans into a single shape. The problem of finding each of the rigid transformations required to rotate and translate the data sets into this common coordinate system is called *registration* or alignment. For 3D laser scanners which produce range images (see Section 2.1.4) as their output these transformations must be computed using the range data itself by iteratively computing the transformations that will align any two given scans most closely. Range image registration can be a very complex and computationally expensive task. See [54–56] for examples of work being done in the area of range image registration.

If the 3D scanner is under precise computer control or is carefully tracked however, then the registration problem is more trivial. This is the case for the laser light-sectioning system shown in Figure 3.1, since the rotations are controlled as well as the vertical increments of the camera/laser platform. For a given plane, the M segments of point coordinates are rotated according to their angular position about the central axis of rotation and assigned z-coordinates which correspond to the proper increment level in order to introduce the 3rd dimension. The data now represent a point cloud because the point coordinates actually form a cloud in 3D space representing the surface of the object.

After registering all data sets into a common coordinate system, ideally all individual points would lie exactly on the surface of the reconstructed object [1]. However, this is not the case due to noise and systemic errors in the data. Therefore, the data

manipulation stage should also include techniques for noise reduction. Residual errors in range data can come from various sources including: effects from using active optical scanning technology, the design of the particular 3D scanner itself, and the object's surface properties [2]. Those due to active optical scanning technology are related to the use of laser and include the "mixed pixel" or edge curl effect [57] and optical blurring. The design of a particular range scanner introduces its own sources of error which can include precision of the camera calibration, the methodology used to obtain coordinate point data for the imaged laser traces, and the alignment process. The object's surface properties introduce noise such as laser speckle (i.e., the interference of light waves emanating from a textured surface), changes in albedo (i.e., surface reflectivity), and concavities which can cause reflections of the illumination source [2].

There are various ways to account for this residual error. Preprocessing techniques could involve removing outliers from the data, filtering the data, or, in the case of a laser light-sectioning scanner with data in planar segments, fitting the individual data segments to polynomials or splines before aligning them about the axis of rotation. Another approach to deal with residual error is to define new estimates of actual surface points by averaging samples from overlapping scans once they have been aligned [1]. Different methods of achieving this are discussed in [1] and include techniques, such as: modeling the measurement error locally and then optimally reconstructing the object surface using the two overlapping sets of data as "estimations" in a probabilistic sense [55], and computing surface values as weighted averages of overlapping range data, where the weight used is proportional to the illuminance received by the sensor [58]. These methods assume range input data and not necessarily data from planar laser light-sectioning.

Finally, one may choose to smooth the data even further by fitting it with a curve. In the case of laser light-sectioning where data is collected in planar sections,

the points for a given planar section could be interpolated by inserting line segments between neighboring points, producing an interpolated reconstruction of the curve. Such a reconstruction would probably still be noisy and may very well have many more line segments than are necessary [19]. A better solution would be to estimate a curve that approximates the shape outline very closely, but does not necessarily interpolate the range data points [19]. Depending on the complexity of the segments, options could include polynomial fitting, splines, or Elliptical Fourier Descriptors [59]. This curve fitting process would be repeated for each of the n planar cross-sections.

Review of object reconstruction techniques

The registered range images or point clouds produced by 3D scanners are typically not used directly. Most applications require a 3D model. The process of converting a point cloud into a polygonal 3D model is called object or surface *reconstruction*. The reconstruction process involves taking all of the range data, precisely registered into a common coordinate system, and connecting adjacent points in order to fuse the data into a single continuous surface. [19]

There has been a lot of research conducted in the area of surface reconstruction. A survey of surface reconstruction methods can be found in [60]. A number of these solutions take advantage of the original structure of the data; that is, they work on range images which have been registered into a common coordinate system (as described above), treat them as surfaces, and merge those surfaces together. This process is called *integration* because although the range images may be registered together, they are still not integrated into one single surface. Examples of work in the area of range image integration are [54, 58]. Unfortunately, like range image registration itself, this 3D stitching process can be quite complex and computationally expensive and "does not always produce high quality meshes" [19].

By contrast, some solutions disregard the structure of the range data altogether

and compute a surface straight from a point cloud of range points rather than merging partial range surfaces [19]. Since data from the scanner described in this thesis is not collected in the form of range images, we require such a surface reconstruction solution. Much work has been done in finding surface representations from sets of unorganized points, including [61,62]. A common approach is to create meshes of connected triangles whose vertices are the points in the clouds. Many of these algorithms are based on the theory of Delaunay triangulation [63], the definition of which ensures that no point in the cloud lies within the circumcircle of any triangle in the mesh (the circumcircle is a circle drawn around a triangle, such that all 3 points of the triangle touch its perimeter). Delaunay triangulations tend to avoid skinny triangles by maximizing the minimum angle of all the angles of the triangles in the mesh. A survey of Delaunay triangulation techniques can be found in [64] and some specific examples of Delaunay triangulation based surface reconstruction algorithms for unorganized point clouds can be found in [65, 66]. Examples of where Delaunay triangulation has been used specifically for point cloud data in the form of planar sections (i.e., from CT, MRI, or a laser-light sectioning scanner) include [67,68]. "Computing the required 3D Delaunay triangulation can be prohibitively expensive in terms of time and memory required" [69] especially for datasets with a large number of points. A relatively new algorithm for defining a triangular mesh, the "ball-pivoting algorithm" (BPA) developed by Bernardini et al. in 1999 [69], aims to circumvent some of the limitations of Delaunay triangulation. "The goal of the BPA is to return the strengths of previous interpolating techniques in a method that exhibits linear time complexity and robustness on real scanned data" [69]. For this algorithm a theoretical 'ball' of fixed radius is rolled over the surface, creating triangles as it encounters new points. Another very widely accepted tool used for surface geometry representation is non-uniform rational B-splines (NURBS) [70,71]; NURBS offer a common mathematical form for representing both standard analytical shapes (such as conics and quadratics)

as well as free-form curves and surfaces [70], as would be the case for unorganized point clouds. The use of Octree-splines has been proposed by [72] for low-level surface representation. Octrees are a data structure which can be used to partition 3D space by recursively subdividing it into eight octants; when combined with a set of 3D points, an octree-spline enables one to build a continuous 3D distance map.

All of the techniques presented above which work from unorganized point clouds "have the merit of being able to compute a surface model using only point information" [58] and avoid the complex range image registration and integration problem discussed in the previous sections. The surface reconstruction method chosen for this scanner would have to have this capability. Many different object surface reconstruction techniques have been implemented in 3D graphics software packages already. When choosing an appropriate solution for object reconstruction it was logical to try and make use of existing software. Keeping in mind the intended biomedical applications of the scanner, a few specific criteria were considered:

- compatibility with data format after data manipulation stage

- accessibility

- cost effectiveness

- versatility of the final 3D model

Possible software applications capable of converting point clouds into triangular meshes include: VRMesh [73] (Seattle, WA), GeoMagic [74] (Research Triangle Park, NC), Meshlab [75] (Pisa, Italy), Cyclone by Leica Geosystems [76] (Heerbrugg, Switzerland), and Polyworks by Innovmetric [77] (Québec, QC) (which often comes bundled with existing 3D scanners such as those form Shapegrabber [33] and Creaform [37]). In terms of cost, VRMesh sells for $895 USD while a licence for Polyworks is about $16 000 USD. Meshlab is open source and therefore free to download.

3.2   Proposed System Design Overview

This section takes into consideration all of the design possibilities discussed above
and presents a refined system design overview for the proposed laser light-sectioning
scanning system.

3.2.1   System hardware setup

The system presented herein (Figure 3.5) consists of a JAI Pulnix (San Jose, CA)
TM-200 High Resolution (768 × 494) charge-coupled device (CCD) camera and a
Stockeryale (Montréal, PQ) Lasiris SNF635S-5 Class IIIa Diode Laser fitted with
line-generating optics. Both are mounted to a platform; the camera is inclined at
$\alpha = 30°$. This angle was chosen to gain a larger measurement range in order to
be able to scan objects with larger variations in surface topography, even though
resolution is slightly reduced with a smaller angle [42]. The object to be scanned is
placed on a turntable approximately 10 cm (4 in) from the camera.

Images are captured with a National Instruments (NI) (Austin, TX) PCI-1411
image acquisition device. Servo motors are used to move the camera/laser platform
vertically and to rotate the turntable in fixed increments. Motors are controlled with
the Lego (Billund, Denmark) NXT controller box (32-bit ARM7 microcontroller, 256
Kbytes FLASH, 64 Kbytes RAM), which interfaces to the computer (and Labview)
via USB. A single Labview (NI) program is used to control the entire system (i.e.,
the servo motors and the image capture device). Matlab is used for offline processing
of the images and data. The computer used is a 3GHz Pentium 4 with 1GB RAM
running Windows XP Professional.
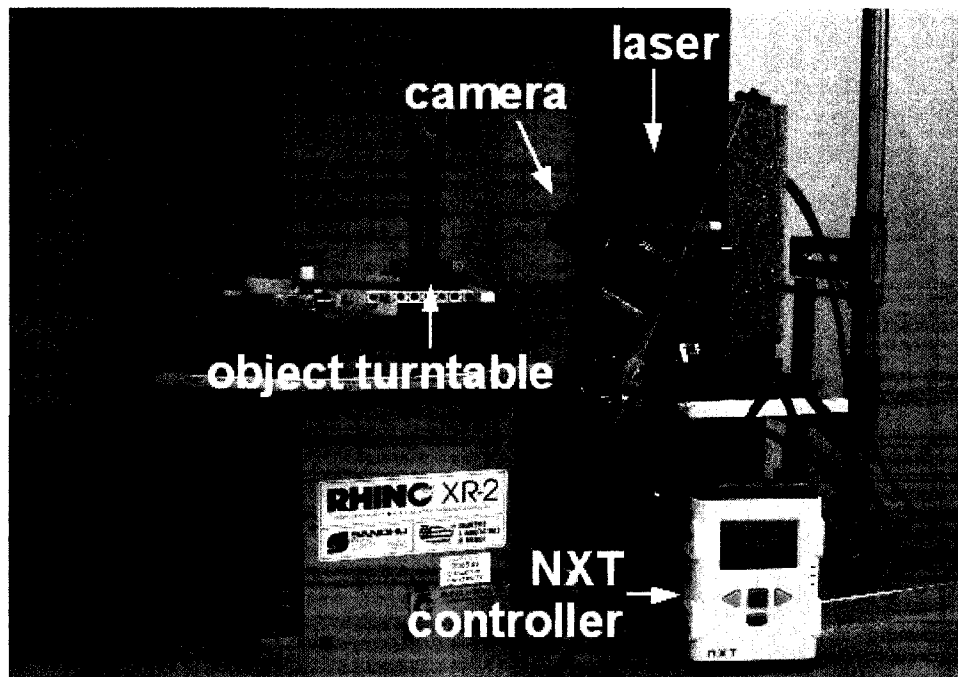
Figure 3.5: System setup.

3.2.2 Image acquisition and data processing steps

The reader is again referred to Figure 3.2 which describes the acquisition and data processing steps for a laser light-sectioning scanning system. These are described in sequence below for the proposed scanning system.

Chosen calibration technique

A coplanar calibration process was chosen for this system, employing the projectivity concepts presented (and used) in [22], [23], and [46]; specifically, the idea of a calibration grid as presented in [46] was employed. *Linear* methods are used for the *implicit* estimation of camera parameters; this allows for a much simpler implementation since it is not necessary to compute a complete camera model in this case. Both an analytical and a least squares (LS) solution, using the Direct Linear Transformation (DLT) algorithm, were explored; the DLT LS method was chosen overall. As discussed above

in Section 3.1.2, since the possibility of obtaining mismatched control pairs is relatively high when points are selected with an automated process, this system requires that the control pairs be semi-automatically selected by the user, although they are still determined to sub-pixel accuracy. The software isolates grid points from within the calibration grid image and the user is required to select the desired points to be used in the computation of H. This avoids the need to prune outliers from the set of control points before the DLT LS algorithm is applied, again leading to a more simple implementation. Complete details on the calibration process implemented can be found in Chapter 4.

Chosen image acquisition technique

For image acquisition, $M = 8$ views of the laser trace are captured in the plane of the section, separated by rotations of $R = 45°$ about the vertical z-axis. An important feature for future development of the system would be the ability of the camera and laser system to rotate about an object with the required number of degrees of freedom making it flexible for numerous applications; particularly for biomedical applications where the apparatus would ideally rotate about a patient and not vice versa. However, the system to rotate the measurement head has not been developed as part of this thesis, necessitating manipulation of the object for the experiments in this thesis. Rather than rotate the measurement head, the equivalent problem was to rotate the object of interest. For the first prototype setup, this was accomplished manually and for the second prototype setup this was done using a turntable controlled by a servo motor with a rotation sensor for accurate control.

Chosen image processing techniques

A simple background eliminating technique is applied to the images to condition them for effective segmentation. Both a 'parallel line thinning' algorithm [50], [51] and a

centre of gravity approach [49] were explored for determining a discrete set of point coordinates for each laser trace; the centre of gravity method was chosen overall. The line thinning method applies a global threshold to the image, converting it to a binary black and white image before finding the point coordinates, whereas the centre of gravity approach applies a threshold while still retaining intensity values in order to calculate a weighted location for the point coordinate within the laser trace. Details about the implementation of the algorithms and results for each can be found in Section 5.2.3. Complete details of the entire image processing process are provided in Chapter 5.

Chosen data manipulation techniques

The set of point coordinates for each segment is mapped to the Cartesian plane using the transformation matrix obtained from the camera calibration procedure. The end regions of the data segments tend to be plagued with noise, excessively steep slopes, and frequent outliers. This is due to the poor quality of the light signal where it intersects the outermost edges of the object [46]. For this reason, the first and last 10% of data points are discarded for the remainder of the analysis.

The data are then registered into a common coordinate system. First, the $M = 8$ segments of each planar section are reassembled about a suitable longitudinal axis, which is determined using a calibration cylinder. Planar sections are then 'stacked' along this axis in the z-direction by simply introducing the appropriate increment level as the 3rd dimension (i.e., the z-coordinate). As suggested by [1], defining new estimates of actual surface points by averaging samples from overlapping scans is an approach to dealing with residual error in the data. A simple method of averaging overlapping segment data was implemented for this scanner.

For the final curve fitting step of the data manipulation stage, Elliptical Fourier Descriptors (EFDs) [59] are used to fit a smooth curve through the segments of each

planar section. For the first prototype setup, the individual data segments were fit with polynomials before aligning them about the axis of rotation in order to attenuate noise in the data. This step was later removed since curve fitting with EFDs performed the same task. We assume that for biomedical applications, surfaces being scanned will be smooth and the use of EFDs is thus appropriate. The scanning of non-smooth surfaces is beyond the scope of this thesis. The validation and implementation of EFDs is explained in more detail in Chapter 6 along with details about the other data manipulation techniques implemented.

The result of the data manipulation stage is a point cloud of data representing planar section outlines of the object. This point cloud can be fit with a surface to produce a final 3D model.

Chosen object reconstruction techniques

The data are first exported from Matlab into a Wavefront Object file (*.obj*), a simple geometry definition file format, in order to facilitate the object reconstruction process. This format was chosen due to the fact that it supports data which is strictly 3D point cloud data. Texture, surface normals, and faces do not necessarily have to be defined in the data file (although they can be). It was also chosen due to its universal acceptability; compatibility with existing software was especially important.

Meshlab [75] (Pisa, Italy), an "open source, portable, and extensible system for the processing and editing of unstructured 3D triangular meshes", was chosen as the surface reconstruction software for this scanner. The Meshlab software satisfies all the desired criteria for the object reconstruction software: it supports the *.obj* file format; it is open source and available for Windows, Linux and MacOSX, and therefore highly accessible and free to acquire, and finally; it is capable of exporting the final 3D model into many different 3D model file formats making the final 3D model versatile and open to many different applications. For converting point clouds into triangulated 3D

surface reconstructions, Meshlab implements the efficient ball-pivoting algorithm [69] mentioned above.

It should be noted that object reconstruction was not the main focus of this thesis. Consequently, the choice of using Meshlab may be sub-optimal, but was made for completeness with convenience and other criteria in mind. More details on the *.obj* file format, Meshlab, and the ball-pivoting algorithm will be presented in Chapter 7.

Chapter 4

Calibration

This chapter describes the calibration process for the proposed 3D laser light-sectioning scanner system. First, a recap of the calibration problem described in Sections 3.1.2 and 3.2.2 is given in Section 4.1. Then, in Section 4.2 an explanation of how 2D projectivity theory can be used for camera calibration in the case of laser-light sectioning is presented, followed by descriptions of the two projective calibration methods investigated for this system. Finally, the methods, results, discussion, and conclusions for an experiment which evaluated these two methods under noisy input data conditions are presented in Sections 4.3, 4.4, 4.5, and 4.6 respectively, justifying the choice of using the least squares Direct Linear Transform method as indicated in Section 3.2.2.

4.1   Introduction

As described in the previous chapter, the result of projecting a plane of laser light onto an object is an illuminated stripe where the plane intersects the object's surface. This stripe is captured in a 2D image by the camera. Since the camera used to capture images is located at a fixed angle α to the laser plane, the basic goal of calibration for a laser light-sectioning system is to eliminate the linear portion of the

resulting camera perspective distortion. In other words, we must find a mapping between the illuminated pixel coordinates of the laser trace from the image to the world coordinates of the corresponding object surface points in the xy-plane. Figure 4.1 shows the image captured by the camera if a grid were to be superimposed on the plane of the laser; the highlighted quadrilateral demonstrates the camera perspective distortion. Figure 4.2 shows how the mapping from image coordinates to real world coordinates would remove this distortion.
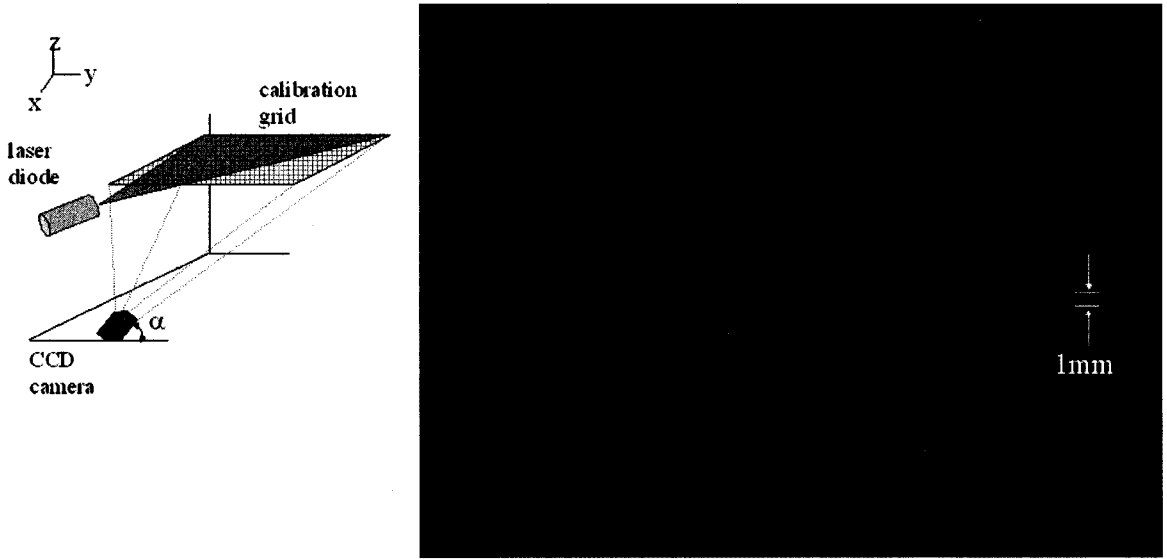
Figure 4.1: Diagram illustrating perspective distortion.

Mathematically, this transformation is represented by Equation 4.1

$$\mathbf{p}'_i = \mathbf{H}\mathbf{p}_i, \tag{4.1}$$

where $\mathbf{H}$ maps a point $\mathbf{p}_i = (x_i, y_i, w_i)^{\mathrm{T}}$ in the image to point $\mathbf{p}'_i = (x'_i, y'_i, w'_i)^{\mathrm{T}}$ in real world coordinates, effectively straightening the lines in Figure 4.2. Note the distinction between the homogeneous coordinates of a point $\mathbf{p}_i = (x_i, y_i, w_i)^{\mathrm{T}}$ and the inhomogeneous coordinates $(\frac{x_i}{w_i}, \frac{y_i}{w_i})^{\mathrm{T}}$; w_i in the homogeneous case represents

a scale factor and not the z-coordinate, and thus in the inhomogeneous case the coordinates represent ratios of this factor (see Appendix B for more details).
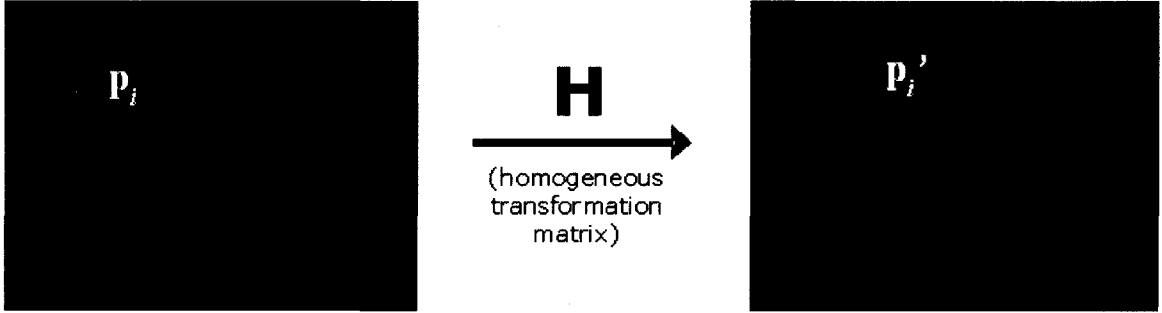
Figure 4.2: Diagram illustrating mapping between image and real world coordinates.

Recall from Section 3.1.2 that [22] describes how the camera calibration problem for a laser light-sectioning scanner can be simplified by applying the fundamental theorem of 2D *projectivity*. A *projectivity* is defined as an invertible mapping h from points in $\mathbb{P}^2$ (the projective space of homogeneous vectors) to points in $\mathbb{P}^2$ that maps lines to lines [45]. In other words, three points $\mathbf{x}_1$, $\mathbf{x}_2$, and $\mathbf{x}_3$ lie on the same line if and only if $h(\mathbf{x}_1)$, $h(\mathbf{x}_2)$, and $h(\mathbf{x}_3)$ do. To give the above definition an algebraic context we can say that h is a projectivity if and only if there exists a non-singular 3×3 matrix H such that for any point in $\mathbb{P}^2$ represented by a vector $\mathbf{x}$ it is true that $h(\mathbf{x}) = H\mathbf{x}$. Essentially, any projectivity can be expressed as a linear transformation in homogeneous coordinates, and conversely, any such mapping is a projectivity [45]. Therefore, in this context, the transformation H is a 3×3 *projective transformation* matrix (Equation 4.2) whose elements can be determined using linear algebra. It should also be noted that H is an *homogeneous matrix*, which means that multiplying the matrix by an arbitrary non-zero scale factor will not alter the projective transformation. Consequently there are eight independent ratios amongst the nine elements of H, hence the transformation H is specified by eight parameters [45]. For a more complete background on projective geometry,

homogeneous coordinates, and projective transformations, the reader is directed to Appendix B.

$$H = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \tag{4.2}$$

Recall also from the calibration overview given in Section 3.2.2 that the methodology chosen for finding this projective transformation H is a linear, implicit approach: *linear* meaning lens imperfections including radial distortion are not included in the model and thus linear equations are used; and *implicit* meaning the physical parameters of the camera are not explicitly computed.

The following section explains in more detail how projective geometry can be used to solve the calibration problem and presents two possible linear implicit methods for determining the required projective transformation H.

4.2 Camera Calibration Using Projective Geometry

The fundamental theorem of 2D projectivity is our link between 2D projective geometry and the camera calibration problem. The theorem is stated below.

The Fundamental Theorem of 2D Projectivity: *Given four distinct non-collinear points on a plane and another four distinct non-collinear points on the other plane, there is one and only one projectivity which carries the first four points respectively into the second four points.* [22]

What this means for the calibration problem of a laser light-sectioning scanner is that if we treat the light stripe plane as a 2D plane and the camera image plane as a 2D plane then each image point can uniquely determine the world coordinates of the corresponding object point [22]. In a more practical sense, this means that since we know all of the recorded laser traces will lie on this laser plane, the laser trace images from the scanner system can be calibrated by first superimposing a calibration grid of known dimensions on this laser plane (as in Figure 4.1), recording an image, and then computing an appropriate homogeneous transformation matrix H. H is computed by taking points from this grid image at certain locations and mapping them to their corresponding points on the known calibration grid in Cartesian space. The point selection process itself is semi-automatic; the software isolates grid points from within the calibration grid image and the user is required to select the desired points to be used in the computation of H.

Two methods have been investigated for finding H: 1) a purely analytical technique, referred to as the Analytical method and 2) a least squares (LS) method using the Direct Linear Transformation algorithm (DLT), referred to as the DLT LS method.

4.2.1 Analytical method

Beginning from a general transformation equation $\mathbf{p}'_i = \mathbf{H}\mathbf{p}_i$, where $\mathbf{p}'_i = (x'_i, y'_i, w'_i)^{\mathrm{T}}$ and $\mathbf{p}_i = (x_i, y_i, w_i)^{\mathrm{T}}$ are 2D homogeneous projective point coordinates, we can rewrite the equation so that the point coordinates are written in homogeneous Cartesian form; that is, their 3rd coordinate is 1 and an independent homogeneous scale factor ρ_i is factored out. This leaves:

$$\rho_i \mathbf{q}'_i = \mathbf{H}\mathbf{q}_i \qquad (4.3)$$

For the specific case of laser light-sectioning calibration, the variables of Equation 4.3 represent the following:

$\mathbf{q}_i' = (x_i', y_i', 1)^{\mathrm{T}}$ are the homogeneous coordinates of a point in real world coordinates on the non-distorted calibration grid,

$\mathbf{q}_i = (x_i, y_i, 1)^{\mathrm{T}}$ are the coordinates of the same point on the distorted calibration grid image, and

ρ_i represents a scaling factor which, when left factored out, leaves the points $\mathbf{q}_i'$ and $\mathbf{q}_i$ as homogeneous Cartesian coordinates.

Equation 4.3 is essentially a set of three linear equations derived from the ith point correspondence. Therefore, four such corresponding point pairs would result in 12 similar equations with 12 unknowns; the eight independent parameters of H and the four independent homogenous scale factors $\rho_i, i = 1, ...4$. Recall from Section 4.1 that H is an homogeneous matrix and can only be determined up to a non-zero scale factor. Consequently there are eight independent ratios amongst the nine elements of H [45].

These 12 equations in 12 unknowns can be determined using standard techniques for solving linear equations (in our case, Maple and the Matlab Symbolic Toolbox software were used). Although solutions for all the unknowns would be obtained, only the elements of H are required to transform all other image points from the projective onto the Cartesian plane. The transformation matrix H may now be used to transform all points in the projectively distorted camera view into the corresponding points on the Cartesian plane. Because the linear projective distortion has been removed, a metric has been reestablished in the image and the Euclidean distance between point pairs may be measured. The quality of the metric (i.e., the precision of the Euclidean distance measurements) is sensitive to the four points chosen to determine the matrix H. For a more complete derivation of the analytical method for finding a general

projective transformation see Appendix B.2.1.

4.2.2 Direct Linear Transformation Least Squares method

The DLT algorithm is a linear algorithm for determining H given a set of *at least* four 2D to 2D point correspondences, $\mathbf{p}_i \leftrightarrow \mathbf{p}_i'$. Again, it is derived from the general transformation given by the equation $\mathbf{p}_i' = H\mathbf{p}_i$. Using vector cross products and isolating the elements of the H matrix into a column vector results in the following:

$$\begin{bmatrix} 0 & 0 & 0 & -w_i'(x_i, y_i, w_i) & y_i'(x_i, y_i, w_i) \\ w_i'(x_i, y_i, w_i) & 0 & 0 & 0 & -x_i'(x_i, y_i, w_i) \end{bmatrix} \mathbf{h} = \mathbf{0} \qquad (4.4)$$

where $\mathbf{h} = (h_1, h_2, ..., h_9)^{\mathrm{T}}$ are the 9 elements of the matrix H. The 2×9 matrix of Equation 4.4 is denoted as A_i, and the whole expression is referred to as the DLT matrix equations for the ith point correspondence. For a more complete derivation of the DLT matrix equations, see Appendix B.2.2.

For the specific case of laser light-sectioning calibration, an inhomogeneous case of the DLT algorithm is used. This means, rather than solve for $\mathbf{h}$ directly as a homogeneous vector, we can impose that one of the 9 elements of H (say, H_{33} and thus h_9) be 1, leaving eight independent ratios amongst the nine elements of H to be found (see Appendix B.2.2 for more details). The result is the inhomogeneous set of linear equations, given in Equation 4.5:

$$\begin{bmatrix} 0 & 0 & 0 & x_i w_i' & y_i w_i' & w_i w_i' & -x_i y_i' & -y_i y_i' \\ x_i w_i' & y_i w_i' & w_i w_i' & 0 & 0 & 0 & -x_i x_i' & -y_i x_i' \end{bmatrix} \tilde{\mathbf{h}} = \begin{pmatrix} w_i y_i' \\ w_i x_i' \end{pmatrix} \qquad (4.5)$$

where $\tilde{\mathbf{h}}^{\mathrm{T}}$ is a 1×8 inhomogeneous vector consisting of the first eight components of $\mathbf{h}$. Furthermore, since the DLT equations hold for any homogeneous representation

(x_i', y_i', w_i') of the point $\mathbf{p}_i'$ a further simplification in the laser light-sectioning calibration case is to set $w_i' = 1$ and $w_i = 1$, meaning (x_i', y_i') and (x_i, y_i) are the 2D world coordinates and the 2D image coordinates, respectively. Thus, Equation 4.5 simplifies to:

$$\begin{bmatrix} 0 & 0 & 0 & x_i(1) & y_i(1) & w_i(1) & -x_i y_i' & -y_i y_i' \\ x_i(1) & y_i(1) & w_i(1) & 0 & 0 & 0 & -x_i x_i' & -y_i x_i' \end{bmatrix} \tilde{\mathbf{h}} = \begin{pmatrix} (1)y_i' \\ (1)x_i' \end{pmatrix} \quad (4.6)$$

Since each set of control point pairs gives two such equations, $2m$ equations are obtained from m calibration point pairs. Concatenating the equations from $m \geq 4$ correspondences then generates a matrix equation of the form:

$$\mathbf{A}_{2m \times 8} \tilde{\mathbf{h}}_{8 \times 1} = \mathbf{b}_{2m \times 8}, \quad (4.7)$$

where $\mathbf{A}$ now has eight columns, $\tilde{\mathbf{h}}$ is the column vector of H matrix parameters and $\mathbf{b}$ is a 1×8 array of known calibration coordinates. If $m > 4$, such an over-determined system can be solved for $\tilde{\mathbf{h}}$ with the least squares methods described in Appendix B (Section B.2.2).

It was hypothesized that this method would produce more accurate transformation mappings when input data are noisy since more input control point pairs are used to find the best (in a linear least-squares sense) transformation mapping between pairs. The analysis in the subsequent section validates this hypothesis and presents other interesting outcomes regarding the influence of calibration method, noise level, and point pair location on transformation accuracy.

4.3 Methods

This section presents the methods for an analysis of the two different techniques, presented above, for finding a homogeneous projective transformation matrix H, which can be used to remove the linear perspective distortion from laser-light sectioning image data: 1) the Analytical method; and 2) the DLT LS method. Both were evaluated in terms of their performance under noisy input data conditions; that is, noise associated with the points taken from the camera image of the calibration grid which are used to compute H. The study looks at how accurately the calculated transformation matrix maps points from the image of the calibration grid to their ideal location on the grid in a Cartesian coordinate system. Noise in the input data could arise from issues such as precision of the calibration grid or artifacts in the image processing methods used to isolate grid corner locations and identify their coordinates within the image. We consider only the case where noise is added to the coordinates of the calibration image due to the fact that point locations for the corresponding real world pairs are assigned with certainty and not estimated from an image.

A test scenario was generated using a transformation matrix H known to be representative of the mapping required to transform points for the laser light-sectioning system presented in this thesis; that is, the transformation was determined from a calibration grid image taken using the first prototype setup [16]. Using the inverse of H to transform the four corners of a known square on the calibration grid, the corresponding point pair locations, as they would appear on the distorted image of the grid, were generated (P1-P4 on Figure 4.3). This produced a complete set of ideal control point pairs and the transformation that maps between the two. In addition to these four point pairs, 16 other point pairs were generated for use with the DLT LS method ($\times$'s on Figure 4.3).

The noise associated with identifying grid intersections would be random in nature
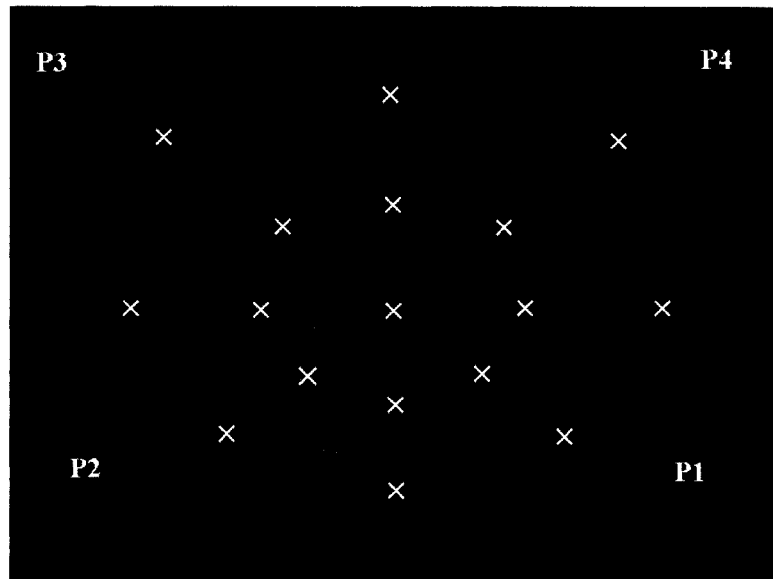
Figure 4.3: Diagram illustrating point selection for calibration noise analysis.

and is not expected to exceed the width of a grid line, which is approximately 3 pixel units in our system (at the centre of the calibration grid image). Assuming a Gaussian distribution with 95% chance of choosing points within 3 pixels, the variance is $\sigma^2 = 2.25$ (i.e., 2 standard deviations σ is 3 pixels). Random Gaussian noise with zero-mean and noise variances above and below 2.25 (varied by a factor of 2; that is $\sigma^2 = 0.140625, 0.28125, 0.5625, 1.125, 2.25, 4.5, 9$, and 18) were added to the four corner points of the distorted quadrilateral and the extra 16 points for the DLT LS method, in both the x and y directions. Noisy points were then used as input to the two calibration algorithms. The DLT LS method was used twice; once with four control point pairs and once with 20 control point pairs.

In total, 20 trials were performed at each of the eight noise variance levels for each calibration algorithm. The radial difference between the transformed points and their known locations according to the ideal mapping was averaged across all 20 trials at each σ^2 value for the four corner grid points (P1-P4). This average radial error was

then scaled according to the scaling used for the test object imaged in [16] in order to present the results in units of mm.

Using SPSS (Chicago, Illinois), a statistical analysis software package, a three-way analysis of variance (ANOVA) test was conducted on the results to determine whether or not calibration method, point location, and/or noise level have a significant effect on the radial error of transformed points. Pairwise comparisons between factors were performed using the Tukey and Dunnett's T3 post-hoc tests to determine which calibration methods, point locations, and/or noise levels produce mean radial errors significantly different from the others.

4.4 Results

Figures 4.4, 4.5, and 4.6 show error results at eight different magnitudes of noise for the analytical method, the DLT LS method using four control point pairs, and the DLT LS method using 20 control point pairs, respectively. Errors for the four test corner points are shown as separate data sets. Average error across the 20 trials is plotted and standard deviations for the 20 trials are indicated with error bars.

Table 4.1 summarizes the p-values for the ANOVA analysis. Statistically, the ANOVA test revealed main effects for all factors ($\alpha = 0.05$): noise variance level ($p < 0.001$), point location ($p = 0.038$), and method ($p < 0.001$). There was also an interaction found between method and noise variance level ($p < 0.001$). An interaction reflects that the effect of the method on the error results depends on the values of noise variance level, and vice versa.

Although the effect of point location was found to be significant, post-hoc tests were inconsistent; the Dunnett's T3 test showed that no point location had an error significantly different from the others. Post-hoc tests also revealed that there is in fact no significant difference between the analytical method and the DLT LS method
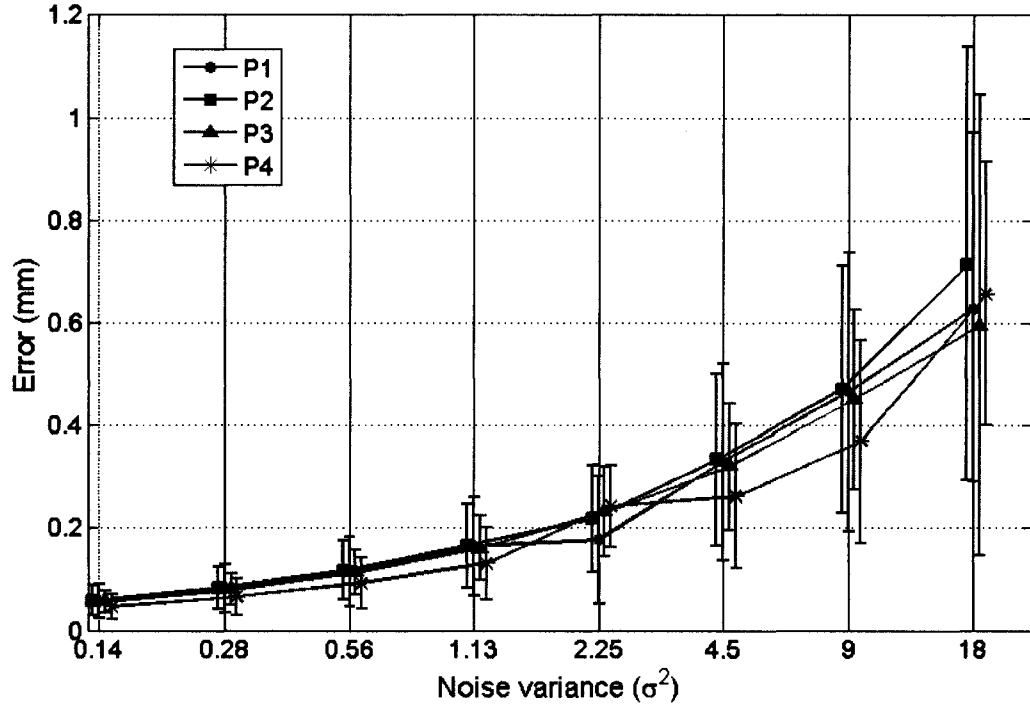
Figure 4.4: Error results for the analytical method using four point correspondences.
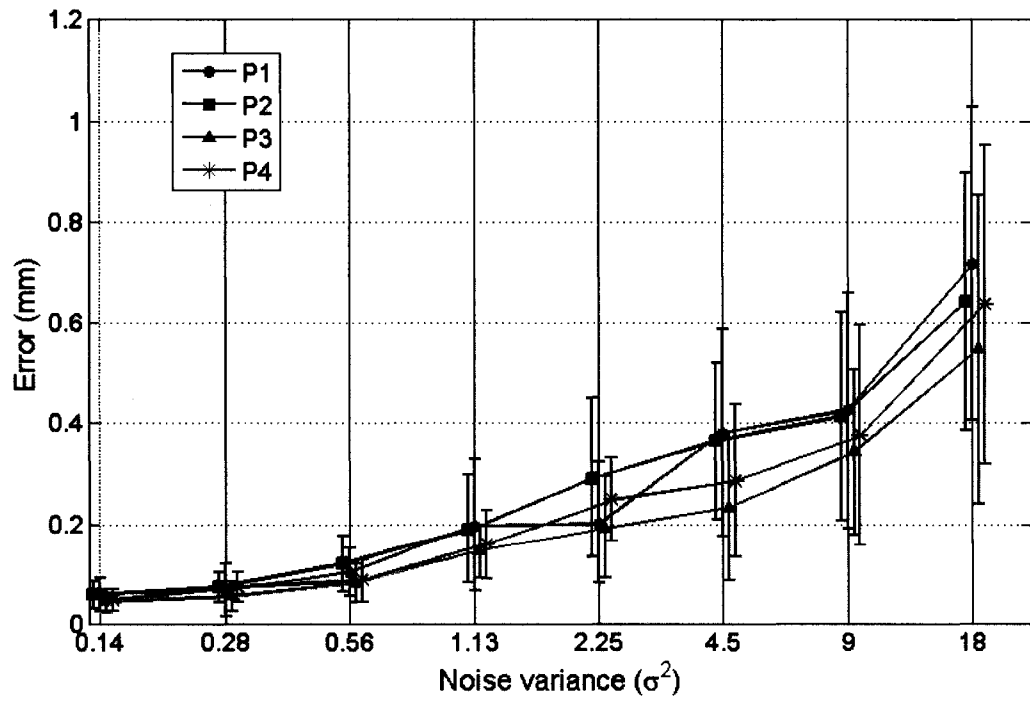
Figure 4.5: Error results for the DLT LS method using four point correspondences.
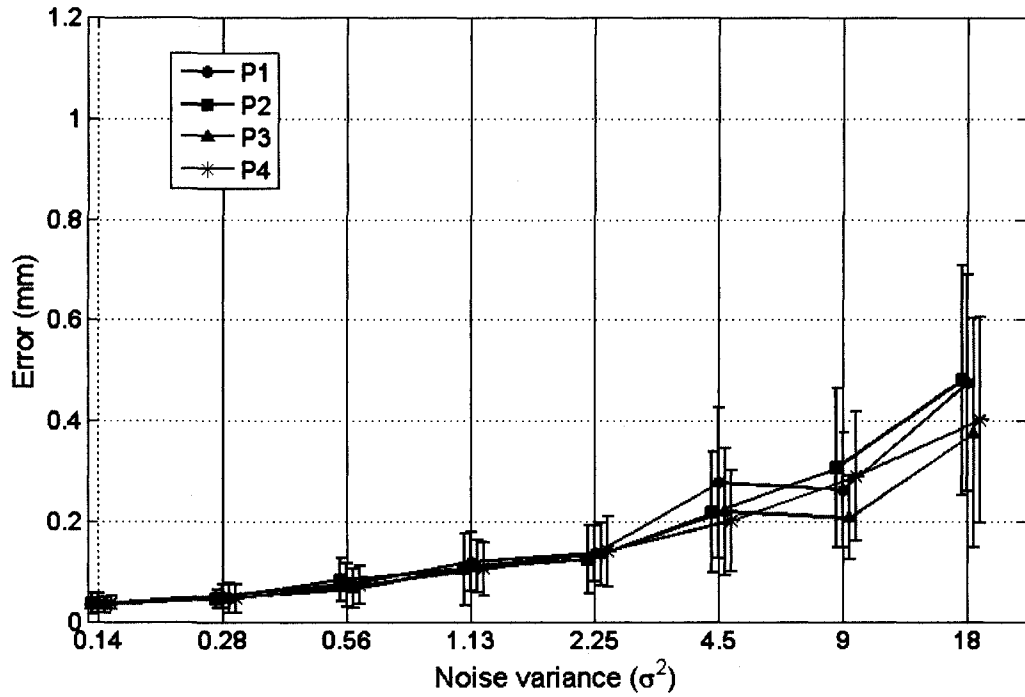
Figure 4.6: Error results for the DLT LS method using 20 point correspondences.

Table 4.1: Univariate Analysis of Variance (ANVOA) Results.

Source	Significance
Method	< 0.001
Point	0.038
Noise level	< 0.001
Method and Point	0.954
Methods and Noise level	< 0.001
Point and Noise level	0.994
Method and Point and Noise level	0.984
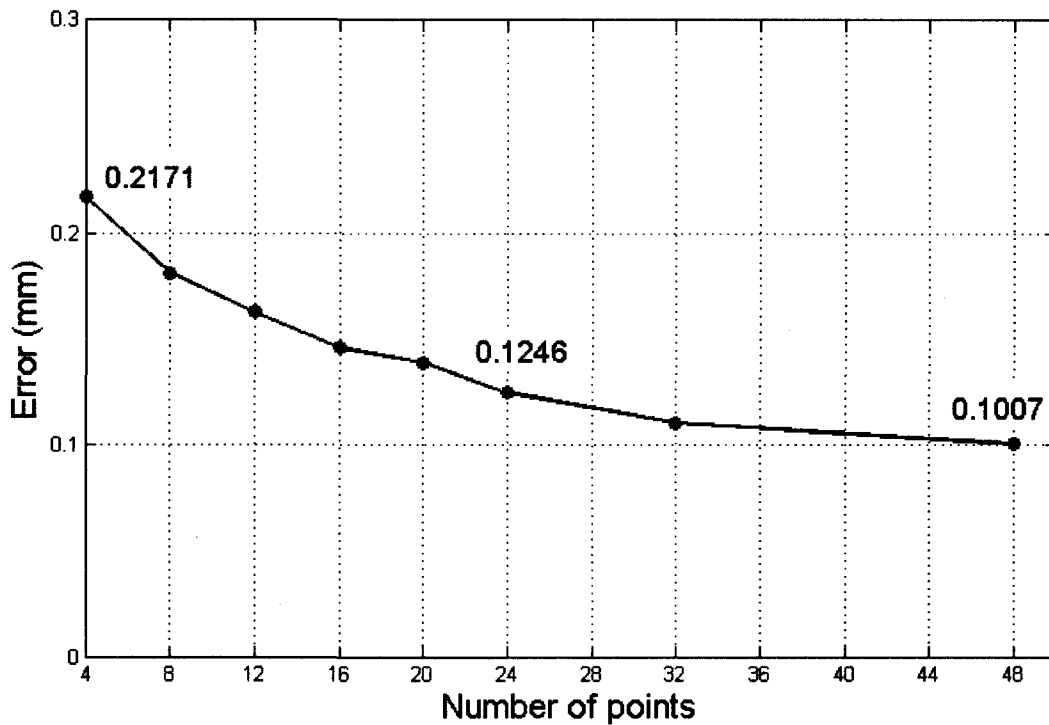
Figure 4.7: Average error vs. number of points used with the DLT LS method (for $\sigma^2 = 2.25$)

using four control point pairs ($p = 0.991$). The DLT LS method using 20 control point pairs, however, is significantly better than both of the previous two methods ($p < 0.001$ in both cases).

Figure 4.7 shows a plot of average error vs. number of points used in the DLT LS method at a noise level of $\sigma^2 = 2.25$. In this figure, error for each of the four tests points has been averaged and is therefore not shown separately as was done previously. Clearly, as the number of points used with the DLT LS method increases, average error decreases. The difference in average error obtained when using 24 points compared to four points is 0.0925 mm whereas the difference in average error obtained when using 48 points compared to 24 is only 0.0239 mm; the improvement in error in the later case is almost four times less compared to the first.

Using Matlab's Symbolic Toolbox, computational speed for the analytical method,

was considerably longer than that of the DLT LS method. Computation of calibration matrices used in the analytical method was approximately 7.5 mins per matrix computation (using a 3GHz Pentium 4 with 1GB RAM). For the DLT LS method, computation of calibration matrices was about 0.002 s per matrix computation, regardless of how many control point pairs were used.

4.5 Discussion

Figures 4.4, 4.5, and 4.6 demonstrate how average radial error increases with increasing noise. The significant main effect of noise variance (see Table 4.1) comes as no surprise; as noise in the input data increases it is expected that output error will also increase. A point's location within the image appears to have no substantial effect on the amount of error associated with its transformation into Cartesian space; at least not for the amounts of noise studied for this analysis. Differences in point spacing due to the distortion could explain the main effect found by the ANOVA test, but overall, looking at Figures 4.4, 4.5, and 4.6 the effect does not seem to be an important factor. Finally, Figures 4.4 and 4.5 show comparable errors for all four transformed test points when the analytical method and the DLT LS method using four control point pairs are used. The DLT LS method using 20 control point pairs (Figure 4.6) appears to be better than both of these two, which is supported by the statistical results (see Table 4.1).

As noise levels increase, the improvement of the DLT LS method with 20 points over the other two methods becomes greater, hence the interaction found between noise variance level and method in the post-hoc tests. The fact that the DLT LS method using 20 points produces much more accurate point mappings indicates that the use of more control point pairs for finding H is beneficial. This is consistent with [43] which reported that the "pseudo-inverse is the best approximation in undistorted

models" [43]. Within the noise levels expected for the proposed laser light-sectioning system ($\sigma^2 \leq 2.25$), the DLT LS method using 20 or more control point pairs would yield the best results. The average error using this method for this noise range, as indicated in Figure 4.6, is less than 0.2 mm for all four points. This is within a reasonable tolerance for the desired accuracy of the system at the current stage of development. Figure 4.7 indicates that average error can be reduced even further when using more than 20 control point pairs however the improvement in error becomes less and less as more points are added. Since the identification of calibration points within the image for this system is semi-automatic (that is, the grid intersection locations are automatically isolated and then points are selected by the user), choosing more than 20 points would not add a substantial amount of time to the entire calibration process and may be worth the benefit gained in terms of error reduction.

These results are consistent with an evaluation in [45], whereby the DLT LS algorithm was evaluated against the Gold Standard for estimating 2D projectivities between two images. The Gold Standard in this case is an iterative method for minimizing a *geometric* cost function. More specifically, it minimizes not only the distance between transformed points $\hat{\mathbf{p}}'_i$ and known world points $\mathbf{p}'_i$, but also the distance between image points $\mathbf{p}_i$ and the subsidiary variables $\hat{\mathbf{p}}_i$, where $\hat{\mathbf{p}}'_i = \hat{\mathbf{H}}\hat{\mathbf{p}}_i$ and $\hat{\mathbf{H}}$ is the estimate of H, for $i = 1, ..., m$. This method "requires the minimization over all choices of points $\hat{\mathbf{p}}_i$ as well as the entries of the transformation matrix H" [45], resulting in a total of $2m + 9$ parameters compared to the eight parameters of the DLT LS method. Results were compared to the theoretical minimum (or residual error). They conclude that for noise up to ± 5 pixels, the DLT algorithm performs adequately, closely matching the results for the Gold Standard. In fact, for less severe projectivities, (as would be the case for this system), the DLT performs almost as well as the Gold Standard. It fails for noise levels of ± 10 pixels or higher, but this is considered to be extreme and not an expected scenario for this scanning system. Our results are also

consistent with those presented for a different linear algorithm (Hall) in [43], where accuracy of 3D coordinate measurement was 0.1615mm on average using a predefined set of control points available at http://www.cs.cmu.edu/ rgw/TsaiCode.html.

The processing time of the DLT method was considerably shorter than the analytical method. Clearly, the solution to the over-determined system of matrix equations in Equation 4.6 converges very quickly. It should be noted that the calibration matrix is only computed once for a given camera setup, and used thereafter for transforming all data points lying along the laser traces captured by the camera. For this system, the difference in computation time between these two methods is therefore not necessarily an important factor when comparing the two approaches.

4.6   Summary

The DLT LS method using 20 points produces much more accurate point mappings indicating the use of more control point pairs for finding H is beneficial when input data are noisy. Based on this analysis, at least 20 point pairs should be used for calibration, when the calibration algorithm successfully isolates this many points in the image.

Chapter 5

Image Acquisition and Image Processing

This chapter describes the image acquisition and image processing steps for the proposed 3D laser light-sectioning scanner system. The relevant portion of the system flow diagram from Section 3.1.2 is repeated in Figure 5.1 for reference. The goals of image acquisition and image processing are: to capture all the images required to create a complete 3D model (Section 5.1); and to perform the image processing necessary to isolate the trace of laser light within each image and determine a set of discrete point coordinates to represent the best path through the trace (Section 5.2). Two different approaches were explored for the final image processing step of identifying point coordinates. The first was a line thinning algorithm and the second was a centre of gravity approach. Each method requires that a different thresholding process take place before it is employed. Both of these thresholding methods are described in Section 5.2.2 followed by descriptions of their respective point coordinate identification algorithms in Section 5.2.3.

5.1 Image Acquisition

For image acquisition, $M = 8$ views of the laser trace are captured by the CCD camera, separated by rotations of $R = 45°$ about the vertical z-axis. The system to rotate
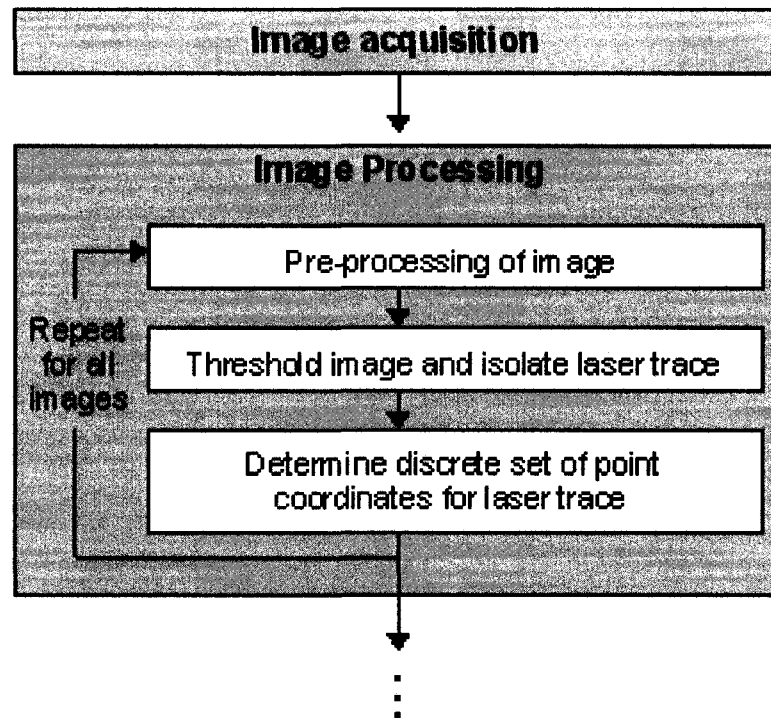
Figure 5.1: Diagram of Chapter 5 organization from system flow diagram.

the measurement head has not been developed as part of this thesis; consequently the object itself is rotated as this is the equivalent problem to rotating the measurement head. This is accomplished by placing the object on a turntable controlled by a servo motor with a rotation sensor for accurate control. After images for a complete planar section have been captured, the platform holding the camera and laser is raised, also using a servo motor, in increments of 2 mm.

A Labview interface, shown in Figure 5.2, was developed to control both the image acquisition device for the camera and the servo motors of the scanner apparatus. Visual feedback of the images being taken is provided as well as numerical indicators displaying the current section and rotation numbers of the scan procedure.
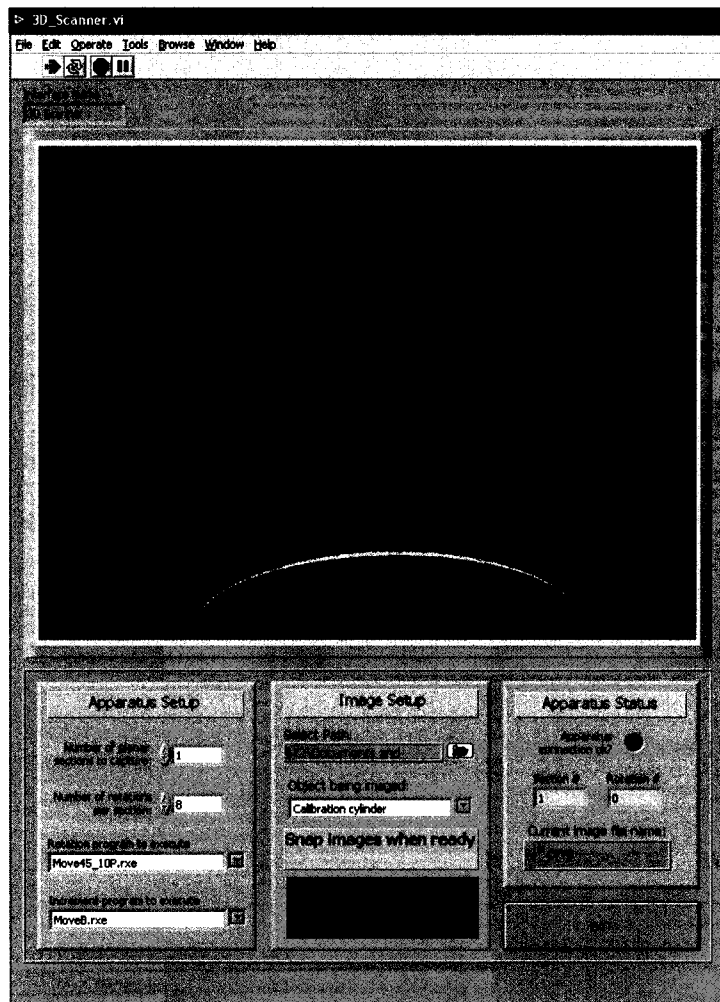
Figure 5.2: Labview interface for controlling the laser light-sectioning scanner apparatus.

5.2 Image Processing

After the images have been acquired and saved, all image processing occurs offline using Matlab and its Image Processing Toolbox. Images obtained from the CCD camera are Portable Network Graphics (.png) files, imported into Matlab as $480 \times 640 \times 3$ RGB images.

5.2.1 Image pre-processing

A background eliminating algorithm is applied to the images to help isolate the laser line traces. If lighting is ideal during data capture (i.e., ambient light is minimal, or at least uniform) this would not be a required step, however, assuming that such a scanner would potentially be used in normal lighting conditions, this step of the process was developed to eliminate non-uniform ambient lighting effects in the images.

The algorithm involves a morphological 'opening' operation, commonly used to "remove small objects from an image while preserving the shape and size of larger objects in the image" [78]. Morphological opening is defined as an *erosion* followed by a *dilation*, two fundamental morphological operations which are often used in combination to implement different image processing operations. "Dilation adds pixels to the boundaries of objects in an image, while erosion removes pixels on object boundaries. The number of pixels added or removed from the objects in an image depends on the size and shape of the structuring element used to process the image" [78]. The structuring element, a matrix consisting of only 0's and 1's, can have any arbitrary shape and size. To perform a morphological operation the center of the structuring element is positioned over the pixel of interest in the input image; the elements with values of 1 define the neighborhood of the pixel of interest. This neighborhood is used to determine the value of the pixels in the output image, based on the specified rules of the morphological operation. For erosion, the value of the output pixel is the minimum value of all the pixels in the input pixel's neighborhood, and for dilation, the value of the output pixel is the maximum value of all the pixels in the input pixel's neighborhood. A rectangular structuring element is used here for both the erosion and dilation operations (see [78] for more information).

Figure 5.3 shows an extreme case of non-uniform lighting effects being reduced by this process. Notice that the artefact (a shadow) in the bottom left corner of Figure

5.3 (a) is less apparent in Figure 5.3 (b), which now depicts only the object itself.

Figure 5.3: Images demonstrating background elimination, (a) before and (b) after the algorithm is applied.

The next preprocessing step is to convert the RGB images into grayscale images, whose pixel values now specify intensities from 0 to 255. This not only reduces the size of the images by one dimension (i.e. they become 480×640), but it makes further processing simpler because now we are only dealing with intensity values.

5.2.2 Image thresholding

As mentioned above, two thresholding processes have been implemented to correspond with the two different algorithms for determining point coordinates. The thresholding process for the line thinning method involves converting the grayscale images to black and white binary images. The threshold level required for this conversion is chosen by a function in Matlab which computes a global threshold level based on Otsu's method [79]. The threshold is applied to each image such that pixels with an intensity value below the threshold are set to 0 (black) and those above it are set to 1 (white).

For the centre of gravity (COG) method, thresholding occurs at the time of calculating the COG rather than universally for the entire image before the algorithm is applied. The threshold value is also chosen using Otsu's method [79]. Pixels below the threshold are not considered in the COG algorithm and those above it are. The difference being that in this case, the actual intensity values are retained because they

influence the location chosen for the point coordinate, as will be explained further in the next section.

5.2.3   Identification of point coordinates to represent the laser trace

Two different approaches were explored for finding a discrete set of point coordinates for the laser trace segments, each requiring the different methods of thresholding described in Section 5.2.2.

The first is an iterative thinning algorithm available in Matlab [50], which operates on the black and white images from the first thresholding method. The algorithm considers the white laser trace area (represented by 1s) as a blob and thins the blob to 8-connected skeletons, retaining diagonal lines and 2×2 squares according to [51]. The results using this method were acceptable for the preliminary experiments published in [16], [17], and [18] however, since this approach did not determine point coordinates with sub-pixel precision, quantization error added to the overall noise in the data. This is because although the blob is thinned until it becomes a line which is only 1 pixel thick, the choice of that one pixel is biased in those cases where the trace is an even number of pixels thick. This is demonstrated in Figure 5.4, which shows a zoomed in portion of a laser trace from a test cylindrical object. The dark gray pixels represent the blob after thresholding, the light gray pixels represent the line (1 pixel thick) which remains after the thinning algorithm has been applied, and the connected dots represent the corresponding point coordinates chosen to best represent the laser trace. In the areas which are two pixels thick, the algorithm chooses one pixel to represent the centre of the trace (usually the uppermost pixel) rather than the exact centre between the two pixels. Consequently, the point coordinate chosen is also biased towards the top.
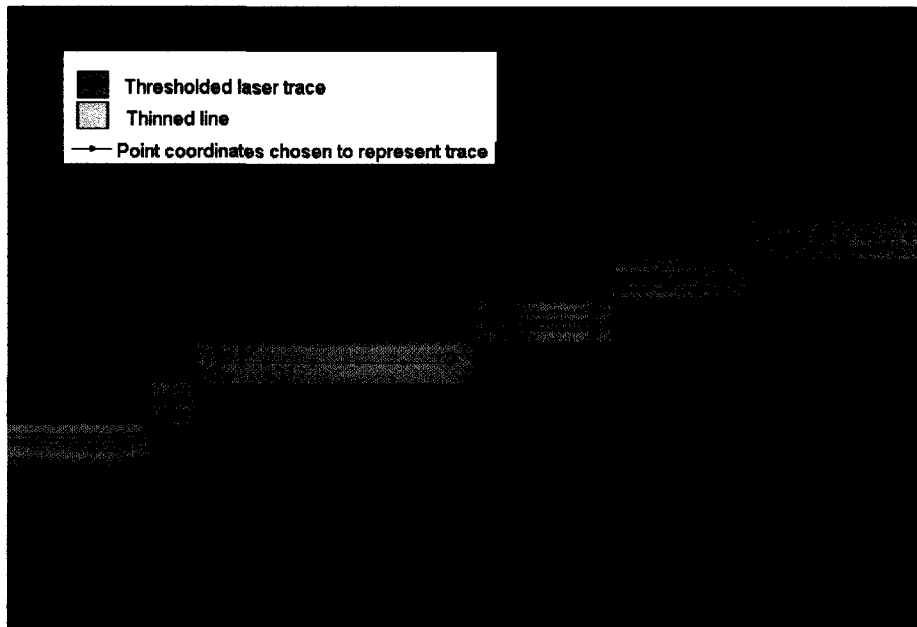
Figure 5.4: Image demonstrating point selection after line thinning algorithm has been applied.

The second approach, the COG algorithm, applies thresholding at the time the discrete point location is computed. Starting from the left side of the image, each column of the image is considered sequentially. The threshold value (see Section 5.2.2) is used to scan each column; the portion of the column with intensities greater than the threshold value represents the laser trace and is used in the calculation of the COG. A one-dimensional moment, y_c, is calculated for the sub-column of pixels exceeding this threshold according to Equation 5.1

$$y_c = \frac{\displaystyle\sum_{y=y_s}^{y_e} y \cdot I^p(x,y)}{\displaystyle\sum_{y=y_s}^{y_e} I^p(x,y)}, \tag{5.1}$$

where x, y are coordinates in the image, $I(x,y)$ is the intensity at (x,y), y_s and y_e

are the start and end indices for the column of pixels which exceed the threshold, and p is a power factor. The power factor can be used to influence the weighting of intensities; a value of p greater than 1 gives higher importance to high intensities, and p less than 1 will emphasize pixels with lower intensities [49]. A power factor of 1 was used here, meaning the moment calculated simply corresponds to the COG. The existence of the power factor allows for extra flexibility in the formula and could be changed if the intensity of the laser trace is not very high in an image due to poor lighting conditions, etc. The point coordinate chosen to represent the centre of the laser trace then becomes (x, y_c), where x is the current column number. This process is illustrated in Figure 5.5.

Figure 5.5: Illustration of centre of gravity algorithm.

It should be noted that the centre of gravity is only calculated in the y-direction. Due to the fact that the projected laser plane is horizontal (i.e., parallel to the xy-plane) the resulting reflection on the object surface is predominantly horizontal in

nature regardless of the surface topography of the object. Since the camera always views the intersection of this plane with the object'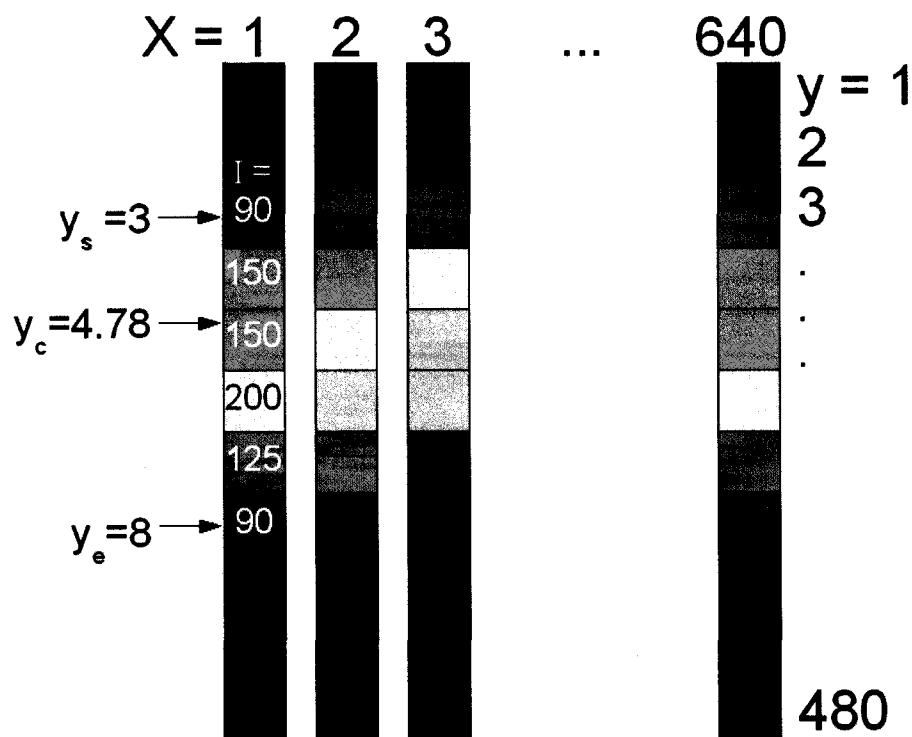s surface from a fixed angle, the resulting image of the laser trace will typically span the horizontal width of the image, with variations in the vertical direction representing changes in surface topography.

Figure 5.6 demonstrates the difference between the two approaches for defining a set of point coordinates. The same zoomed in portion of the laser trace depicted in Figure 5.4 is shown except this time, the original grayscale image is used rather than the thresholded version; that is, the pixels are shown with their true intensity values. The string of point coordinates that results after the line thinning algorithm is applied is again shown with connected dots. The string of point coordinates that results after the COG algorithm is applied is shown with squares connected by a dotted line. The line thinning algorithm tends to be biased towards the top of the trace, whereas the COG method produces point coordinates which match the centre of the trace much more closely. Note also that with this approach, since the chosen point locations do not necessarily have to lie on the centre of a particular pixel, the resulting set of point coordinates produces a smoother line without as many steep jumps.

In summary, the COG algorithm has been implemented in order to reduce noise due to quantization error produced by the first approach attempted, the line thinning method. Overall, it appears that the COG algorithm produces superior results for finding a set of point coordinates which best represents the laser trace within the images captured.

The overall result of the image acquisition and image processing phase is that the laser traces from each image are now represented by sets of (x, y) coordinates, as shown in Figure 5.7. By transitioning from images to point coordinates, the amount of data has been substantially reduced, facilitating further manipulation in order to create a final 3D model.
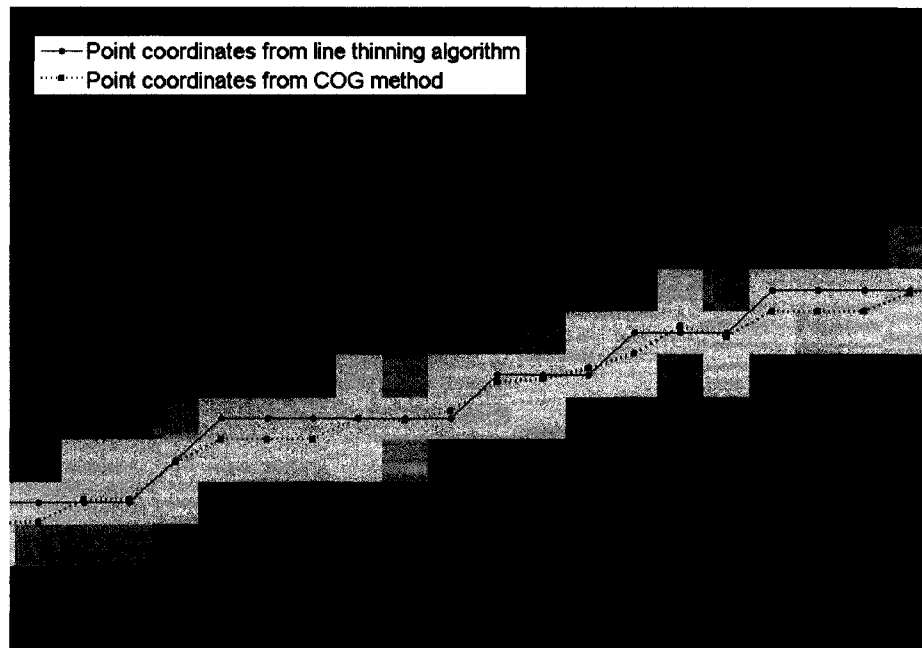
Figure 5.6: Comparison of thinning algorithm vs. centre of gravity algorithm. Thinning algorithm is biased towards the top of the trace.
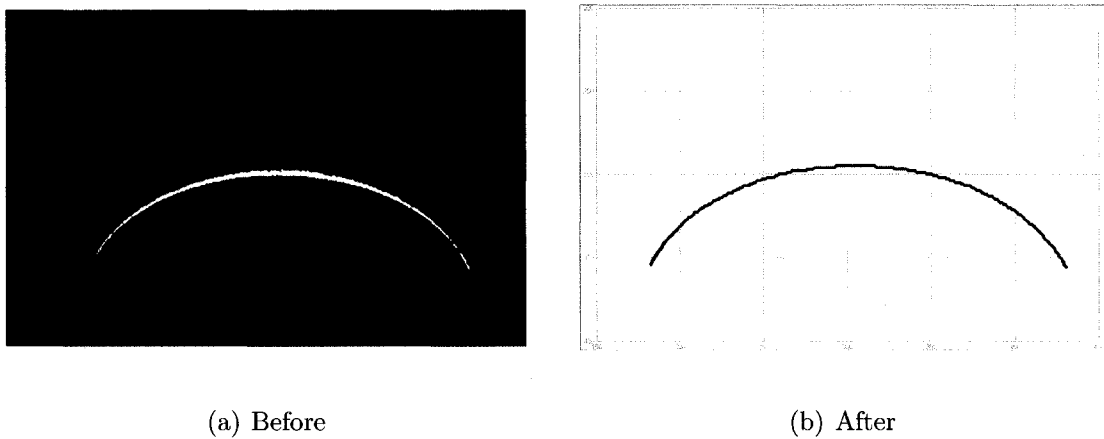
(a) Before

(b) After

Figure 5.7: Image of a laser trace (a) before and (b) after the centre of gravity algorithm is applied

Chapter 6

Data Manipulation

This chapter describes the data manipulation process for the proposed 3D laser light-sectioning scanner system. The relevant portion of the system flow diagram from Section 3.1.2 is repeated in Figure 6.1 for reference. The main goal of the data manipulation stage is to reconstruct the data segments of each planar section about a central axis so that a final 3D model can be created. Section 6.1 describes the transformation process which removes the linear perspective distortion from the segments. Section 6.2 presents the process for aligning the segments of data about an appropriate axis of rotation. The algorithm for averaging overlapping data segments is described in Section 6.3. Finally, Section 6.4 discusses the use of Elliptical Fourier Descriptors (EFDs) for reducing noise in the segments as well as for defining a single continuous curve through them. Contained in this section is an analysis performed on both simulated and real data demonstrating the noise reduction effects of EFDs and their ability to define smooth contours representing the planar section data from this 3D laser light-sectioning scanner.
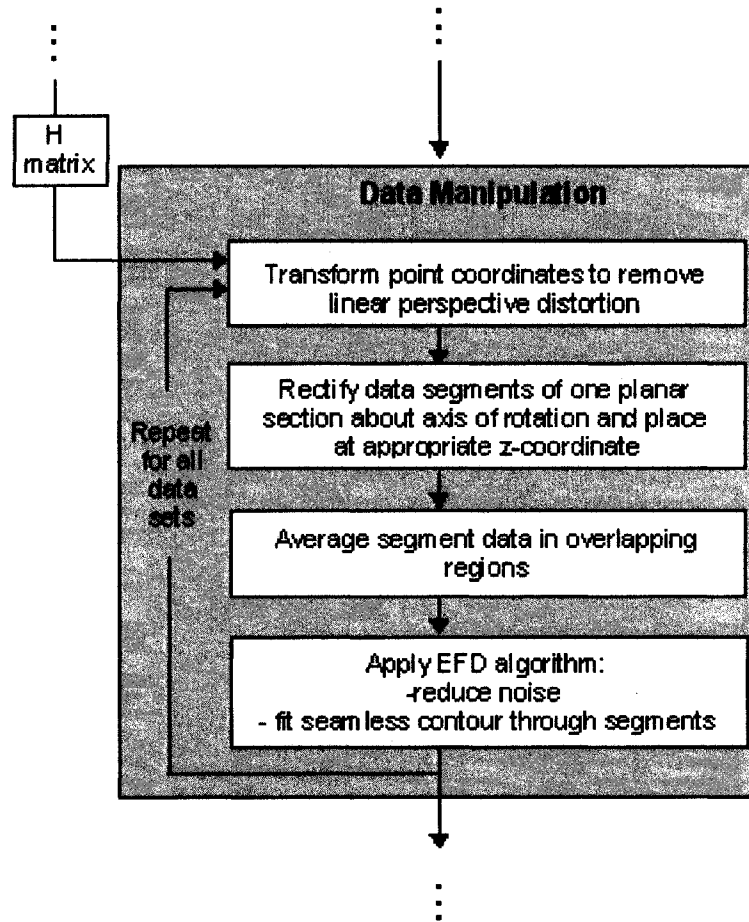
Figure 6.1: Diagram of Chapter 6 organization from system flow diagram.

6.1 Transform Point Coordinates to Remove Distortion

As described in Chapter 4, the camera used to capture images is located at a fixed angle α to the laser plane, hence the images contain perspective distortion which must be compensated for. The goal of the calibration process presented in Chapter 4 is to find a mapping which will remove this distortion; that is, a mapping between the illuminated pixel coordinates of the laser trace in the image to the world coordinates

of the corresponding surface points on the object in the xy-plane. This mapping is a 3×3 projective transformation matrix H. Recall from Chapter 5 that since all the laser trace segments are converted into sets of (x, y) point coordinates at the image processing stage, the process of transforming the laser trace coordinates into real world coordinates is the simple matrix multiplication given in Equation 4.1 and repeated again below in Equation 6.1.

$$\mathbf{p}'_i = H\mathbf{p_i}, \qquad (6.1)$$

where $\mathbf{p}_i = (x_i, y_i, w_i)$ are the homogeneous coordinates of the laser trace point being transformed, $\mathbf{p}'_i = (x'_i, y'_i, w'_i)$ are the homogeneous coordinates of its corresponding point in real world coordinates with the perspective distortion removed, and H is the mapping found through calibration.

This transformation is performed for each point of a given laser trace segment, and repeated for each segment captured during the scan. An example of an entire transformed laser trace for a cylindrical object is shown in Figure 6.2.

(a)   (b)
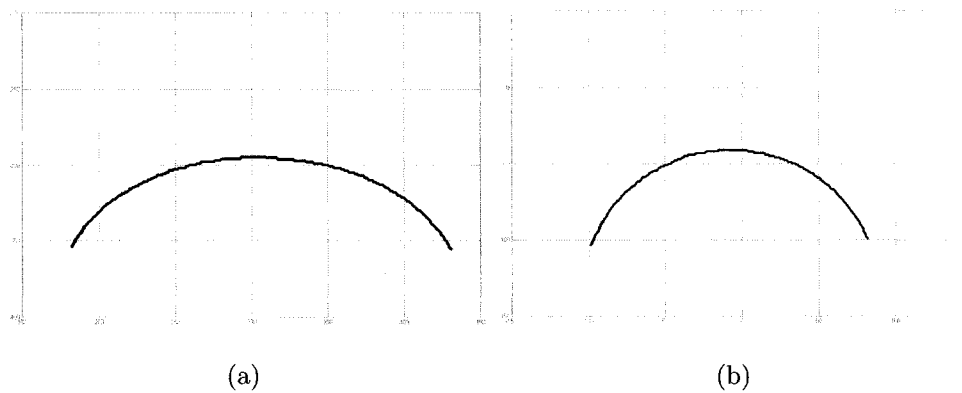
Figure 6.2: Figure of a laser trace (a) before and (b) after the point coordinates are transformed using H to remove perspective distortion.

Due to the poor quality of the light signal where it intersects the outermost edges of the object, the end regions of the data segments are typically plagued with noise,

excessively steep slopes, and frequent outliers [46]. For this reason, the first and last 10% of data points from each segment are discarded at this point in the data processing.

6.2   Rectify Data Segments About a Central Axis

In order to reassemble the data segments into complete planar sections, a suitable axis perpendicular to the plane of the image must first be established. Intuitively, this axis should be the object's axis of rotation however, determining the location of this axis within the images is not a trivial process, especially if the object is placed off-centre on the turntable. The axis of rotation must be determined irrespective of the object data collected; this is accomplished using a calibration cylinder of known diameter. The procedure for finding the axis of rotation and reassembling object data is as follows:

1. Eight images of the calibration cylinder are captured in rotational increments of $R = 45°$ in order to create one complete planar section.

2. The images are processed according to Chapter 5 and the resulting point coordinates are transformed according to Section 6.1. The example data segments shown previously in Figures 5.7 and 6.2 are from such a calibration cylinder and demonstrate the arc-shaped nature of the segments.

3. The centres of the arc segments of the calibration cylinder are determined by fitting a circle to each segment (See Section 6.2.1 for more details).

4. These arc centres form a circular cluster, the centroid of which is calculated; this centroid becomes an estimate of the axis of rotation (See Section 6.2.2 for more details).

5. Using the centroid of the arc centres as a reference frame origin, coordinates of the *object* segments are translated accordingly, and rotated to their respective orientation about this origin (See Section 6.2.3 for more details).

6. All data is scaled according to a scaling factor, also determined from the calibration cylinder (See Section 6.2.4 for more details).

Steps 3 through 6 of this procedure will now be explained in more detail.

6.2.1 Finding arc centres

The procedure for finding the arc centres is based on algorithms presented in [80] and [81]. The Matlab code used to carry out this procedure is credited to [82]. The algorithm of [80] takes a set of data points and finds the best circle fit by minimizing the normal distance from points of the segment to the circle circumference in an RMS sense. This algorithm returns the centre coordinates of the fitted circle and its radius. This result is used as the initial estimate for an improved fit computed according to [81]. This procedure is iterated by using the improved fit as the next best estimate for subsequent fits until a termination criterion is met. Termination occurs when the percent difference in mean squared error between the new fit and the previous fit falls below $\frac{0.1}{K}$, where K is the number of points in the arc, or when 500 iterations is reached, whichever occurs first. The centres of the final best fit circle for each of the eight arc segments are considered to be the arcs' respective centres.

6.2.2 Finding the centroid of arc centres

The arc centres of the eight calibration cylinder arc segments form a cluster about the true axis of rotation. This is shown in Figure 6.3 where the arc segments and arc centres have been plotted on the same axis, and the arc centres form a circle. The centroid is computed such that it minimizes the sum of point-to-centroid distances,

where distance refers to the squared Euclidean distance. More simply, the centroid of the cluster is the mean of the points in the cluster. The calculated centroid is used as an estimate for the true axis of rotation.

The calibration cylinder has been placed off-centre on the turntable in the example shown in Figure 6.3 to magnify the distance between arc centre locations. If the cylinder was to be centred exactly on the turntable, the arc centres *and* their computed centroid would all lie on the axis of rotation. Since it is not a likely scenario that this can be accomplished with certainty, the centroid of the arc centres is used as an estimate of the true axis of rotation. In a regular scenario, there would likely be much less deviation among the arc centre coordinates.
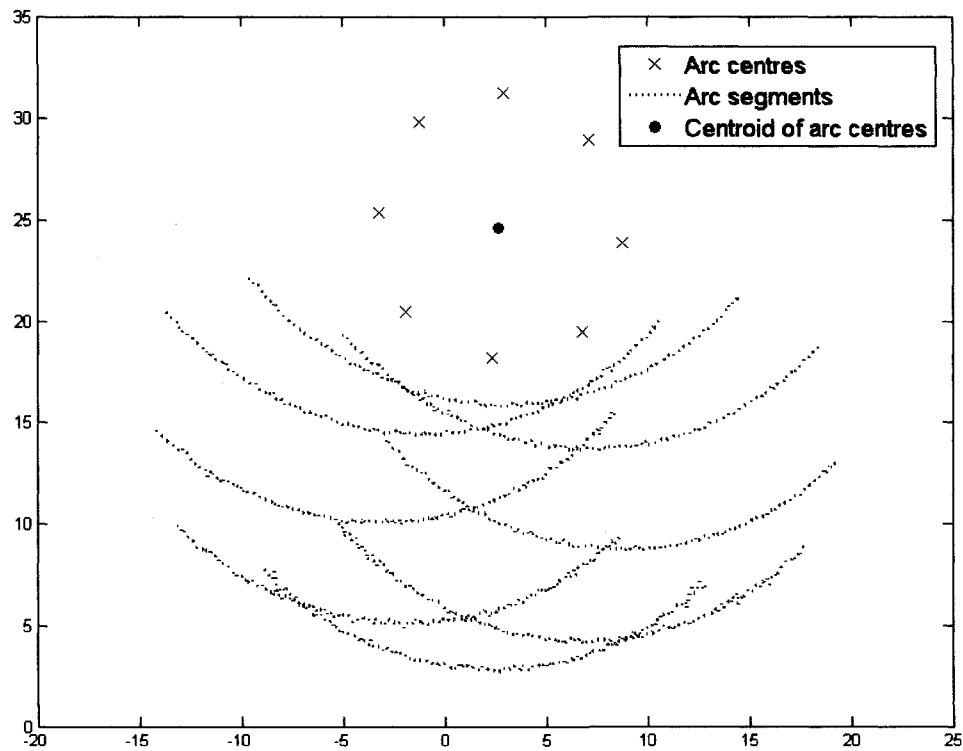
Figure 6.3: Diagram of arc segments, their computed arc centres, and the corresponding centroid of the arc centres.

6.2.3 Object segment reassembly

After all the object data has been transformed into the real world coordinate system, and the axis of rotation has been established, the segments are then reassembled about this axis. First, all point coordinates are translated such that the axis of rotation becomes a reference frame origin. Then, segments are rotated about this origin to their respective angles in increments of $45\,°$. This is accomplished by simply multiplying each point in a given segment by the 2×2 rotation matrix which corresponds to the angle for that segment, according to Equation 6.2,

$$
\begin{bmatrix} x_i' \\ y_i' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix},
\tag{6.2}
$$

where (x_i, y_i) is the ith point to be rotated, (x_i', y_i') is the corresponding rotated point and θ is the required angle of rotation in radians.

6.2.4 Data correction factor

All data are multiplied by a correction factor so that object dimensions are correctly reflected in mm. The correction factor is computed using the same calibration cylinder data used above for determining the axis of rotation. The known diameter of the cylinder is compared to the diameter of the cylinder found using the data collected. The correction factor C_f is calculated by:

$$
C_f = \frac{D_a}{D_e},
\tag{6.3}
$$

where D_a is the actual diameter and D_e is the estimated diameter based on the arc data segments. Ideally, the correction factor should be 1, however due to error in the calibration process the collected data may vary slightly from this ideal. If the

percent difference between the ideal diameter and the estimate is greater than 5%, the software produces a warning as there may be a more serious problem with the reconstruction process up to this point. Otherwise, the data for the object of interest is corrected such that it is in accordance with the calibration cylinder dimensions.

6.3 Average Segment Data in Overlapping Regions

Once the data segments have been translated to the reference frame origin, rotated to their respective orientations, and corrected appropriately, substantial overlapping exists between the segments. This is demonstrated in Figure 6.4 which shows the eight data segments for a cylinder object, with line thicknesses exaggerated in order to distinguish segments from each other. Segment 1 is indicated and subsequent segments are numbered counterclockwise.

Before smoothing the segments and fitting them with one seamless contour these overlapping regions are averaged together. As suggested by [1], defining new estimates of actual surface points by averaging samples from overlapping scans is an approach to dealing with residual error in the data. This is only a preliminary method for noise reduction, which will be addressed further in Section 6.4. Figure 6.5 shows an exaggerated and magnified view of two overlapping segments and helps illustrate the process used for averaging overlaps, which is described below.

1. Polar coordinates (r, θ) with $\theta \in \{0, 2\pi\}$ are computed for each (x, y) point.

2. The start (OL_S) and end (OL_E) points of the overlapping region are defined.

3. For each point on segment 2, the reference segment, with a corresponding point on segment 1 at the same θ value, a new averaged point is computed as the
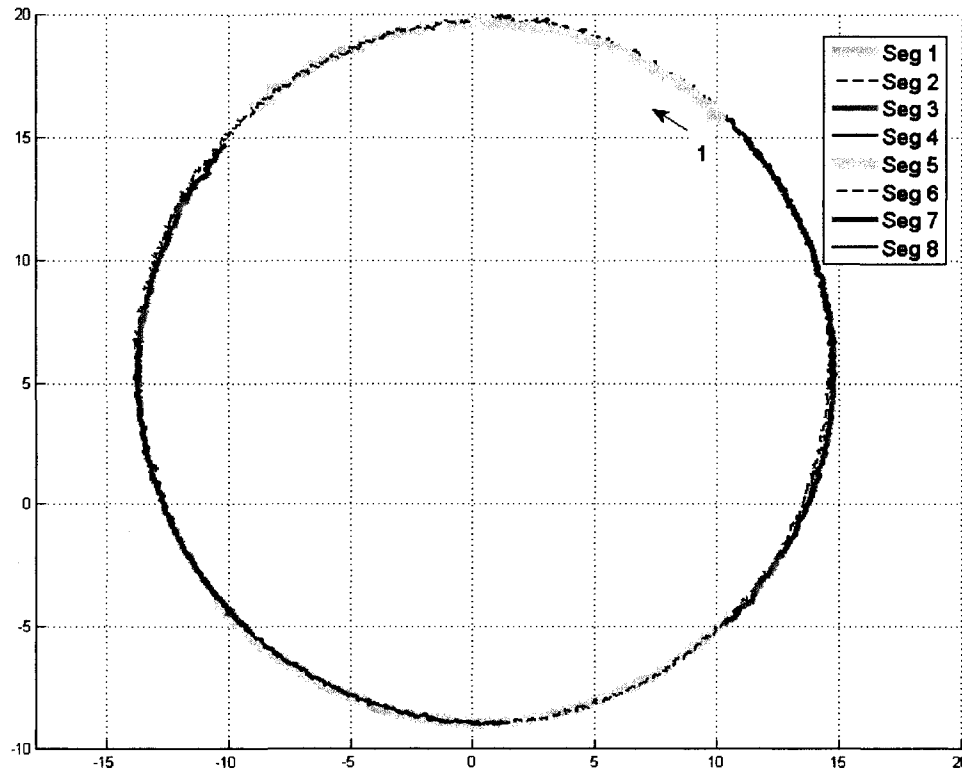
Figure 6.4: Diagram of translated and rotated segments for a cylinder object demonstrating segment overlapping.

mid-point, radially, between the points. This is demonstrated in Figure 6.5 by $r1_{s2}$ and $r1_{s1}$, both of which are at $\theta1_{s2}$.

4. If for a given point on segment 2 no corresponding point exists on segment 1, then a point is linearly interpolated and used to compute an average point between the two segments at that particular value of θ. This is demonstrated in Figure 6.5 by $r2_{s2}$ and the interpolated point $r2'_{s1}$ which are both now at $\theta2_{s2}$.

5. The same procedure is repeated using segment 1 as the reference segment and points on segment 2 are matched (or interpolated) in order to compute average points for each point on segment 1.

6. The two sets of averaged points are merged.

7. The portions of segment 1 and segment 2 that lie between (OL_S) and (OL_E) are replaced with the set of new averaged points.

8. This process is repeated for each overlapping region of each planar section.
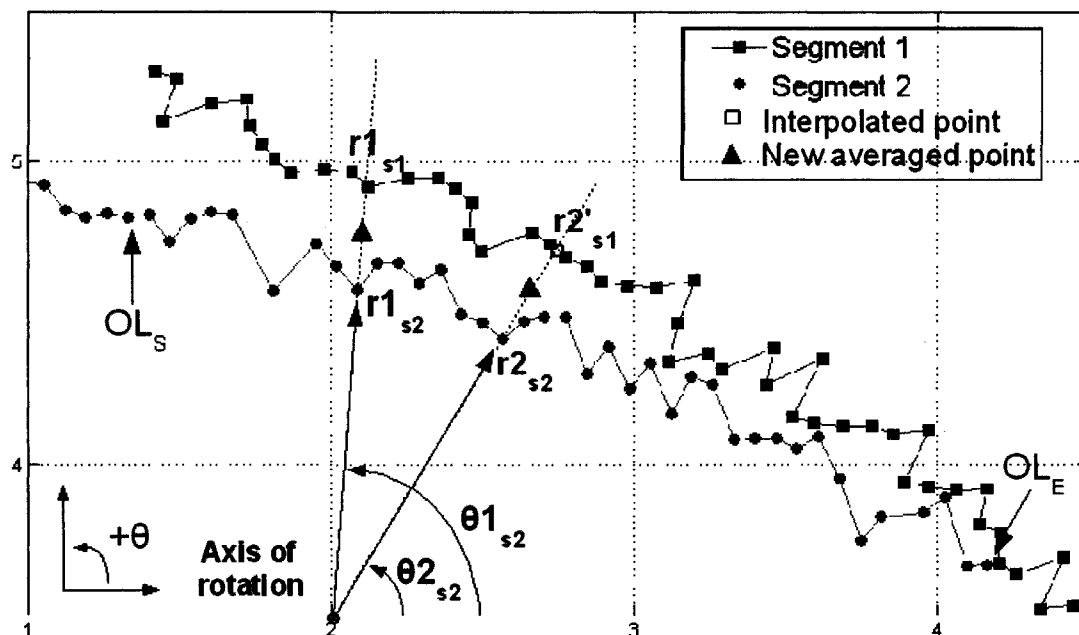
Figure 6.5: Exaggerated overlapping regions with averaged point indicated for both normal and interpolated cases.

6.4 Elliptical Fourier Descriptor Algorithm

The final step of the data manipulation stage is curve fitting. Recall from Section 3.1.2 that this step is required because when employing the laser light-sectioning technique, the segments of data are inherently noisy, furthermore the object model must be reconstructed by aligning numerous segments of data into complete planar contours. For the laser light-sectioning scanner presented in this thesis Elliptical Fourier Descriptors (EFDs) are used for data smoothing and planar section reconstruction.

6.4.1 Introduction to Elliptical Fourier Descriptors

Kuhl and Giardina [59] were the first to introduce the concepts of Elliptical Fourier Functions (EFFs) and Elliptical Fourier Descriptors (EFDs). Building upon the theory of conventional Fourier descriptors (see Appendix C), Kuhl and Giardina proposed that a 2D continuous closed contour can be represented, parametrically, as a function of time, $V(t)$ (Figure 6.6 (a)). Projections of this vector function on the x- and y-axes, represented by $x(t)$ and $y(t)$, are periodic with period T, where T is the time required to trace the entire contour at a constant speed (Figure 6.6 (b))

(a)   (b)
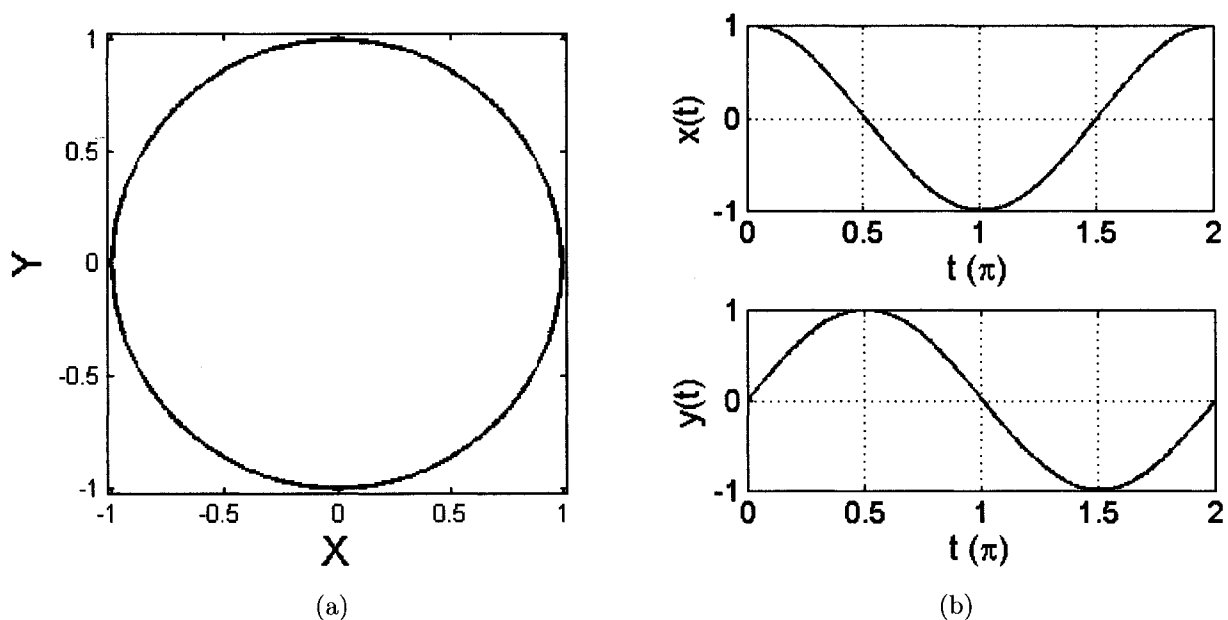
Figure 6.6: (a) a unit circle and (b) its x and y components plotted as functions of the parameter "time", t. T in this case is 2π.

These projections can be represented by Fourier trigonometric series, as given in Equations 6.4 and 6.5, and are referred to as Elliptical Fourier Functions.

$$x(t) = A_0 + \sum_{n=1}^{\infty} a_n \cos \frac{2\pi n t}{T} + \sum_{n=1}^{\infty} b_n \sin \frac{2\pi n t}{T}, \tag{6.4}$$

$$y(t) = C_0 + \sum_{n=1}^{\infty} c_n \cos \frac{2\pi nt}{T} + \sum_{n=1}^{\infty} d_n \sin \frac{2\pi nt}{T}, \tag{6.5}$$

where

$$A_0 = \frac{1}{T} \int_0^T x(t)dt, \tag{6.6}$$

$$a_n = \frac{2}{T} \int_0^T x(t) \cos \frac{2\pi nt}{T} dt, \tag{6.7}$$

$$b_n = \frac{2}{T} \int_0^T x(t) \sin \frac{2\pi nt}{T} dt. \tag{6.8}$$

(Similar expressions can be derived for C_0, c_n, and d_n.)

Different levels of approximation to the closed contour represented by $x(t)$ and $y(t)$ can be obtained by truncating the Elliptical Fourier Functions after different numbers of harmonics [83] as given by Equations 6.9 and 6.10 where n equals the harmonic number and N equals the maximum harmonic number before truncation. Complex object profiles are thus represented with higher order harmonics. Figure 6.7 shows the EFD fit found for a square using 1, 3, 6 and 12 harmonics.

$$x_N(t) = A_0 + \sum_{n=1}^{N} a_n \cos \frac{2\pi nt}{T} + \sum_{n=1}^{N} b_n \sin \frac{2\pi nt}{T}, \tag{6.9}$$

$$y_N(t) = C_0 + \sum_{n=1}^{N} c_n \cos \frac{2\pi nt}{T} + \sum_{n=1}^{N} d_n \sin \frac{2\pi nt}{T}, \tag{6.10}$$

Kuhl and Giardina also developed formulae for the coefficients A_0, C_0, a_n, b_n, c_n, and d_n which do not require the evaluation of integrals; these are given in Equations 6.11, 6.12 and 6.13. For a more complete explanation of the derivation of these coefficient expressions, see Appendix C.
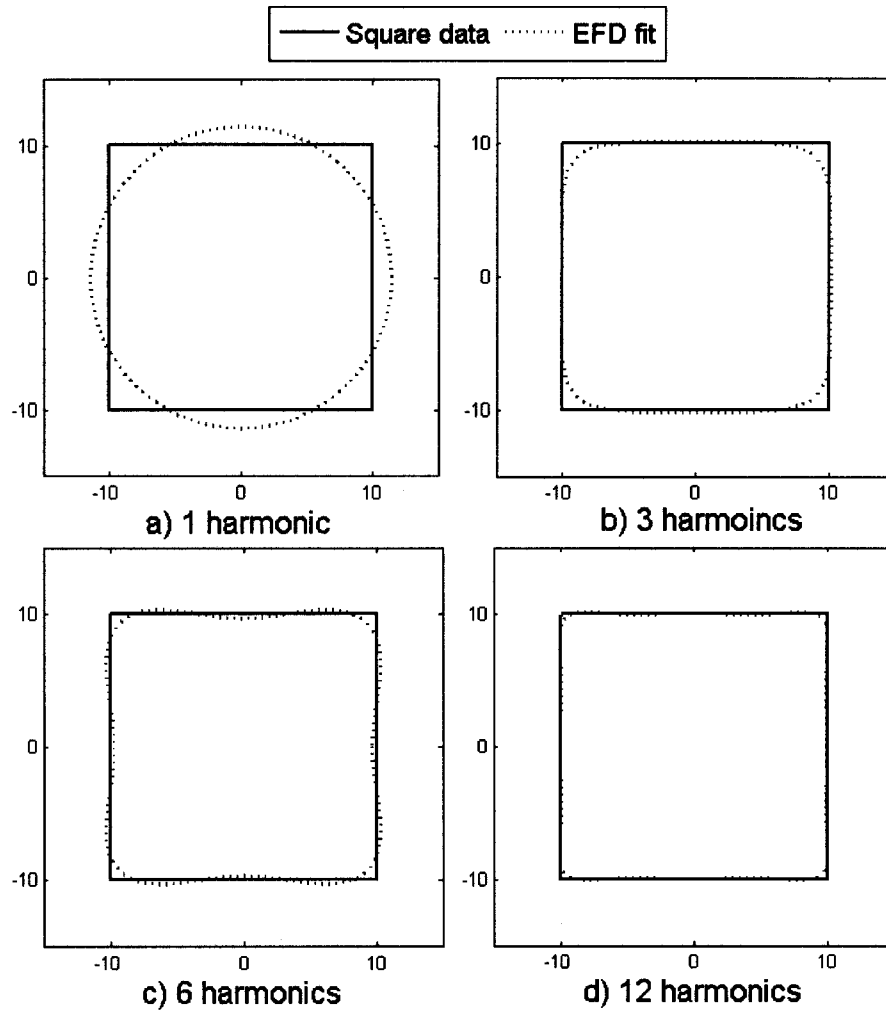
Figure 6.7: EFD fit for a square using 1, 3 ,6, and 12 harmonics.

$$A_0 = \frac{1}{T} \sum_{p=1}^{K} \frac{\Delta x_p}{2 \Delta t_p} (t_p^2 - t_{p-1}^2) + \xi_p (t_p - t_{p-1}), \tag{6.11}$$

$$a_n = \frac{T}{2\pi^2 n^2} \sum_{p=1}^{K} \frac{\Delta x_p}{\Delta t_p} \left[\cos \frac{2\pi n t_p}{T} - \cos \frac{2\pi n t_{p-1}}{T} \right], \tag{6.12}$$

$$b_n = \frac{T}{2\pi^2 n^2} \sum_{p=1}^{K} \frac{\Delta x_p}{\Delta t_p} \left[\sin \frac{2\pi n t_p}{T} - \sin \frac{2\pi n t_{p-1}}{T} \right], \tag{6.13}$$

where

$$\xi_p = \sum_{j=1}^{p-1} \Delta x_j - \frac{\Delta x_p}{\Delta t_p} \sum_{j=1}^{p-1} \Delta t_j \qquad (6.14)$$

(Similar expressions can be derived for C_0, c_n, and d_n.)

Elliptical Fourier Descriptors are broadly defined as the 'output' from the Fourier curve fit [84] and numerically describe the contour boundary information. With these descriptors, a contour can be re-created at any time in the absence of the original data.

EFDs have been shown to be particularly suited for fitting the contours of biological objects, further supporting their use for a biomedical scanner. This has been demonstrated in [85], where they are used to quantify the shape of human mandibles in order to analyze variability and sexual dimorphism between individuals, and in [86], where they are used to fit contours of the prostate from a series of computed tomography (CT) scans in order to construct a 3D model. This work also explores the interpolation of EFD coefficients *between* parallel horizontal contours, so that a new set of 'slices' can be generated at any location along the z-axis. A survey of many other biological applications of EFDs is presented in [84].

We propose that EFDs [59] are an appropriate curve fitting technique for a laser light-sectioning scanner, both for reducing noise in the individual data segments and for fitting a single curve through the data segments. Furthermore, we assume that for *biomedical* applications, the surfaces being scanned will be smooth and the use of EFDs is thus suitable. The scanning of non-smooth surfaces is beyond the scope of this thesis.

6.4.2 Methods

To demonstrate the validity of this approach, experiments were conducted on both simulated data and real data collected from a small portion of a simple cylindrical

test object. The Matlab code used for fitting EFDs to the data contours is credited to [87].

Simulated data

For the simulation, EFDs were used to fit a set of ideal input data points with additive white Gaussian noise (AWGN). This would demonstrate the effectiveness of EFDs for reducing random noise in the absence of systemic noise. Random white Gaussian noise with zero mean and a variance of $\sigma^2 = 3.9 \times 10^{-3}$ was added to a circle data set with an ideal radius of $r = 9.525$ mm ($\frac{3}{8}$ in). The chosen variance is an estimate of the variance in the random noise observed in the data of previous experiments.

Real data

Images for $n = 4$ planar sections of a test object were captured; this test object was a cylinder with radius r $= 9.525$ mm ($\frac{3}{8}$ in). For each plane, $M = 8$ images of the laser trace on the object were captured in rotational increments of $R = 45°$. Planar sections were captured at 3.175 mm ($\frac{1}{8}$ in) increments in the negative z-axis direction. It should be noted that since this experiment was conducted with the first prototype that was developed, the object had to be manually turned in its place and manually raised using 3.175 mm ($\frac{1}{8}$ in) gauge blocks. In total, 32 images of laser trace segments were captured for reconstructing the four planar sections.

Images were processed according to Chapter 5 (with the line thinning algorithm implemented for point coordinate identification), and the resulting discrete sets of point coordinates were transformed, aligned, and averaged according to the processes described in Sections 6.1 to 6.3. EFDs were used to fit a smooth seamless curve through the points comprising each planar section. In this case, only 1 harmonic was required to accurately fit the test cylindrical object. The final object contours were stacked along the reconstruction z-axis.

6.4.3 Results

Figure 6.8 shows a profile of the radii of the EFD fit for the simulated circle data with AWGN. Figure 6.9 is a plot of the radii obtained from the real data, both before and after the contour was fit with a smooth curve using EFDs. Only the first planar section is shown; however, results were similar for all four layers that were reconstructed for this experiment. As indicated, the ideal radius in this case was $r = 9.525$ mm ($\frac{3}{8}$ in). Figure 6.10 shows the four planar sections of the test cylinder, fitted with EFDs and scaled appropriately.
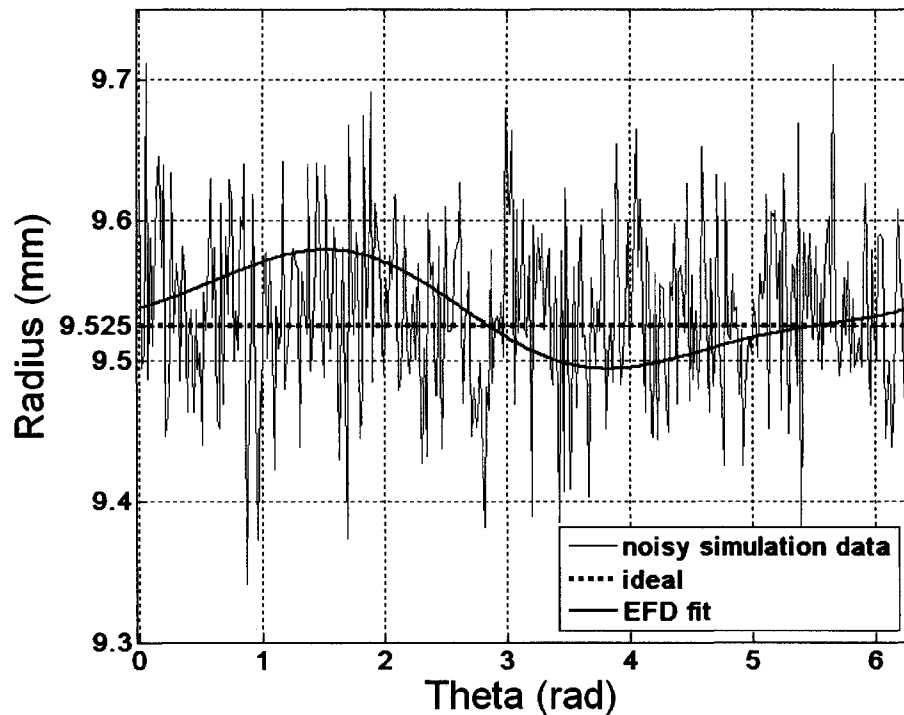
Figure 6.8: Profile of simulation data (circle with r = 9.525 mm). Radius shown as a function of theta from 0 to 2π for the noisy data before the EFD fit and after the EFD fit.
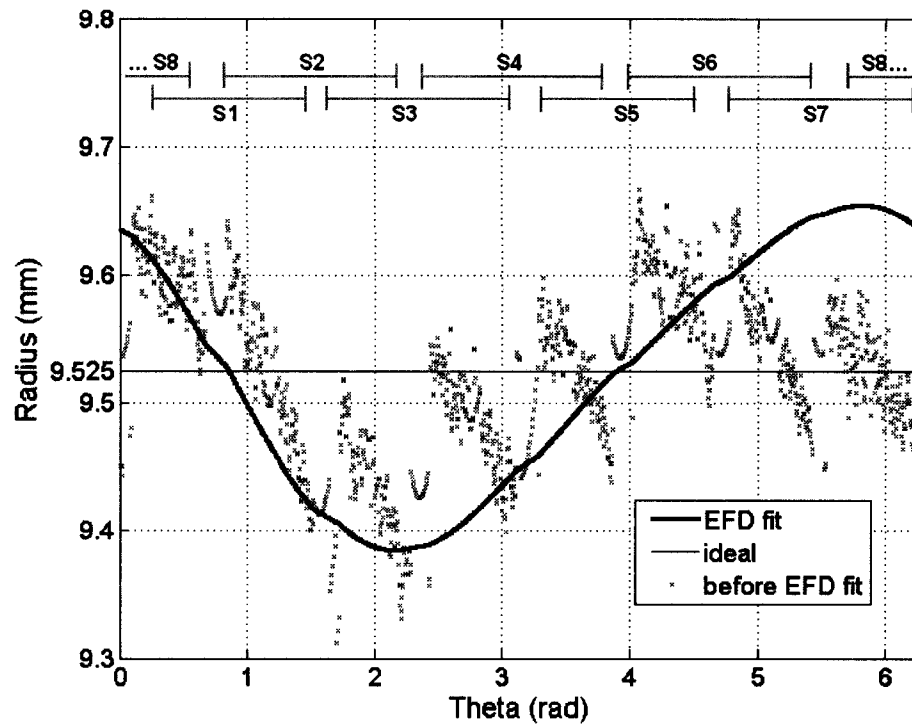
Figure 6.9: Profile of test object (cylinder with r = 9.525 mm). Radius shown as a function of theta from 0 to 2π for the raw data before the EFD fit (constructed from 8 segments, S1-S8) and after the EFD fit.

6.4.4 Discussion

In the simulated scenario shown in Figure 6.8, the noise is strictly random. The EFD fit provides a marked improvement over the noisy data; error does not exceed 0.1 mm in this case. As shown in Figure 6.9, the rectified segments for the given planar section of the test cylinder appear to produce a fairly good result, even before a curve was fit to the data with the EFD technique. The percent error between the radius of the reconstructed test object and the actual radius of the object is about 1 to 2%. Errors do not exceed 0.25 mm. The subsequent EFD contour for this first layer shows a fairly close fitting through the points using only 1 harmonic.

Although sub-mm accuracy is achieved overall, there is observable noise in the error plot of Figure 6.9. This noise appears to be made up of a random noise component and a systemic noise component. Since no smoothing was performed on the data
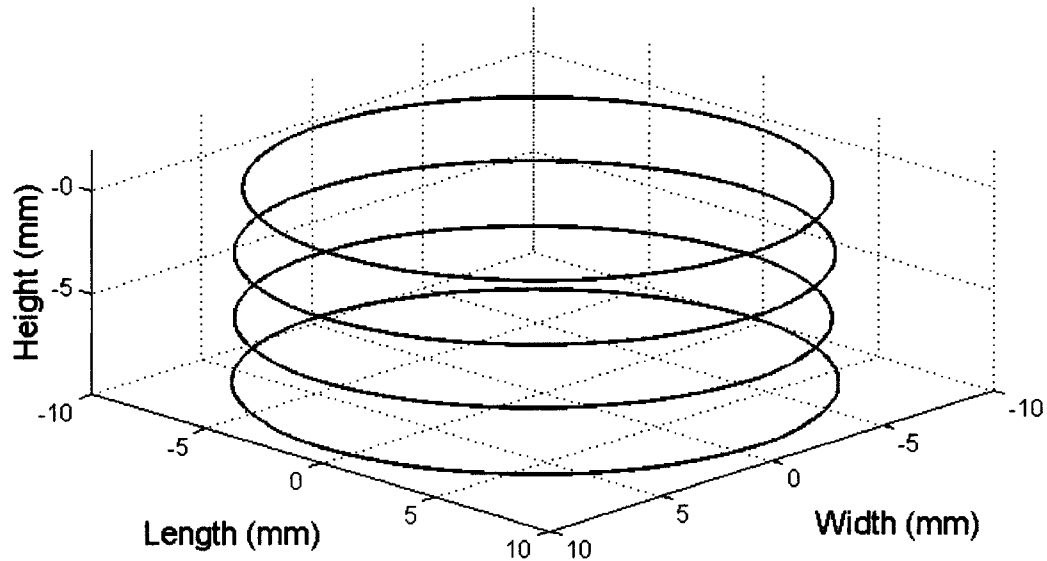
Figure 6.10: Four reconstructed planar sections of a test cylinder fitted with EFDs

before the EFD fit (except for averaging the overlapping regions), the random error could be due to noise inherent in the measurement system and noise in the resulting point coordinates obtained from the line thinning algorithm. Note the COG method was not implemented at the time of this analysis. This random noise is clearly attenuated in the EFD fit. The systemic noise however, is present in the data both before and after the EFD fit. The most prominent sources of this error for this setup were procedural in nature, such as inaccuracies associated with manually turning the object in its place. Also, in the unfitted data, one can observe obvious jumps in the radii obtained where segments were fused together and a general downwards trend in the radii for these different segments. This could be due to misalignment of the test object about the reconstruction axis after manually raising it with the gauge blocks, resulting in a slight shift in the segments. All of these systemic errors are associated with limitations of the first prototype setup, and should be attenuated in the second version of the prototype in which the camera and laser diode head are automatically incremented vertically on a platform and the object is rotated automatically on a

turntable. Although the EFD fit does not appear to have a significant effect on the overall error in these results, it is anticipated that it will with these systemic noise sources addressed. This claim is substantiated with the simulated scenario presented above in Figure 6.8 where the noise is strictly random.

Due to the shape of the object used for this experiment, the number of harmonics chosen to accurately fit the data using EFDs was 1. Further work has been done to develop a preprocessing technique that will automatically identify an appropriate number of harmonics to use for a given set of data segments (see Section 6.4.6), since not all objects imaged would have circular symmetry as in this test case. This would result in accurate contour fitting without excessive data processing.

Finally, visual inspection of the reconstructed planar sections of the test cylinder (Figure 6.10) reveals that the proposed approach can produce a planar section model, which is reasonably complete in desired external detail.

6.4.5   Summary

In conclusion, the data presented shows the advantage of using EFDs for fitting a contour through the segment data obtained from a laser light-sectioning scanner. Although the EFD technique does not provide a substantial improvement in overall accuracy, random error is attenuated considerably, resulting in a smooth curve that closely matches the dimensions of the object of interest. Sub-mm accuracy is achieved, both before and after the EFD fit, where the majority of the errors are the result of manual manipulation of the device. After conducting this analysis, there was good reason to believe that an acceptable level of accuracy could be achieved with future iterations of the design, where sources of systemic error would be reduced. Therefore we conclude that using EFDs for the curve fitting stage of the data manipulation process is appropriate for this application.

6.4.6   Choosing an appropriate number of harmonics

As mentioned above, some work has been done to develop a preprocessing technique that will automatically identify an appropriate number of harmonics to use for a given set of data segments. This solution is not ideal but has been implemented here for convenience so that harmonic number selection is not completely arbitrary.

A unique harmonic number HN is found for each planar section of data segments and is based on two parameters: the *perimeter/area ratio* PA for that section and a *roundness factor* RF. The perimeter pm for a given section is determined from the set of point coordinates for that contour by adding the distances between adjacent points. The area a is found with a function in Matlab which computes the area of the polygon formed by the set of point coordinates. PA is simply the ratio of these two. RF is a simple metric between 0 and 1 which indicates how 'round' the contour is (Equation 6.15). For a perfect circle, RF is 1.

$$RF = \frac{4\pi a}{pm^2} \tag{6.15}$$

The harmonic number calculation for the kth section is given in Equation 6.16 and would be rounded to the nearest integer.

$$HN_k = \frac{1}{PA_k \times RF_k} \tag{6.16}$$

Figure 6.11 demonstrates how this harmonic calculation works for simulated data of a square, pentagon, octagon, and arbitrary blob object. The HN values calculated for these data sets were 6, 5, 5, and 18 respectively.

Because the pentagon and octagon are more 'round' than the square, they required fewer harmonics. The fitting works well for the blob contour however, the algorithm is not very robust for objects with sharp corners. Since the target application for this
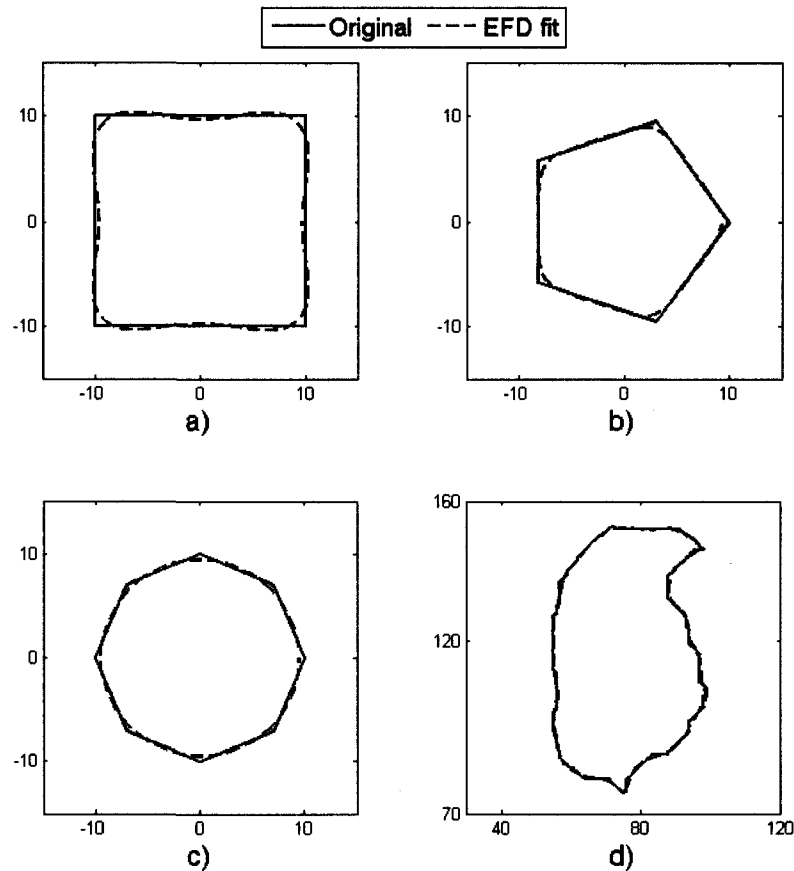
Figure 6.11: EFD fit using calculated HN for a (a) square (b) pentagon (c) octagon and (d) arbitrary blob object

scanner will be smooth objects, it is assumed that the HN metric is sufficient for harmonic calculation at this stage in the research.

This topic has been suggested as an area for future work (see Section 9.2). There is a *maximum* number of terms which can be taken in the series since it is subject to Nyquist frequency requirements; that is, higher harmonics whose wavelengths are less than twice the spacing between sampled points cannot be detected [84]. In this context, where we are not exactly sampling data from a signal but rather enforcing that all measured points are a fixed unit of time apart and that all points are measured in one cycle, the number of points on the planar section contour effectively defines

the 'sampling rate', F_s. According to the Sampling Theorem [88], this rate must be greater than or equal to twice the maximum frequency in the signal to avoid aliasing. In other words, the maximum frequency (harmonic) which can be taken for the EFD approximation is $N_{max} = \frac{F_s}{2}$ (or $N_{max} = \frac{F_s-1}{2}$ when F_s is odd).

Many applications tend to just use N_{max} because no information is lost that way, and with the computing power available today, it is probably not computationally impractical to do so. That being said, a number less than this maximum can also be chosen, which is what Equation 6.16 is meant to accomplish. It is possible to get very good approximations of the original contour with relatively few harmonics [84]. Otherwise, for a contour approximated with $N_{max} = \frac{F_s}{2}$ harmonics, there will be $2F_s + 2$ coefficients to store, which is not a savings compared to the number of points in the original data set.

One approach to selecting the number of harmonic components to be used would be to plot the harmonic amplitude against harmonic number to produce a power spectrum. This allows one to assess the contribution of each harmonic to the overall form. Furthermore, since successive harmonic numbers do not necessarily explain decreasing amounts of variance, important information might be lost if only lower harmonics are kept. Therefore, rejecting harmonics based on their low power rather than a high harmonic number might be a better option.

As suggested by Kuhl and Giardina [59], it could be useful to specify the number of harmonics required such that the EFD approximation to the contour be in error by no more than a specified error value ϵ. They provide a definition as well as a bounding expression for ϵ which can be solved for N, the maximum number of harmonics. However, with this approach the problem then becomes how does one choose the maximum ϵ such that the EFD approximation is fitting the curve closely enough, but not *so* closely that it is also representing noise in the contour, which would defeat the purpose of using EFDs as a noise reduction technique.

Chapter 7

Object Reconstruction

This chapter describes the object reconstruction process for the proposed 3D laser light-sectioning scanner system. The relevant portion of the system flow diagram from Section 3.1.2 is repeated in Figure 7.1 for reference. The goal of object reconstruction is to convert the point cloud of measured data points into a 3D continuous surface. Section 7.1 describes the data export process required to prepare the data for reconstruction. Section 7.2 describes how a triangulated mesh is applied to the data in order to obtain a complete 3D model.

7.1   Exporting into Wavefront Object File Format

The data are first exported from Matlab into a Wavefront Object file ($.obj$) in order to facilitate the object reconstruction process. The $.obj$ file format is a simple geometry definition file format which represents 3D geometry using the position of each vertex, the texture coordinate associated with a vertex, the normal at each vertex, and the faces that make each polygon [89]. It was originally developed by a computer graphics company called Wavefront Technologies (Santa Barbara, California) but has since become a universally accepted format, having been adopted by many other 3D
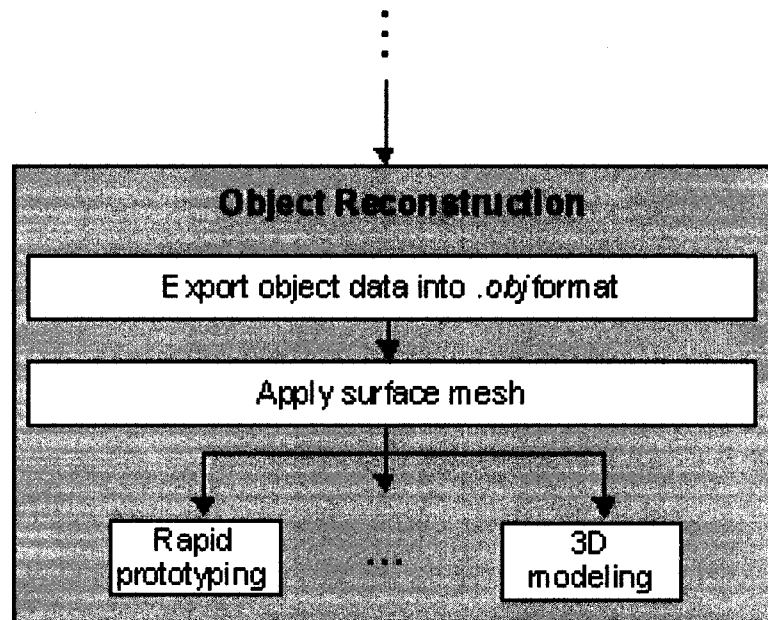
Figure 7.1: Diagram of Chapter 7 organization from system flow diagram.

graphics application developers, including Autodesk [90] (San Rafael, CA), the developer of Maya and 3D Studio Max, and open source initiatives such as Blender [91] (Amsterdam, the Netherlands) and MeshLab [75] (Pisa, Italy). An example of an *.obj* file for a simple cube object is provided in Figure 7.2.

This format is appropriate for this laser light-sectioning scanner due to the fact that it supports data which is strictly a 3D point cloud; that is, a data set entirely comprised of vertices (as denoted by the preceding 'v' in Figure 7.2). Texture, surface normals, and faces do not necessarily have to be defined in the data file. The *.obj* format is also universally accepted and compatible with many existing surface reconstruction software packages.

```
♯ This is a comment
♯ Vertices are preceded by 'v' and defined by their (x,y,z) coordinates.
v       0.0     0.0     0.0
v       0.0     0.0     1.0
v       0.0     1.0     0.0
v       0.0     1.0     1.0
v       1.0     0.0     0.0
v       1.0     0.0     1.0
v       1.0     1.0     0.0
v       1.0     1.0     1.0

♯ Normals are also in (x,y,z) form; normals might not be unit.

vn      0.0     0.0     1.0
vn      0.0     0.0     -1.0
vn      0.0     1.0     0.0
vn      0.0     -1.0    0.0
vn      1.0     0.0     0.0
vn      -1.0    0.0     0.0

♯ Texture coordinates would be preceded by vt.

vt      ...

♯ Faces are defined by listing all vertices which comprise the face on the same line, in the format 'vertex/texture/normal'
for a given vertex, where the values refer to the index of the desired vertex, texture, or normal as given by the definitions
above. The faces below are thus squares with no specific texture specified.
f       1//2    3//2    7//2    5//2
f       1//6    2//6    4//6    3//6
f       3//3    4//3    8//3    7//3
f       5//5    6//5    8//5    7//5
f       1//4    5//4    6//4    2//4
f       2//1    6//1    8//1    4//1
```

Figure 7.2: Example of the *.obj* file format for a cube object.

7.2 Applying a Surface Mesh

Meshlab [75], an "open source, portable, and extensible system for the processing and editing of unstructured 3D triangular meshes", was chosen as the surface reconstruction software for this scanner. Developed at the University of Pisa (Pisa, Italy), this program is ideal for processing the data that arises from 3D scanners, particularly for this system as it supports point cloud data and the *.obj* format. Once an *.obj* file is imported into the system, Meshlab builds a triangulated 3D surface reconstruction of the object by using the efficient ball-pivoting algorithm [69], explained further in Section 7.2.1.

Meshlab is also a suitable choice for object surface reconstruction because in addition to providing a set of tools for editing, cleaning, inspecting, and rendering surface meshes in its own environment, it is capable of exporting the final 3D model into 10 different 3D model file formats, including Stanford Polygon file (*.ply*), 3D Studio file (*.3ds*) and of course Wavefront Object file (*.obj*), making the final 3D model compatible with a number of different applications and versatile for whatever the desired use of the model may be (i.e., visualization, design, reverse engineering, etc.). For this laser light-sectioning system, the reconstructed object surface is again saved as an *.obj* file, however with more complete normal and face information now defined in the file.

7.2.1   Ball-pivoting algorithm for surface reconstruction

The ball-pivoting algorithm (BPA), developed by Bernardini *et al.* [69] in 1999, is based on the concept that three points in a point cloud form a triangle if a ball of a user-specified radius touches them without containing any other point. The BPA thus computes a triangle mesh as follows: a ball of user defined radius is placed in contact with three initial sample points, referred to as seed points. Keeping the ball in contact with two of these these initial points, the ball is pivoted about the edge between the two points until it touches another point, forming another triangle. "The process continues until all reachable edges have been tried, and then starts from another seed triangle, until all points have been considered" [69]. The set of triangles formed while the ball "rolls" along the object surface constitutes the triangulated 3D surface reconstruction of the object. The BPA is efficient in terms of execution time, exhibiting linear time performance on datasets consisting of millions of points [69]. Bernardini *et al.* also report that the algorithm is robust when it comes to handling noise in real scanned 3D data, and the quality of results obtained compare favorably with existing techniques [69].

Figure 7.3 shows the point cloud for four planar sections of a test cylinder object imported into Meshlab. Figure 7.4 shows the resulting triangulated surface mesh after applying the ball-pivoting algorithm to the data. The ball radius chosen for this example was 2mm and 599 triangular faces were generated in order to form the mesh. Figure 7.5 shows a rendered image of the 3D model with shading and lighting effects, also generated in Meshlab.
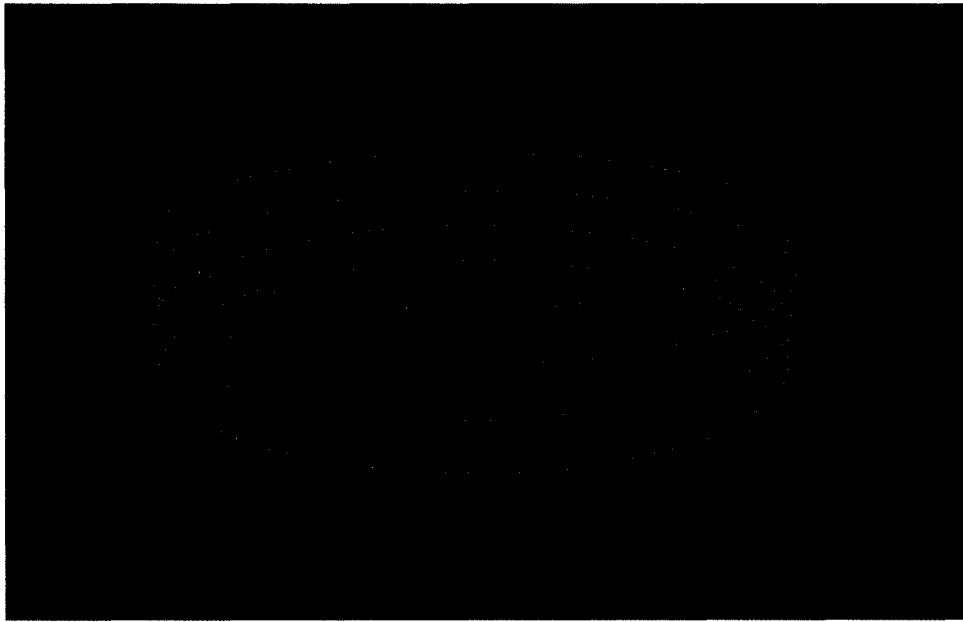
Figure 7.3: Point cloud for four planar sections of a test cylinder object imported into Meshlab.
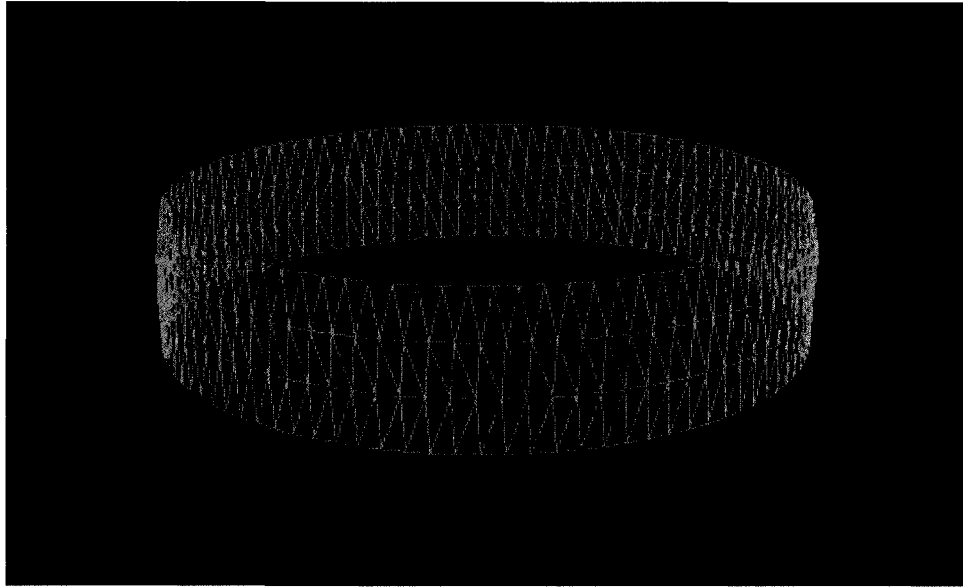
Figure 7.4: Triangulated surface mesh after applying the ball-pivoting algorithm to the point cloud data.
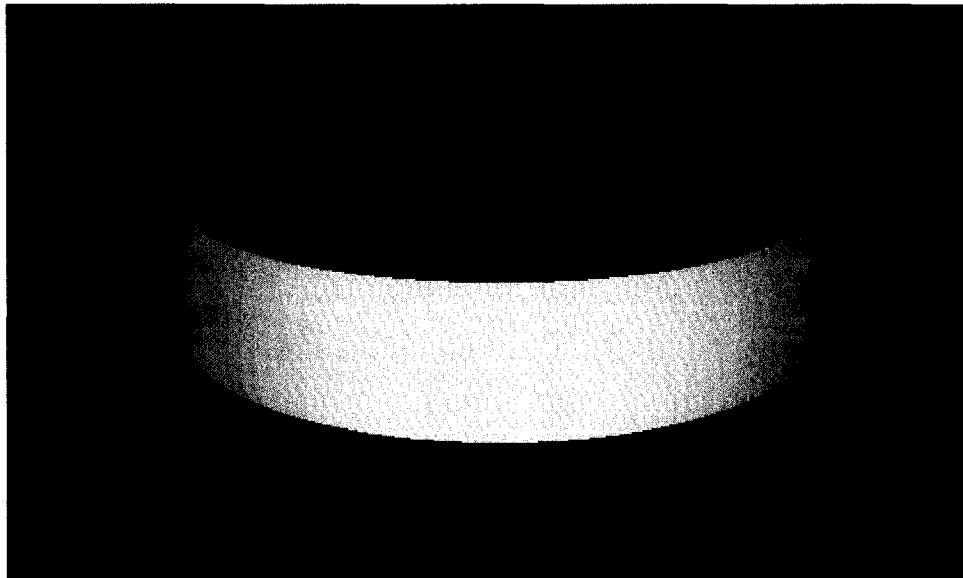
Figure 7.5: Rendered image of the 3D model.

Chapter 8

Overall Results and Discussion

This chapter presents the overall results obtained using the proposed 3D laser light-sectioning scanner system. Most of the results presented thus far throughout this thesis were collected using the first prototype setup (Prototype I). This initial setup allowed for us to collect data and perform the studies necessary (i.e., the calibration analysis of Chapter 4 and the EFD analysis of Chapter 6) to be able to make decisions about which final implementation to use. The majority of results presented in *this* chapter are those obtained from the second prototype setup (Prototype II), where the final chosen calibration, image acquisition, image processing, data manipulation, and object reconstruction processes have been implemented *together*, along with a more automated scanner apparatus itself.

Section 8.1 will review the methods for acquiring results using Prototype I and present the methods for acquiring results using Prototype II; this will help distinguish the differences between the two setups. Section 8.2 will present results from both prototypes, emphasizing those obtained using Prototype II. A discussion is presented in Section 8.3 followed by conclusions in Section 8.4.

Table 8.1: Summary of Differences between Prototype I and Prototype II.

Stage of System Process	Implementation	
	Prototype I	Prototype II
Calibration	Analytical method with 4 control point pairs	DLT LS method with > 20 control point pairs
Image acquisition	Object rotated ($R = 45°$) and incremented manually (3.175 mm)	Object rotated automatically ($R = 45°$), camera/laser platform incremented automatically (2 mm)
Image processing	Point coordinates determined using line thinning algorithm	Point coordinates determined using COG algorithm
Data manipulation	Overlapping segments averaged; planar sections fit with EFDs	Overlapping segments averaged; planar sections fit with EFDs
Object reconstruction	n/a	Ball-pivoting algorithm using Meshlab

8.1 Methods

To clarify the differences between the two prototype setups, Table 8.1 summarizes the different methods used according to the stages of the system flow diagram presented in Section 3.1.2.

Experiments using Prototype I were limited to a cylindrical test object with diameter 19.05 mm ($\frac{3}{4}$ in). Although discussed in Chapter 6, methods will be briefly reviewed here. The analytical approach, which only requires four control point pairs, was used to compute the transformation matrix H. The system to rotate the object and increment the camera/laser platform head was still under development, necessitating manual manipulation of the object. Images for $n = 4$ planar sections of the cylinder test object were captured in rotational increments of $R = 45°$. Planar sections were captured at 3.175 mm ($\frac{1}{8}$ in) increments in the negative z-axis direction by raising the object with respect to the laser plane using 3.175 mm ($\frac{1}{8}$ in) gauge blocks. Discrete sets of point coordinates representing the laser traces were determined using

a line thinning algorithm. Data were transformed using the H found from the analytical calibration process, then translated to, and rotated about, the axis of rotation. Overlapping segments were averaged and planar sections were fit with EFDs using 1 harmonic. The application of a mesh surface was not implemented for Prototype I.

The same test cylinder that was scanned with Prototype I was scanned again with Prototype II. As before, four sections were captured in rotational increments of $R = 45\,^\circ$, except this time sections were captured at 2 mm increments. Rotations and vertical increments were performed automatically. Discrete sets of point coordinates representing the laser traces were determined using a center of gravity approach. Data were transformed using the H found through calibration, this time using the DLT LS approach using 40 control point pairs, then translated to, and rotated about, the axis of rotation. Overlapping segments were averaged and planar sections were fit with EFDs using 1 harmonic. A mesh surface was computed for the cylinder using Meshlab software [75] and the ball-pivoting algorithm.

In addition to the test cylinder, two other objects were scanned with Prototype II, an oval cap and a miniature toy arm. Not all scans were acquired during the same session and therefore required separate calibrations (i.e., separate grid images were taken before the objects were scanned, resulting in variation in the isolated grid point locations and subsequent selection of control points). Table 8.2 summarizes the number of calibration points used, the vertical resolution, the number of sections captured, and the number of points in the EFD outline reconstruction for each object, including the test cylinder. The arm was coloured black to reduce surface reflectivity. All objects were scanned at an approximate distance of 10 cm (4 in).

Figure 8.2: Comparison of four reconstructed planar sections for (a) Prototype I and (b) Prototype II.

In order to demonstrate the dimensional accuracy of Prototype II in the xy-plane, Figure 8.3 shows the radius profiles of all four planar sections of the test cylinder that were captured, each section centered at its own centroid. This however cannot give a true picture of overall system accuracy as vertical alignment may vary from section to section. In order to show the scanner's precision in vertical alignment the centroids of each of the four planar sections are plotted in Figure 8.4, where the area shown is 1 mm^2, centered at the mean of all centroids $(-1.093, -6.573)$. To combine all sources of error, Figure 8.5 shows the radius profiles for planar sections centered about this mean centroid location.

Table 8.3 summarizes the root mean squared error (RMS) in point location along the section outline for each planar section, where the error is the difference between the ideal radius and the distance to each point on the section's outline from the section centre. Section reconstructions contain 500 points each. However, as explained above, since calculating the radius from the sections' respective centroids will not reflect errors in vertical alignment, RMS error is shown for error in radii calculated from both the sections' respective centroids and from the mean centroid.
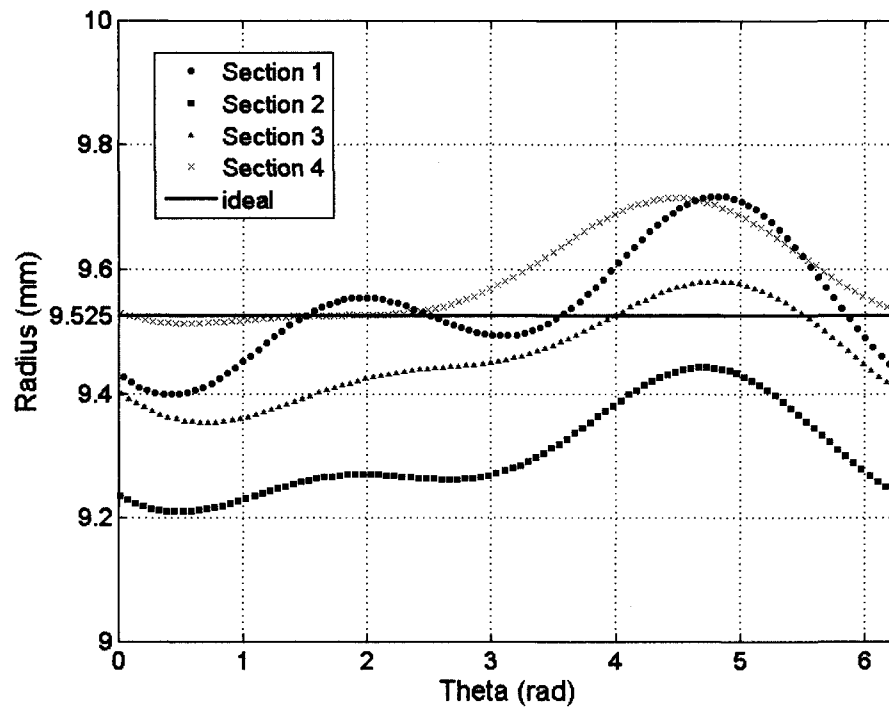
Figure 8.3: Radius profiles of four planar sections of the test cylinder captured with Prototype II. Radius is calculated from the centroid of each section.

8.2.2 Prototype II: Other objects

This section presents results for the two other objects scanned with Prototype II. Since it is difficult to determine system accuracy for these objects without ground truth information for comparison, Table 8.4 summarizes other results of interest for these objects: namely, the scan acquisition time, the number of harmonics calculated for the EFD fit, the number of vertices in the EFD point cloud, and the number of faces in the final surface reconstruction. Figures 8.6 to 8.8 show images of the original objects, plots of the EFD planar sections, and their 3D surface reconstructions, including the test cylinder discussed above. Scan time refers to the total time required for acquiring image data, including the time to rotate the object and increment the camera/laser platform. Data processing time refers to the CPU time for the computation of the EFD point cloud of planar sections from the scan images, and does not include the
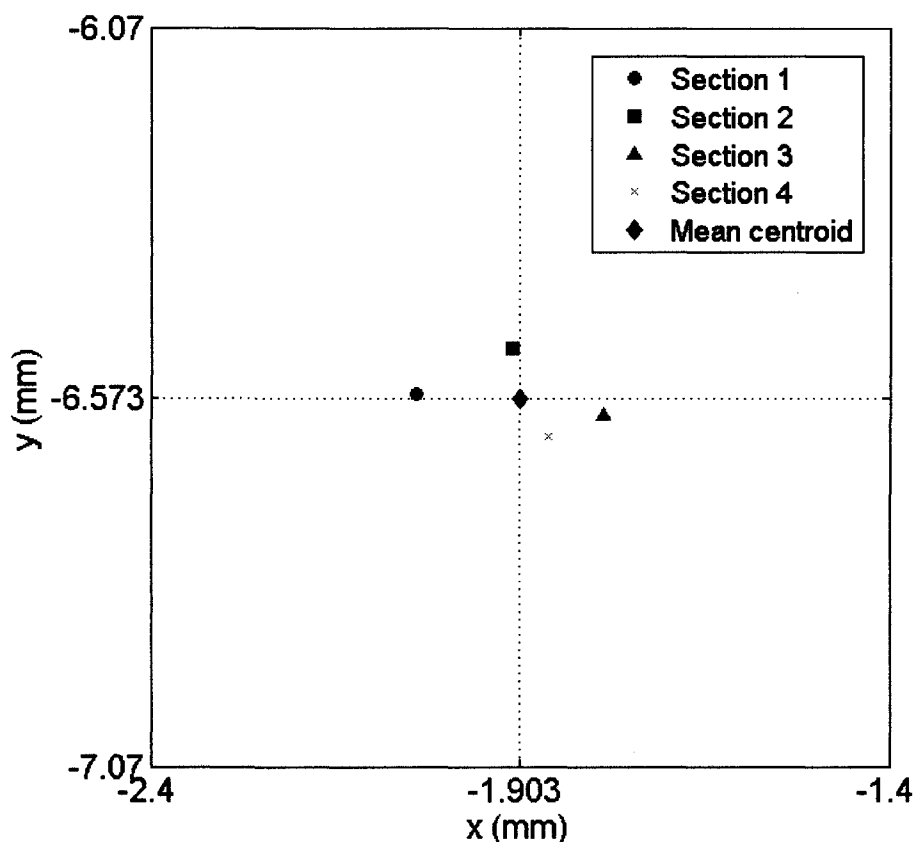
Figure 8.4: Centroids of four planar sections of test cylinder.

surface mesh reconstruction in Meshlab.

8.2.3 Prototype II: General results

Some other general results to report concern scan time versus acquisition time, and the overall system cost. The shutter speed of the camera is set to 1/60 s however this particular model can be set to as high as 1/10,000 s. The system takes about 40 s to capture the eight images required for one planar section, including the mechanical rotation time of the object. Actual image acquisition with Labview takes between 500 and 600 ms per image, meaning that actual image acquisition time is only about 10% of the total scan time. For example, for a 10 min scan consisting of 15 planar sections and 120 images, image acquisition time would only be 1 min. This is important

Table 8.3: RMS Error for Each Planar Section of the Cylinder Object Before and After Translating to the Mean Centroid.

Section	RMS Error (mm)	
	Relative to section centroid	Relative to mean centroid
1	0.0666	0.1235
2	0.2222	0.2245
3	0.0716	0.1076
4	0.0674	0.0861
Mean	*0.1069*	*0.1354*

Table 8.4: Summary of Prototype II Results.

Shape	Scan time	No. of harmonics (range)	No. of vertices	Data Processing time	No. of faces
Cylinder	2mins 40s	1	404	10.41s	600
Oval cap	6mins 40s	5 - 6	1010	20.28s	1894
Mini arm model	13mins 20s	5 - 8	2020	27.48s	2053
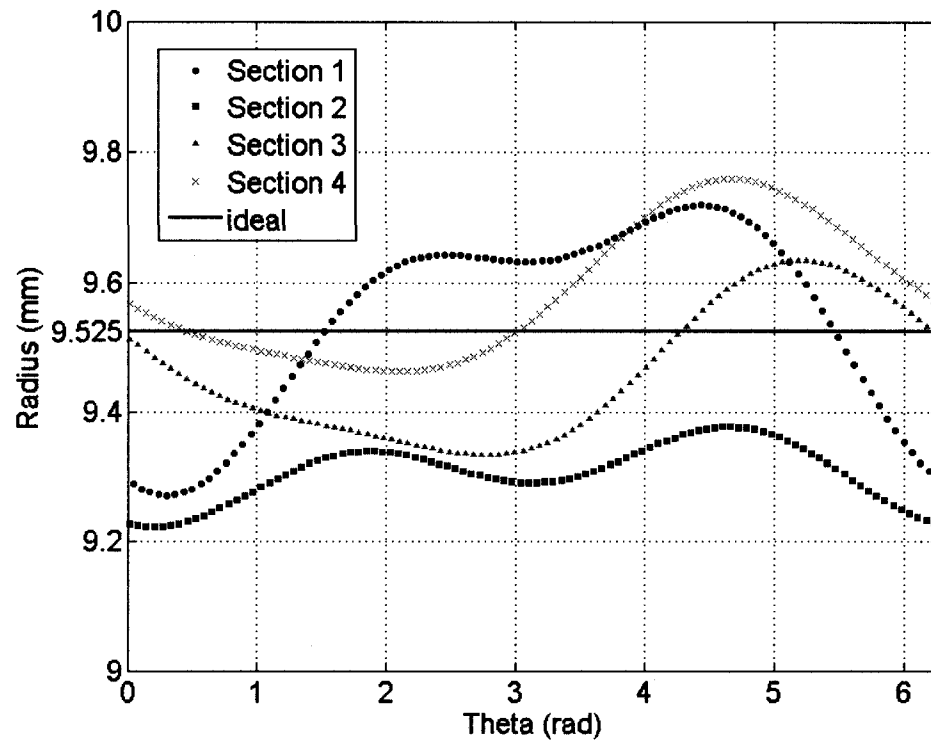
Figure 8.5: Radius profiles of four planar sections of the test cylinder captured with Prototype II. Sections are translated to mean centroid before calculating radius.

to note because with improvements in the overall mechanical design, scan time can be reduced significantly; it is currently limited by the mechanical movements of the apparatus, not acquisition speed.

Total estimated cost of the system (camera, laser, image acquisition device, motors and mechanical setup) is about \$3,000 CAD (~\$2,800 USD [1]). The cost of commercial software used (Matlab and Labview) has not been included. The use of Meshlab for surface mesh reconstruction is free.

---

[1]at an exchange rate of 1 CAD = 0.936867 USD as of August 8th, 2008
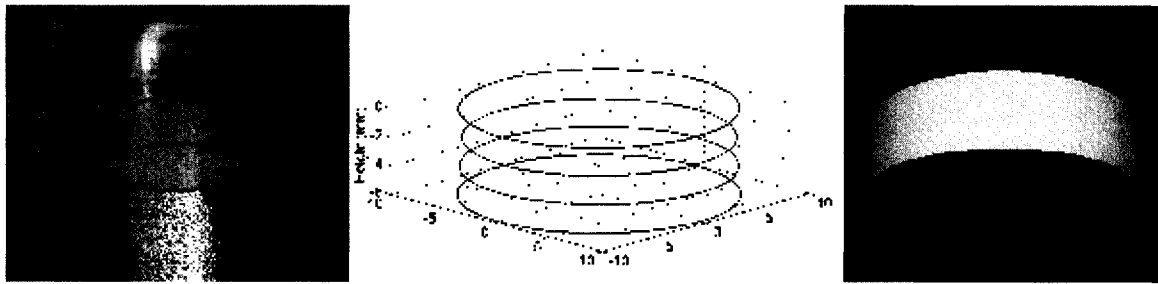
Figure 8.6: Cylinder test object.
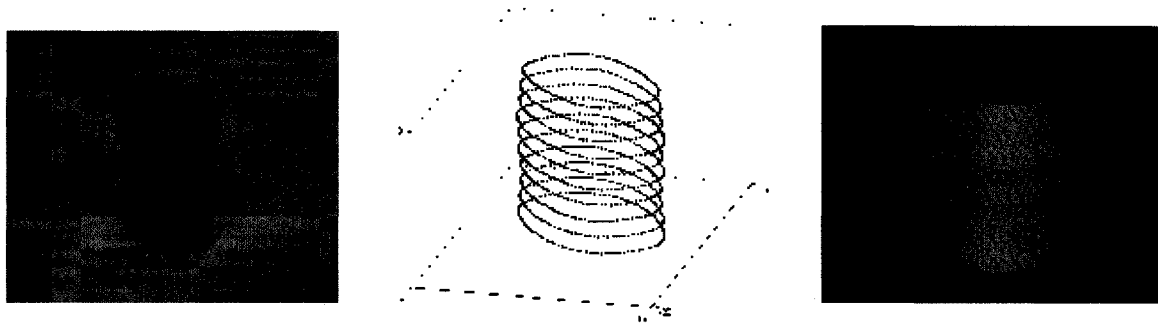
Figure 8.7: Oval cap object.

8.3 Discussion

8.3.1 Prototype I

A discussion of the test cylinder results using Prototype I was presented in Chapter 6 but will be summarized here for convenience. As shown in Figure 8.1 (a), the rectified segments for the given planar section of the test cylinder produce a fairly good result, before and after a curve was fit to the data with the EFD technique. The percent error between the radius of the reconstructed test object and the actual radius of the object is about 1 to 2%. Errors do not exceed 0.25 mm. The subsequent EFD contour for this first layer shows a close fitting through the points using only 1 harmonic. There is however, observable noise in the error plot of Figure 8.1 (a), comprised of both a random and systemic component. Possible sources identified in Chapter 6 were: noise inherent in the measurement system and noise in the resulting
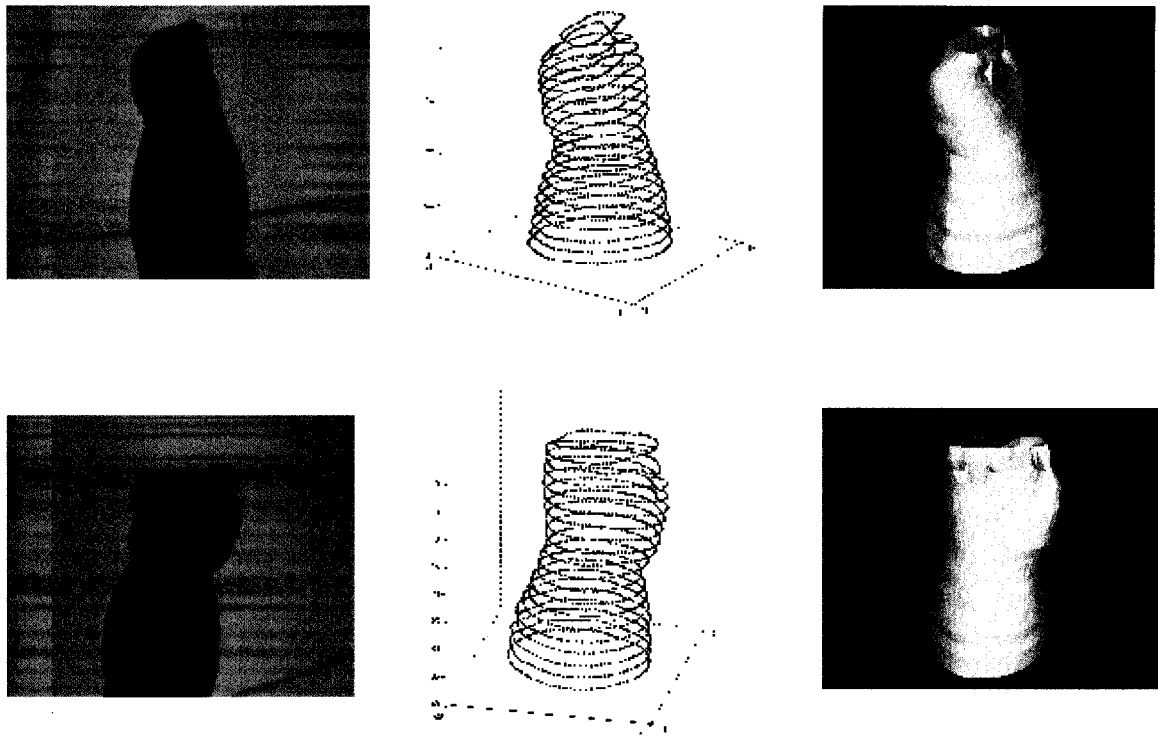
Figure 8.8: Miniature arm model object.

point coordinates obtained from the line thinning algorithm (producing the random component) and; inaccuracies associated with manual manipulation of the object (producing the systemic component). Systemic noise is observed in Figure 8.1 (a) by the obvious jumps in the radii obtained where segments were fused together and a general downwards trend in the radii for these different segments. The analytical calibration procedure used may also be a source of systemic error. The random noise is clearly attenuated in the EFD fit. After obtaining these results it was anticipated that the EFD fit would have an even greater effect with the systemic noise sources addressed; this was supported with simulated data consisting of only random noise. Overall, visual inspection of the reconstructed planar sections of the test cylinder (Figure 8.2 (a)) reveals that Prototype I produced a planar section model, which is reasonably complete in desired external detail.

8.3.2 Prototype II

The most prominent sources of errors for Prototype I were procedural in nature and we attempted to address them with the second version of the prototype in which the camera and laser diode head are automatically incremented vertically on a platform, the object is rotated automatically on a turntable, and more robust algorithms are used for the identification of point coordinates within the laser trace images and for the calibration procedure.

The recreated results for Prototype II (Figure 8.1 (b)) reveal a couple of improvements over Prototype I. Firstly, the systemic noise that was observed where overlapping segments had been averaged (i.e., the steep jumps in the radii where segments were fused together and the general downwards trend in the radii for these different segments) has been attenuated, particularly in the area between 0 and 4 radians. Automatic rotation of the object on a turntable has helped achieve better initial alignment between segments and improved the effect of averaging overlaps. Secondly, although the magnitude of random noise has not changed significantly from Prototype I to Prototype II, the EFD fit does show slightly less deviation from the ideal radius of 9.525 mm, with errors not exceeding 0.125 mm for this section (a more quantitative discussion of error for Prototype II will be discussed subsequently as there is some variation from section to section). Comparing the EFD planar section models in Figure 8.2, there are no observable differences between the results from the two prototypes.

Figure 8.3 shows that although each radius profile is quite close to the ideal value, there are some slight issues of scale across sections, particularly for Section #2 which has the highest RMS error ($\sim$ 0.2 mm) of all sections captured (see Table 8.3). The radius in this case is consistently too small. Possible sources for this could be due to the apparatus setup. Although the camera and laser are fixed to a platform, and

ideally fixed relative to each other, slight deviations in the horizontal alignment of the laser plane with respect to the xy-plane would change the laser profile on the object, and would thus propagate through the data processing as an error in overall scale. Instability of the platform could be the cause of the scaling issues. The RMS error for the other sections ranges from about 0.067 to 0.072 mm. This is considered to be quite good considering the low-cost components which have been used for this setup.

Figure 8.4 shows that the process used to align segments about the axis of rotation (discussed in Section 6.2) produces very accurate vertical alignment of planar sections. In this case, the cylinder object was not exactly centered on the turntable (i.e., not centered at $(0,0)$), resulting in the centers of the sections themselves being slightly offset in the negative x and y directions. Nevertheless, sections are aligned precisely about their true location in 3D space, with deviations from the mean centroid not exceeding 0.15 mm ($\sigma = 0.1$).

Finally, Figure 8.5 demonstrates overall system accuracy. Radius profiles calculated from the mean centroid show slightly higher amplitudes meaning a wider range of error values for each section. This is also reflected in the RMS error calculated for this scenario (see Table 8.3) where error is increased for all sections; Section 2 still shows the highest inaccuracy. Despite these results, error is still considered to be quite low, with RMS error for all sections below 0.23 mm.

Although no exact value of accuracy can be provided for the other objects scanned, visually, the EFD planar section models and their respective 3D models (Figures 8.7 and 8.8) are reasonably complete in external detail. There still exists slight variation in scale across sections, as shown by the bumpiness in the rendered 3D models. However, recall that these objects are already quite small to begin with therefore these errors are still within an acceptable tolerance for this system. The miniature arm model lacks detail in the hand region. This is mostly attributed to the

vertical resolution of the scan (2 mm increments), resulting in triangulation which spans across gaps existing on the real model. This also accounts for the flat surface on top; the next highest planar section image contained no light reflections. The system does have the capability of scanning with finer vertical resolution, however this has not been tested. Due to imprecision in the motor used to rotate the objects, it should be noted that the segments and sections were found to be under-rotated when using the regular data processing scheme (i.e., rotating segments by 45° increments). This was not an issue with the cylinder test object, but became apparent in the non-symmetrical objects, which turned out skewed due to rotation undershooting. Each 45° rotation of the turntable undershot by about 0.2°. Although minute, this error propagated through the processing. Turntable overshoots had to be accounted for in the data processing when segments were rotated to their respective orientations about the axis of rotation. This would not be required with motors that could guarantee better rotation precision. The motors used could guarantee accuracy up to ±1°. Error in vertical increments are not considered an issue because motor rotations are smaller in this case.

Despite these limitations, reconstruction for these test objects is considered quite good. Accuracy for the cylinder is excellent; a more detailed discussion of how these results fare against other systems is provided in the subsequent section. The other objects appear to match reasonably closely to the original objects. Results show that the number of harmonics chosen to accurately fit the data was sufficient, but this is certainly an area for future work. These results also show that the point cloud generated by the system can be successfully processed with open source software and would not necessarily have to be imported into expensive point cloud processing applications in order to create a 3D model.

Comparison of test cylinder results to literature

Since the cylinder results are the only results for which we can estimate system accuracy, these results are used to compare to the literature. We have chosen to compare to 5 different systems. From the research field, results are compared to the scanner being developed by Tognola *et al.* [4], because they also have the goal of scanning anatomical parts, and the scanner by Chen and Kak [22], because they use a linear calibration technique as we have done. From the grouping of commercial scanners presented in Chapter 2 we have chosen to compare to the Creaform ERGOScan, because it is a laser scanner developed specifically for biomedical purposes; the NextEngine Desktop 3D Scanner, because it is the most economical scanner found to date; and the Minolta Vivid 700 scanner, because it was used by Kusnoto and Evans [24] in a study which assessed the scanner's accuracy in scanning objects which are similar, dimensionally, to those scanned in these experiments, including a cylinder. Table 8.5 gives a brief description of each system, the item(s) scanned (if applicable), the accuracy (as given by either a study or the scanner company's website), the cost, and the approximate scanning time.

Since accuracy is measured and quoted in different ways for each scanner, and the objects scanned are of different sizes from different distances, these values are meant to just give an idea of the general range of measured errors reported. Similarly, we recognize that comparing the costs of systems still in the research stages (e.g., ours and that of [4]) to those of commercial systems may not be entirely valid. However, the cost estimate of our system is based on off-the-shelf component prices, which already include a profit margin, and could potentially be much less if one was to consider that a commercial product would not actually be manufactured with the components in their current form. Our intention is to demonstrate what is possible when low cost components are used, and to show how this fares against other systems

Table 8.5: Comparison of Cylinder Results for Prototype II with Existing Comparable Systems.

3D Scanning System	Description	Object(s) scanned	Given Accuracy/Error (mm)	Scan Time	Cost (USD)
Prototype II	Laser light-sectioning system with object on motorized turntable	cylinder, 4 sections (r=9.525mm)	RMS error: 0.135mm	2mins 40s	$2,800
Tognola et al. [4]	MaxReflex motion capture system. Laser is manually swept over object.	-sphere (s) (r=11mm) -silicon cast hearing aid shell (c) ($18 \times 14 \times 22$mm^3) -heart model (h) ($70 \times 104 \times 99$ mm^3)	RMS error for (s): 0.074mm	(s) - unknown (c) - 90s (h) - 10mins	~$50,000-$100,000
Chen and Kak [22]	Light-stripe robot vision system using linear calibration techniques	block ($143.8 \times 143.8 \times 152.4mm^3$)	<1.016mm (horiz) <3.556mm (vert)	1 min (12 stripes from 4 locations)	unknown
Creaform ERGOScan [37]	Handheld laser scanner	n/a	accurate up to 0.5mm	"quick data acquisition"	$40,000
NextEngine Desktop 3D Scanner [34]	Multistripe Laser technology	n/a	±0.127mm (macro); ±0.381mm (wide)	2 mins per scan of each facet	$3,500 - $5,000
Minolta Vivid 700 [29] (data as reported by [24])	Laser-light sectioning scanner	cylinder (c) ($r = 46mm$) dental model (d)	(c) - 0.3mm ±0.3mm (horiz); 0.5mm ±0.1mm (vert) (d) - molar width: 0.2mm ±0.1mm; palatal vault depth: 0.7mm ±0.2mm	0.6s per view	$60,000

which are both comparable in cost, and possibly orders of magnitude more expensive.

In general, results from Prototype II for the test cylinder compare favorably with these systems, in terms of accuracy and cost, however it did not fare as well in terms of scan time. The most accurate of these systems was developed by Tognola *et al.* [4]. Their use of a Qualysis (Gothenburg, Sweden) motion capture system (two CCD cameras and a real-time video processor) produced a reconstruction of a 11.33 mm radius sphere. The RMS error calculated between the real sphere and the reconstructed surface was 0.074 mm. They also scanned 3D models of anatomical objects - a silicon cast hearing aid shell and a heart model - but no estimation of error

are given due to the complexity of the shapes. In terms of scan time, our system does not compare as well with their more manual technique; the hearing aid shell and heart model took 90 s and 10 mins to scan, respectively. Dimensionally, the hearing aid shell is similar to the four sections of the cylinder object and the cap object, which had scan times of 2 mins 40 s and 6 mins 40 s, respectively.

Chen and Kak's [22] system is a structured light-stripe unit on the end of a robot consisting of an infrared light projector and a CCD camera. They used a linear calibration technique for finding the mapping between imaged and real world points, and tested the system by scanning a test block. The block dimensions as given by the scanner had errors of <1.016 mm and <3.556 mm for the width and height respectively, which is an order of magnitude greater than the accuracy achieved with Prototype II. It should be noted that the object scanned here was much larger and was imaged from a further distance (~ 350 mm) than the test cylinder in our experiment. Scan time was about 1 min to capture 12 laser traces, but the robot head was moved to four different locations during the scan.

Comparing with the commercial scanners in Table 8.5, Prototype II accuracy is better than the ERGOScan and comparable to the NextEngine scanner. However, the ERGOScan system requires a standoff distance of about 300mm, which is greater than the scanning distance for our scanner (~ 100 mm). Cost-wise, the estimated cost of components for Prototype II is considerably less expensive than the ERGOScan system ($40,000 USD), and comparable to NextEngine's 3D Desktop Scanner ($3,500 - $5,000 USD). No examples of scan times are available for these systems since information for these scanners did not come from a specific study, however the NextEngine website [34] states that a scan for one view takes 2 mins. Considering that multiple views would be required to capture an entire 3D model, it would probably take about 16 mins to capture eight views about the vertical axis (not including the time required to rotate the object and/or move the scanner), which our scanner did not exceed for

any of the objects scanned. Scan time for the NextEngine scanner would not be limited by object size, whereas each additional horizontal section for our scanner takes an extra 40 s. However, recall that scan time is currently limited by the very crude mechanical setup and not the image acquisition time, hence the potential exists to exceed the performance of this commercial scanner in terms of scan time with improvements in the mechanical setup. Finally, Kusnoto and Evans [24] assessed the accuracy of generating 3D models of both a dental model and a plaster facial model using the Vivid700 3D surface laser scanner (Konica Minolta [29]). They first captured the height and width for a test cylinder object; measurements were accurate to 0.5 mm ($\pm$ 0.1 mm) in the vertical dimension and 0.3 mm ($\pm$ 0.3 mm) in the horizontal dimension. Measurements taken from the dental model (the intermolar width and palatal vault depth) were shown to be accurate to 0.2 mm ($\pm$ 0.1 mm) and 0.7 mm ($\pm$ 0.2 mm) respectively [24]. Prototype II fares well against all of these results. The Vivid 700 is quite expensive ($60,000 USD) but can offer scan times in the order of milliseconds, a huge advantage when scanning human subjects.

Overall, Prototype II is comparable to these systems in many respects. Scan time can certainly be improved upon, as our scan time matches most closely the system with the most manual process of acquiring data (Tognola et $al.$ [4]). The majority of the scan time is currently due to limitations in the mechanical apparatus and not image acquisition time. Therefore scan time can be reduced considerably with improvements in the mechanical setup, which has been recommended for future work.

8.4 Summary

After achieving sub-mm accuracy with Prototype I, where the majority of the errors are the result of manual manipulation of the device, there was good reason to believe that an acceptable level of accuracy could be achieved with future iterations of the

design. Improvements implemented for Prototype II included automatic manipulation of the camera/laser platform and turntable, and more robust algorithms for the identification of point coordinates within the laser trace images and for the calibration procedure. Accuracy improved for Prototype II, where RMS error for the radius of a test cylinder object was 0.135 mm across the four planar sections captured. These results are considered to be well within the realm of acceptability given the low cost components used and limitations of the mechanical setup, especially when compared to existing systems. The accuracy achieved compares well with laser light-sectioning systems in the literature as well as some commercial systems. Currently available commercial 3D scanning systems employing the laser light-sectioning technique have been found to cost anywhere from $3,500 to $350,000 USD; cost is often a major roadblock in many healthcare institutions. The overall cost of this system's components (about $2,800) is below that of the most economical system found on the market.

Current limitations of the system lie mostly with the mechanical setup, and include servo motor rotation accuracy, structural stability, overall scan time, and the general need to rotate the object rather than rotate the camera/laser about the object. The volume of objects which can be scanned is currently quite small, a limitation which can be rectified with a different zoom setting on the camera. Also, although the collection of raw data for objects with discontinuous profiles (i.e., the finger area of an open hand or an object with a hole through it) is possible with the current system (because images are captured from many different views), additional data processing would be required to represent such surfaces with EFDs. As with all systems employing the laser light-sectioning technique, both the camera and the laser require a direct line of sight to the object's surface in order to capture surface data. This poses a limitation when it comes to objects with concavities or hollow interiors. Surface reflectivity was also an issue for the arm model. Improving these aspects of the current design are recommended topics for future work.

As the system exists now, potential applications would be similar to those presented by Kusnoto and Evans [24] or Tognola *et al.* [4], where inanimate, but still anatomical-like objects, can be placed on a turntable and scanned, such as a hearing aid shell or dental model. This was demonstrated by the arm model experiment, which had anatomical-like features.

Overall, by including more complexity in software algorithm development, we were able to use simple, inexpensive, off-the-shelf components in a straightforward setup avoiding complex optical schemes. Specifically, the coplanar calibration process chosen avoids the need to explicitly calculate extrinsic or intrinsic camera parameters or precise component orientations. Imperfections in alignment will be quantified and compensated for by the transformation matrix H found through the calibration procedure. The calibration process itself is semi-automatic; the software isolates grid points within the calibration grid image and the user is required to select the desired points to be used in the DLT LS algorithm. The whole process is very simple so that any user can perform the calibration.

Automation of the image acquisition process is also an important feature of the design, to help reduce the need for user intervention. The Labview interface developed facilitates image acquisition and combines image capture and control of the scanner's turntable and platform into one easy-to-use interface. The straightforward data collection setup avoids the complex range image registrations problem, because the data collected is easily registered into the proper coordinate system based on object rotations and camera/laser increments.

Data processing was another area where including more complexity in algorithm development contributed to the ability to use simple, inexpensive, off-the-shelf components in a straightforward setup. It was observed through reviewing current available 3D scanning technology that with increased accuracy comes increased system complexity and cost. But since noise in measurement data is inevitable in many

biomedical scenarios, certain levels of accuracy would be impossible to achieve. A key feature of this system is robust data processing specifically focused on addressing noise in the measurement data; namely, Elliptical Fourier Descriptors (EFDs), which are particularly suited for modeling biological objects. Additionally, the averaging of overlapping segment data also contributes to noise reduction.

Chapter 9

Conclusions and Recommendations

Three-dimensional laser imaging is quickly becoming a valuable and efficient imaging modality in healthcare. However, there is an increasing need for cost effective 3D scanning equipment in the biomedical field, as most of the currently available state of the art is not specifically designed for this purpose, or is prohibitively expensive. Our main objective for this thesis was to develop a simple, low cost 3D laser light-sectioning scanner for biomedical purposes. We wanted to explore multiple design options at different stages of the scanning process, specifically in the areas of calibration and data manipulation, and validate the chosen techniques experimentally by implementing prototype systems. Section 9.1 summarizes the major conclusions drawn from this thesis work and Section 9.2 presents recommendations for areas of future work.

9.1 Conclusions

We conclude that the overall system design for the 3D laser light-sectioning scanner presented in this thesis is viable; two prototypes validate aspects of the calibration, image acquisition, image processing, data manipulation, and object reconstruction capacity of the chosen design. Accuracy for Prototype II is quite good, with average

RMS error for the radius of a test cylinder object only 0.135 mm across the four planar sections captured. These results are considered to be well within the realm of acceptability given the low cost components and the mechanical setup used, especially when compared to existing systems. The accuracy achieved compares well with laser light-sectioning systems in the literature as well as some commercial systems. Although many studies have concluded that certain available commercial scanners are useful for biomedical applications, we have shown comparable accuracy at a reduced cost.

In terms of image calibration for the system, we conclude that the Direct Linear Transform Least Squares (DLT LS) algorithm for computing the transformation matrix H is better than the Analytical method when input data are noisy. Based on an analysis comparing these two methods, we have shown that the Analytical method and the DLT LS method are comparable when the same number of points are used, but that the DLT LS method improves as more points are added, thus outperforming the Analytical method. Since noise in the input data is an anticipated consequence of the more simplistic mechanical setup employed for this system, the DLT LS method is the recommended technique of choice for this scanner.

Finally, we conclude that the use of Elliptical Fourier Descriptors is appropriate and beneficial for fitting a smooth contour through the segment data collected from this laser light-sectioning scanner. Analyses on both simulated and experimental data have shown the usefulness of this curve approximation technique for both reducing noise in the measurement data and for representing planar sections as seamless contours.

9.2 Recommendations for Future Work

Due to the many diverse aspects of this entire research project, there are several different avenues for potential future development. Directions for future work have been categorized according to the stages of the system flow diagram.

Calibration

The major improvement which could be made in the area of calibration would be to explore non-linear algorithms for characterizing the mapping between image coordinates and real world coordinates. It was shown in this thesis that projective geometry can be used for a coplanar 2D to 2D mapping, and that explicit computation of the camera and pose parameters is not necessary for this calibration scenario. However, exploring a non-linear approach to the coplanar calibration case would allow for the compensation of non-linear distortions in the images, such as radial lens distortion. The added benefit of this compensation would have to be weighed against the increased complexity in the processing required to implement such an algorithm.

Image Acquisition

In general, a more robust mechanical setup is required to capture even more precise laser trace images. This would involve incorporating servo motors with better precision so that over- and/or under-rotation does not have to be compensated for in the data processing. Overall stability of the structure can be improved with a more rigid structure. Testing the system's accuracy using a finer vertical resolution (i.e., < 2 mm) is also recommended, especially if motors with finer and more accurate rotation control are incorporated into the system. Results from Prototype II have shown that the proposed design is applicable to small, inanimate but anatomical-like objects. However, an important feature for a biomedical scanner targeted at scanning

human subjects would be the ability of the camera and laser system to rotate about the part of interest with the required number of degrees of freedom. Therefore, the next stage would be to develop the required mechanics for the added flexibility of not having to use a turntable. This could be accomplished by mounting the camera and laser to a robotic arm-like structure, which could accurately track its location in 3D space. The data processing would be identical, as capturing images about the object while it stays stationary is the equivalent problem to the scenario of Prototype II. Moreover, a black and white camera has been used in this thesis; exploring the use of a colour camera could lead to more flexibility in the choice of light source (i.e., coloured light stripes) as well as open up possibilities for incorporating colour and texture information from the original object into the final 3D model. Scan time is also an area for improvement. Scan time can be reduced with an improved mechanical setup.

Image Processing

The images captured for the experiments presented here had low ambient lighting conditions. Although a background elimination technique was applied to all images, further work in this area of image processing could lead to more flexibility in the range of lighting conditions possible for the surrounding environment during image capture. For example, taking a difference image (laser on minus laser off) could isolate the laser even better. This would require a trigger to turn the laser on and off during the scan process, which is not currently a feature of the prototype.

Data Manipulation

The technique of using EFDs for fitting a smooth contour through data segments is currently only implemented for planar sections in the horizontal xy-plane. Further work could be done to explore their use for smoothing data in the vertical direction,

as demonstrated in [86] where EFD coefficients are interpolated across horizontal sections in order to provide surface contour data at any value of z. Also, more work is required to develop a preprocessing technique that would better identify an appropriate number of harmonics to use for a given set of data segments. The current system allows for the collection of raw data for objects with discontinuous profiles (i.e., the finger area of an open hand or an object with a hole through it) however, additional data processing would be required to fit EFDs to individual elements within the planar sections of such objects rather than one EFD contour for the entire planar section. For example, the raw data for a planar section of an open hand would be a series of circular outlines representing individual fingers. An additional step would have to be incorporated whereby these individual regions could be identified as individual objects and fit with separate EFD approximations. Finally, while EFDs performed well and seem like an appropriate solution because they are particularly suitable for representing biological objects, other methods could also be investigated in more detail, such as polynomials or splines. A comparison should be done to better understand the advantages and disadvantages of these different methods, in terms of accuracy of the fit, data processing time, etc. Also, it would be worthwhile to investigate methods which could account for cases where discontinuities in the laser trace exist (e.g. sharp edges).

Object Reconstruction

The choice of using open source 3D modeling software for object reconstruction was convenient and economical for this work. Other possibilities could be explored which might produce 3D models with greater fidelity.

List of References

[1] F. Bernardini and H. Rushmeier, "The 3D model acquisition pipeline," *Computer Graphics Forum*, vol. 21, no. 2, pp. 149–172, 2002.

[2] R. Bajcsy and R. Pito, "Data acquisition and representation of mechanical parts and interfaces to manufacturing devices," in *Proc. Int'l Conf. Recent Advances in 3-D Digital Imaging and Modeling*, pp. 2–9, May 1997.

[3] C. Istook and S. Hwang, "3D body scanning systems with application to the apparel industry," *J. Fashion Marketing and Management*, vol. 5, no. 2, pp. 120–132, 2001.

[4] G. Tognola, M. Parazzini, P. Ravazzani, F. Grandori, and C. Svelto, "3D acquisition and quantitative measurements of anatomical parts by optical scanning and image reconstruction from unorganized range data," *IEEE Trans. Instrum. Meas*, vol. 52, pp. 1665–1673, October 2003.

[5] M. Hajeer, D. Millett, A. Ayoub, and J. Siebert, "Applications of 3D imaging in orthodontics: Part i," *J. of Orthodontics*, vol. 31, pp. 62–70, 2004.

[6] R. Hennessy, S. McLearie, and A. Kinsella, "Facial surface analysis by 3D laser scanning and geometric morphometrics in relation to sexual dimorphism in cerebral-craniofacial morphogenesis and cognitive function," *J. of Anatomy*, vol. 207, pp. 283–295, 2005.

[7] N. Trinh, J. Lester, B. Fleming, G. Tung, and B. Kimia, "Accurate measurement of cartilage morphology using a 3D laser scanner," in *Proc. of 2nd Int'l ECCV Workshop on Computer Vision Approaches to Medical Image Analysis, CVAMIA '06*, (Graz, Austria), pp. 37–48, May 2006.

[8] Z. Azouz, C. Shu, R. Lepage, and M. Rioux, "Extracting main modes of human body shape variation from 3D anthropometric data," in *Proc. 5th Int'l Conf.*

3-D Digital Imaging and Modeling, 3DIM '05, (Ottawa, Canada), pp. 335–342, June 2005.

[9] K. Schwenzer-Zimmerer, J. Haberstok, L. Kovacs, B. Boerner, N. Schwenzer, P. Juergens, H.-F. Zeilhofer, and C. Holberg, "3D surface measurement for medical application-technical comparison of two established industrial surface scanning systems," *J.Med. Sys.*, vol. 32, pp. 59–64, 2008.

[10] "3DMD." 3Q Technologies, *http://www.3dmd.com*, 2006.

[11] J. Jalkio, R. Kim, and S. Case, "Three dimensional inspection using multistripe structured light," *Optical Engineering*, vol. 24, pp. 966–974, 1985.

[12] G. Taylor and L. Kleeman, "Stereoscopic light stripe scanning: Interference rejection, error minimization and calibration," *Int'l J. Robotics Reasearch*, vol. 23, pp. 1141–1156, December 2004.

[13] M. Petrov, A. Talapov, T. Robertson, A. Lebedev, A. Zhilyaev, and L. Polonskiy, "Optical 3D digitizers: Bringing life to the virtual world," *IEEE Computer Graphics and Applications*, vol. 18, pp. 28–37, May/June 1998.

[14] J. Forest and J. Salvi, "A review of laser scanning three-dimensional digitizers," in *Proc. Int'l Conf. Intelligent Robots and Systems*, (Lausanne, Switzerland), pp. 73–78, October 2002.

[15] T. Terauchi, Y. Oue, and K. Fujimura, "A flexible 3D modeling system based on combining shape-from-silhouette with light-sectioning algorithm," *Proc. 5th Int'l Conf. 3D Digital Imaging and Modeling, 3DIM'05*, pp. 196–203, 2005.

[16] B. Bradley, M. Hayes, and A. Chan, "A simple, low cost, 3D scanning system using the laser light-sectioning method," in *Proc. IEEE Int'l Inst. Meas. Tech. Conf., I2MTC '08*, (Victoria, Canada), pp. 299–304, May 2008.

[17] B. Bradley, A. Chan, and M. Hayes, "Calibration of a simple, low cost, 3D laser light-sectioning scanner system for biomedical purposes," in *Proc. 31st Annual Canadian Medical and Biological Engineering Conference, CMBEC31*, (Montréal, Canada), June 2008.

[18] B. Bradley, A. Chan, and M. Hayes, "A 3D scanning system for biomedical purposes using the laser light-sectioning method and elliptical fourier descriptors," in *Proc. IEEE Int'l Workshop on Medical Measurements and Applications, MeMeA '08*, (Ottawa, Canada), pp. 101–105, May 2008.

[19] B. Curless, "From range scans to 3D models," *Computer Graphics*, vol. 33, pp. 38–41, November 1999.

[20] M. Rana, H. Setan, Z. Majid, and A. Chong, "Computer assisted surgical planner for craniofacial reconstruction: Imaging techniques," in *Proc. IEEE Geometric Modeling and Imaging Conf., GMAI'06*, pp. 215–220, July 2006.

[21] R. Siegwart, "3D triangulation-based range finder for an inspection robot." Swiss Federal Institute of Technology, Hamburg, Germany, *http://www.asl.ethz.ch/education/proj_apps/pdfs/doku31.pdf*, 2004. Accessed: June 25, 2008.

[22] C. Chen and A. Kak, "Modeling and calibration of a structured light scanner for 3D robot vision," in *Proc. IEEE Int'l Conf. Robotics and Automation, ICRA '87*, vol. 4, pp. 807–815, Mar 1987.

[23] I. Reid, "Projective calibration of a laser-stripe range finder," *Image and Vision Computing*, vol. 14, pp. 659–666, 1996.

[24] B. Kusnoto and C. A. Evans, "Reliability of a 3D surface laser scanner for orthodontic applications," *Am J Orthod Dentofacial Orthop.*, vol. 122, pp. 342–348, April 2002.

[25] L. Kovacs, A. Zimmermann, G. Brockmann, H. Baurecht, K. Schwenzer-Zimmerer, N. Papadopulos, M. Papadopoulos, R. Sader, E. Biemer, and H. Zeil-hofer, "Accuracy and precision of the three-dimensional assessment of the facial surface using a 3D laser scanner," *IEEE Trans. Medical Imaging*, vol. 25, pp. 742–754, June 2006.

[26] L. Kovacs, A. Zimmermann, G. Brockmann, M. Ghring, H. Baurecht, N. Papadopulos, K. Schwenzer-Zimmerer, R. Sader, E. Biemer, and H. Zeilhofer, "Three-dimensional recording of the human face with a 3D laser scanner," *Journal of Plastic Reconstructive and Aesthetic Surgery*, vol. 59, pp. 1193–1202, November 2006.

[27] C. Boehnen and P. Flynn, "Accuracy of 3D scanning technologies in a face scanning scenario," in *Proc. 5th Int'l Conf. 3D Digital Imaging and Modeling, 3DIM'05*, pp. 310 – 317, June 2005.

[28] J. Thilmany, "Working backward," *Mechanical Engineering*, pp. 30–34, July 2008.

[29] "Konica Minolta Inc.." *http://www.konicaminolta.com*, 2007-2008.

[30] "Cyberware Rapid 3D Scanning." *http://www.cyberware.com*, 1999.

[31] "Metris." *http://www.metris.com*, 2006.

[32] "Polhemus." *http://www.polhemus.com*, 2006.

[33] "Shapegrabber." *http://www.shapegrabber.com*, 2008.

[34] "NextEngine." *https://www.nextengine.com*, 2008.

[35] "3D Digital Corp." *http://www.3ddigitalcorp.com*, 2008.

[36] "Vitronic." *http://www.vitronic.de/en*, 2008.

[37] "Creaform." *http://www.creaform3d.com*, 2002-2008.

[38] "Hanger Orthopedic Group Inc." *http://www.hanger.com*, 2008.

[39] G. Tognola, M. Parazzini, C. Svelto, P. Ravazzani, and F. Grandori, "A fast and reliable system for 3D surface acquisition and reconstruction," *Image and Vision Computing*, vol. 21, pp. 295–305, 2003.

[40] G. Tognola, M. Parazzini, P. Ravazzani, F. Grandori, and C. Svelto, "Simple 3D laser scanner for anatomical parts and image reconstruction from unorganized range data," in *Proc. IEEE Inst. Meas. Tech. Conf., IMTC '02*, (Anchorage, Alaska), pp. 171–173, May 2002.

[41] G. Tognola, M. Parazzini, P. Ravazzani, C. Svelto, and F. Grandori, "3D reconstruction of anatomical surfaces from unorganized range data," in *Proc. 23rd Ann. EMBS Int'l Conf.*, (Istanbul, Turkey), pp. 2534–2536, October 2001.

[42] A. Lichtschnitt, "Laser light section: a key feature in 3D laser measurement technique." Schafter and Kirchhoff, Hamburg, Germany, *http://www.sukhamburg.de/dl/lightsect-e.pdf*, 2004. Accessed: June 29, 2007.

[43] J. Salvi, X. Armangué, and J. Batlle, "A comparative review of camera calibrating methods with accuracy evaluation," *Pattern Recognition*, vol. 35, pp. 1617–1635, 2002.

[44] C. Chatterjee and V. Roychowdhury, "Algorithms for coplanar camera calibration," *Machine Vision and Applications*, vol. 12, pp. 84–97, 2000.

[45] R. Hartley and A. Zisserman, *Multiple View Geometrey in Computer Vision.* Cambridge, UK: Cambridge University Press, 2000.

[46] M. Hayes, M. Leitner, P. O'Leary, R. Ofner, and C. Sallinger, "An integrated optical-robotic measurement system," in *Proc. 8th Canadian Congress of Applied Mechanics, CANCAM '01*, (St. John's, Canada), pp. 287–288, 2001.

[47] H. Kikuchi, Y. Maeda, M. Sugi, and T. Arai, "Automated calibration for assembly device installation based on plug and produce concept," in *Initiatives of Precision Engineering at the Beginning of the Millennium: 10th International Conference on Precision Engineering, ICPE*, (Yokohama, Japan), Kluwer Academic Publishers, July 2001.

[48] A. Wong and J. Orchard, "Efficient and robust non-rigid least-squares rectification of medical images," in *Proc. 2006 International Conference on Image Processing, Computer Vision, and Pattern Recognition,, IPCV '06*, (Las Vegas, Nevada), pp. 67–73, June 2006.

[49] R. Ofner, P. O'Leary, and M. Leitner, "A collection of algorithms for the determination of construction points in the measurement of 3D geometries via light-sectioning," in *2nd Workshop on European Scientific and Industrial Collaboration promoting: Advanced Technologies in Manufacturing, WESIC '99*, (Newport, South Wales, UK), pp. 505–512, 1999.

[50] L. Lam, S.-W. Lee, and C. Suen, "Thinning methodologies - a comprehensive survey," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, pp. 869–885, September 1992.

[51] Z. Guo and R. Hall, "Parallel thinning with two-subiteration algorithms," *Comm. ACM*, vol. 32, no. 3, pp. 359–373, 1989.

[52] J. Forest, J. Salvi, E. Cabruja, and C. Pous, "Laser stripe peak detector for 3D scanners - a fir approach," in *Proc. 17th Int'l Conf. Pattern Recognition, ICPR '04*, vol. 3, pp. 646–649, August 2004.

[53] R. Fisher and D. Naidu, "A comparison of algorithms for subpixel peak detection," in *Proc. 1991 British Machine Vision Association Conf*, pp. 217–225, 1991.

[54] K. Pulli, T.Duchamp, H. Hoppet, J. McDonald, L. Shapiro, and W. Stuetzle, "Robust meshes from multiple range maps," in *Proc. Int'l Conf. Recent Advances in 3D Digital Imaging and Modeling*, pp. 205–211, May 1997.

[55] M. Rutishauser, M. Sticker, and M. Trobina, "Merging range images of arbitrarily shaped objects," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR '94*, (Seattle, Washington), pp. 573–580, 1994.

[56] D. Huber and M. Hebert, "Fully automatic registration of multiple 3D data sets," *Image and Vision Computing*, vol. 21, pp. 637–650, 2003.

[57] P. Tang, D. Huber, and B. Akinci, "A comparative analysis of depth-discontinuity and mixed-pixel detection algorithms," in *Proc. 6th Int'l IEEE Conf. 3D Imaging and Modeling, 3DIM '07*, pp. 29–38, August 2007.

[58] M. Soucy and D. Laurendeau, "A general surface approach to the integration of a set of range views," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 17, pp. 344–358, April 1995.

[59] F. Khul and C. Giardina, "Elliptic fourier features of a closed contour," *Computer Graphics and Image Processing*, vol. 18, pp. 236–258, 1982.

[60] R. Bolle and B. Vemuri, "On three-dimensional surface reconstruction methods," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, no. 1, pp. 1–13, 1991.

[61] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Surface reconstruction from unorganized points," in *Proc. Int'l Conf. Computer Graphics and Interactive Techniques, SIGGRAPH '92*, (Chicago, Illinois), pp. 71–78, July 1992.

[62] M.-C. Chang, F. Leymarie, and B. Kimia, "Surface reconstruction from point clouds by transforming the medial scaffold," in *Proc. 6th Int'l IEEE Conf. 3D Imaging and Modeling, 3DIM '07*, pp. 13–20, August 2007.

[63] B. Delaunay, "Sur la sphere vide. A la mémoire de Georges Voronoi," *Bulletin of the Academy of Sciences of the USSR*, vol. 7, pp. 793–800, 1934.

[64] F. Cazals, J. Giesen, and M. Yvinec, *Rapport de recherche 5394: Delaunay triangulation based surface reconstruction: a short survey*. Cedex, France: Institut National de Recherche en Informatique et en Automatique, 2004.

[65] L. DeFloriani, "Surface representations based on triangular grids," *The Visual Computer*, vol. 3, pp. 27–50, February 1987.

[66] L. DeFloriani, B. Falcidieno, and C. Pienovi, "Delaunay-based representation of surfaces over arbitrarily shaped domains," *Computer Vision, Graphics, and Image Processing*, vol. 32, pp. 127–140, 1985.

[67] C. Bajaj, E. Coyle, and K.-N. Lin, "Surface and 3D triangular meshes from planar cross sections," in *5th International Meshing Roundtable*, (Sandia National Laboratories, Livermore, California), pp. 169–178, October 1996.

[68] J. Boissonnat, "Shape reconstruction from planar cross sections," *Computer Vision, Graphics and Image Processing*, vol. 44, pp. 1–29, 1988.

[69] F. Bernardini, J. Mittleman, H. Rushmeier, C. Silva, and G. Taubin, "The ball-pivoting algorithm for surface reconstruction," *IEEE Trans. Visualization and Computer Graphics*, vol. 5, pp. 349–359, October 1999.

[70] L. Piegl, "On nurbs: A survey," *IEEE Computer Graphics and Applications*, vol. 11, no. 1, pp. 55–71, 1991.

[71] G. Farin, "From conics to nurbs: A tutorial and survey," *IEEE Computer Graphics and Applications*, vol. 12, pp. 78–86, September 1992.

[72] G. Champleboux, S. Lavallée, R. Szeliski, and L. Brunie, "From accurate range imaging sensor calibration to accurate model-based 3D object localization," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR '92*, pp. 83–89, June 1992.

[73] "VRmesh." *http://www.vrmesh.com*, 2008. Accessed: July 25, 2008.

[74] "Geomagic." GeoMagic Inc., *http://www.geomagic.com/en*, 2008. Accessed: July 25, 2008.

[75] "Meshlab." *http://meshlab.sourceforge.net*, March 2008. Accessed: May 22, 2008.

[76] "Leica Cyclone." Leica Geosystem, *http://www.leica-geosystems.com*, 2008. Accessed: July 25, 2008.

[77] "Polyworks." Innovmetric Software, *http://www.innovmetric.com*, 2008. Accessed: July 25, 2008.

[78] Matlab, "Matlab documentation (Morphology fundamentals: Dilation and erosion)." The MathWorks, Inc., 2007.

[79] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[80] Z. Wu, L. Wu, and A. Wu, "The robust algorithms for finding the center of an arc," *Computer Vision and Image Understanding*, vol. 62, no. 3, pp. 269–278, 1995.

[81] S. Joseph, "Unbiased least squares fitting of circular arcs," *Graphical Models and Image Processing*, vol. 56, no. 5, pp. 424–432, 1994.

[82] P. O'Leary, *Institute for Automation*. Leoben, Austria: University of Leoben, 2000.

[83] R. Safaee-Rad, K. Smith, B. Benhabib, and I. Tchoukanov, "Application of moment and fourier descriptors to the accurate estimation of elliptical shape parameters," in *Proc. Int'l Conf. Acoustics, Speech and Signal Processing, ICASSP '91*, pp. 2465 – 2468, April 1991.

[84] P. Lestrel, ed., *Fourier descriptors and their applications in biology*. Cambridge, UK: Cambridge University Press, 1997.

[85] M. Schmittbuhl, J.-M. L. Minor, A. Schaaf, and P. Mangin, "The human mandible in lateral view: Elliptical fourier descriptors of the outline and their morphological analysis," *Annals of Anatomy*, vol. 184, pp. 199–207, 2002.

[86] Y. Jeong and R. Radke, "Reslicing axially sampled 3D shapes using elliptic fourier descriptors," *Medical Image Analysis*, vol. 11, pp. 197–206, 2007.

[87] D. Thomas, *University of Melbourne*. Melbourne, Australia: Matlab Central File Exchange, 2006.

[88] J. Proakis and D. Manolakis, *Digital Signal Processing: Principles, Algorithms, and Applications*. Upper Saddle River, NJ: Pearson Prentice Hall, 2007.

[89] J. Murray and W. VanRyper, *Encyclopedia of Graphics File Formats: Second Edition*. Sebastopol, CA: O'Reilly and Associates, pp.946-952, 1996.

[90] "Autodesk." *http://usa.autodesk.com*, 2008. Accessed: July 25, 2008.

[91] "Blender." *http://www.blender.org*, 2008. Accessed: May 22, 2008.

[92] "Lasiris diode laser products instruction manual, version 3." StockerYale, *http://www.stockeryale.com/lasers*, 2007. Accessed: June 29, 2008.

Appendix A

Design Considerations for a Biomedical-specific Scanner

Although many of the studies presented in Chapter 2 have concluded that certain available commercial scanners have proven to be useful for biomedical applications, and the advantages of using laser-light scanners are apparent, there were still many *disadvantages* highlighted in these studies which indicates that not all available scanners address the specific needs of the medical community. Therefore, the following list of design considerations for biomedical-specific 3D scanners has been compiled.

- **Safety and laser light sources**

The major safety issue concerning the use of laser technologies with human subjects is the accidental exposure of the eyes to the laser beam and the risk of causing permanent damage to the retina, especially in growing children [5]. There are certain classes of lasers (i.e., Class II and lower), which are considered eye-safe meaning normal exposure to this type of beam will not cause permanent harm. "The blinking reflex of the eye is fast enough to avoid any damage" [92]. That being said, special attention needs to be given to the surrounding environment; "no mirrors, lenses or other reflecting objects are allowed to be placed in the path of the laser beam, since

there is a danger that this could influence the direction of the laser beam, provoke eye damage or even ignite flammable substances" [9].

When using laser, there is also the risk of skin burns, but this would only occur in the extreme case that a laser be left projected on a human subject for an extended period of time. As long as safety precautions are established when this type of equipment is used in a healthcare setting (ie. automatic shut off safeguards before the threshold of risk is reached), no serious threat should be expected from using this technology. In general, "any system that involves patients must be absolutely innocuous with respect to health" [9].

- **Scanning time, accuracy, and resolution**

In applications where human subjects are to be scanned, fast acquisition speed is important for reducing human body movement artefact in the measurement data. It also allows for many different subjects to be scanned in a short period of time [3]. Many 3D laser scanner systems need only two seconds to capture a range image, but some believe this is even "too long to reliably avoid head movements, especially when dealing with children" [5]. When these errors cannot be avoided, an important design consideration is appropriate data filtering and smoothing techniques. As well, one must consider the required accuracy and resolution for the application. For example, "in order to acquire precise, detailed 3D images of facial structures, in the nasal and upper-lip area, measurement accuracy in the sub-millimeter range is required" [9], whereas other applications such as whole body scanning, may not require such accuracy and resolution.

- **Physical scanner size, and device manipulation**

This is a fairly important consideration and one that would not necessarily be thought about during the initial design of a scanner. Many healthcare institutions

have very limited space available for large cumbersome equipment, especially if the device is not easily moved, or if moving it requires expensive and elaborate re-calibration. Furthermore, "removing the need to shift the scanner...could be advantageous when examining live persons because of a lessened influence of motion artefacts and minimization of shifts in patient position" [13].

- **Data size, type, and integration**

File size is an important issue in many circumstances, not just in healthcare. When designing a 3D scanner, issues of data management, storage, transmission and the intended use of the final 3D model must be considered. With the advent of e-health records on the horizon, and data transfer among intuitions happening increasingly more by electronic means, smaller, more manageable files will be essential if 3D models are to become an integrated part of a person's health care management.

Integration and compatibility with existing medical file formats should also be considered, as there is huge potential to integrate 3D surface information with existing imaging modalities, such as CT, MRI, PET and ultrasound. According to [72] there is a real need is to register all of these 3D images into the same reference system. There is a role for 3D scanning to play in acquiring the skin surface of a patient, so that this reference surface can be registered with the skin surface segmented from another imaging device [72]. For this to be successful, the data from a biomedical 3D scanning system must exist in a form compatible for this registration to occur.

- **Ease of use and training required**

In order to minimize the amount of training required, "the equipment must be reliable and appropriate for use by technically untrained personnel" [9]. It cannot be assumed that the user will have technical knowledge of the equipment. Furthermore, the likelihood of a 3D scanning device being adopted into a specific medical practice will depend on its ease of use.

Appendix B

Background on Projective Geometry

This section provides background information on the basic principles and theory of projective geometry. This background is intended to introduce the reader to these concepts so that the techniques used in Chapter 4 for coplanar scanner image calibration may be well understood. The majority of this explanation is adapted from *Multiple View Geometry in Computer Vision* by Hartley and Zisserman [45].

This discussion presented on projective geometry describes the process of finding the transformation between images of 2D planes. Chapter 4 describes how this discussion of projective geometry and methods can be adapted to the camera calibration problem and thus this 3D scanner.

B.1   The 2D Projective Plane

B.1.1   Homogeneous representation of lines

A line in a plane is represented by an equation such as $ax + by + c = 0$, where different choices of a, b and c give rise to different lines [45]. The vector representation of this line $(a, b, c)^T$, however, is not a one-to-one correspondence; $ax + by + c = 0$ and $(ka)x + (kb)y + (kc) = 0$ are the same, for any non-zero constant, k. "Thus,

the vectors $(a, b, c)^T$ and $k(a, b, c)^T$ represent the same line, and are considered equivalent. An equivalence class of vectors under this equivalence relationship is known as a *homogeneous* vector. The set of equivalence classes of vectors in $\mathbb{R}^3$ forms the *projective space*, $\mathbb{P}^2$" [45].

B.1.2 Homogeneous representation of points

Similarly, points are represented by homogeneous vectors as well. Consider a point $\mathbf{x} = (x, y)^T$ which lies on the line $\mathbf{l} = (a, b, c)^T$; then $ax + by + c = 0$ must be true, which, written as an inner product of vectors would be $(x, y, 1)(a, b, c)^T = 0$. The point $(x, y)^T$ in $\mathbb{R}^2$ is thus represented as a 3-vector by adding the coordinate of 1. For any non-zero k and line $\mathbf{l}$, the equation $(kx, ky, k)\mathbf{l} = 0$ if and only if $(x, y, 1)\mathbf{l} = 0$. Therefore, the set of vectors $(kx, ky, k)^T$ for varying values of k can be considered to represent the point $(x, y)^T$ in $\mathbb{R}^2$. Thus, just as with lines, points are represented by homogeneous vectors. An arbitrary homogeneous vector representation of a point would be of the form $\mathbf{x} = (x_1, x_2, x_3)^T$, which represents the point $(x_1/x_3, x_2/x_3)^T$ in $\mathbb{R}^2$.

B.1.3 Projective transformations

A *projective transformation*, or a homography, is an invertible mapping h from $\mathbb{P}^2$ to itself such that three points x_1, x_2 and x_3 lie on the same line if and only if $h(x_1), h(x_2)$ and $h(x_3)$ do as well [45].

A *projective transformation* is thus defined as: a linear transformation on homogeneous 3-vectors represented by a non-singular 3×3 matrix:

$$
\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix}
\tag{B.1}
$$

or more briefly, $\mathbf{x'} = \mathbf{Hx}$. Note that multiplying the matrix H by an arbitrary non-zero scale factor will not alter the projective transformation. "Consequently we say that H is an *homogeneous matrix* since, as in the homogeneous representation of a point, only the ratio of the matrix elements is significant". There are eight independent ratios amongst the nine elements of H, and thus eight degrees of freedom; therefore the transformation H is completely specified by eight parameters [45].

B.2   Estimation of 2D Projective Transformations

The estimation problem is the "computation of some transformation based on measurements" [45]; that is, given a set of i point correspondences $\mathbf{x}_i \leftrightarrow \mathbf{x}_i'$ between 2 images, compute a 3×3 matrix H such that $\mathbf{x}_i' = \mathbf{Hx}_i$ for each i. Since the transformation H has eight degrees of freedom, it is necessary to specify four point correspondences in order to fully describe H because each point has two degrees of freedom corresponding to the x and y components. This gives rise to the fundamental theorem of 2D projective geometry stated below:

The Fundamental Theorem of Two Dimensional Projectivity:

Given four distinct non-collinear points on a plane and another four distinct non-collinear points on the other plane, there is one and only one projectivity which

carries the first four points respectively into the second four points. [22]

If exactly four correspondences are given, then an exact solution for the matrix H is possible. This situation is referred to as the *Analytical Method* in this thesis. However, since points are measured inexactly (due to noise, for example), then point correspondences may not be fully compatible with *any* projective transformation, if more than four such correspondences are given. In this case, one can find the 'best' transformation, given the data. This is generally done by finding the transformation H that minimizes some cost function [45]. We call one such situation the *Direct Linear Transformation (DLT) Method* in this thesis.

B.2.1 The Analytical method

Beginning from the transformation equation $\mathbf{x}_i' = \mathbf{H}\mathbf{x}_i$, we can re-write the equation so that the point vectors are written in Cartesian homogeneous form, that is, their 3rd coordinate is 1 and an independent homogeneous scale factor ρ is factored out. This leaves:

$$\rho_i \mathbf{q}_i' = H\mathbf{q}_i \tag{B.2}$$

"With the help of the free variable ρ, we are ensured that regardless of the homogeneous coordinates chosen, the above expression for the projectivity solution will always be satisfied" [22].

Expanded, Equation B.2 becomes the following:

$$\rho_i \begin{bmatrix} x_i' \\ y_i' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \tag{B.3}$$

Rearranging the equation to be $\mathbf{H}\mathbf{q}_i - \rho_i \mathbf{q}'_i = 0$ results in:

$$
\begin{bmatrix}
h_{11}x_i + h_{12}y_i + h_{13} - \rho_i x'_i \\[1em]
h_{21}x_i + h_{22}y_i + h_{23} - \rho_i y'_i \\[1em]
h_{31}x_i + h_{32}y_i + h_{33} - \rho_i
\end{bmatrix} = \mathbf{0}
\qquad\qquad \text{(B.4)}
$$

The above is essentially a set of 3 equations derived from the ith point correspondence. Therefore, four such corresponding point pairs would result in 12 similar equations with 12 unknowns; the 8 independent parameters of H and the 4 independent homogenous scale factors $\rho_i, i = 1, ...4$. (Recall from Section B.1.3 that the matrix H is homogeneous and therefore we can impose that one of the 9 elements (say, h_{33}) be 1, leaving eight independent ratios amongst the nine elements of H to be found.)

These 12 equations in 12 unknowns can be determined using standard techniques for solving linear equations. Although solutions for all the unknowns would be obtained, only the elements of H are required to transform all other image points from the projective into the Cartesian plane.

B.2.2   The Direct Linear Transformation (DLT) algorithm

The DLT algorithm is a simple linear algorithm for determining H given a set of ≥ 4 2D to 2D point correspondences, $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$. As described above, the transformation is given by the equation $\mathbf{x}'_i = \mathbf{H}\mathbf{x}_i$ but recall that since the vectors on either side of this equation are homogeneous vectors, they are not necessarily equal; they have the same direction and may differ by a non-zero scale factor. If we are not concerned about working with the homogeneous vector representation of points, we can proceed with the original notation for the transformation equation $\mathbf{x}'_i = \mathbf{H}\mathbf{x}_i$.

Expressing the above transformation as a vector cross product produces:

$\mathbf{x}'_i \times H\mathbf{x}_i = 0$. If the jth row of the matrix H is denoted by $\mathbf{h}^{jT}$, then we may write:

$$H\mathbf{x}_i = \begin{pmatrix} \mathbf{h}^{1T}\mathbf{x}_i \\ \mathbf{h}^{2T}\mathbf{x}_i \\ \mathbf{h}^{3T}\mathbf{x}_i \end{pmatrix} \tag{B.5}$$

Writing $\mathbf{x}'_i = (x'_i, y'_i, w'_i)^T$, the cross product may then be given explicitly as:

$$\mathbf{x}'_i \times H\mathbf{x}_i = \begin{pmatrix} y'_i\mathbf{h}^{3T}\mathbf{x}_i - w'_i\mathbf{h}^{2T}\mathbf{x}_i \\ w'_i\mathbf{h}^{1T}\mathbf{x}_i - x'_i\mathbf{h}^{3T}\mathbf{x}_i \\ x'_i\mathbf{h}^{2T}\mathbf{x}_i - y'_i\mathbf{h}^{1T}\mathbf{x}_i \end{pmatrix} \tag{B.6}$$

Isolating the $\mathbf{h}^j$ entries of H, the above can be written in the form:

$$\begin{bmatrix} \mathbf{0}^T & -w'_i\mathbf{x}_i^T & y'_i\mathbf{x}_i^T \\ w'_i\mathbf{x}_i^T & \mathbf{0}^T & -x'_i\mathbf{x}_i^T \\ -y'_i\mathbf{x}_i^T & x'_i\mathbf{x}_i^T & \mathbf{0}^T \end{bmatrix} \begin{pmatrix} \mathbf{h}^1 \\ \mathbf{h}^2 \\ \mathbf{h}^3 \end{pmatrix} = \mathbf{0} \tag{B.7}$$

These equations have the form $A_i\mathbf{h} = \mathbf{0}$, where A_i is a 3×9 matrix, and $\mathbf{h}$ is a vector made up of the 9 entries of the matrix H. Of the three equations in Equation B.7, only two are linearly independent, thus each point correspondence gives two

equations with the unknown parameters of H. Omitting the 3rd equation in B.7 gives:

$$\begin{bmatrix} \mathbf{0}^T & -w_i'\mathbf{x}_i^T & y_i'\mathbf{x}_i^T \\ w_i'\mathbf{x}_i^T & \mathbf{0}^T & -x_i'\mathbf{x}_i^T \end{bmatrix} \begin{pmatrix} \mathbf{h}^1 \\ \mathbf{h}^2 \\ \mathbf{h}^3 \end{pmatrix} = \mathbf{0} \qquad (B.8)$$

where A_i, now a 2 × 9 matrix is referred to as the DLT matrix equations for the ith point correspondence.

Since each point correspondence gives rise to two independent equations in the entries of H, given a set of four such point correspondences, "we obtain a set of equations $A\mathbf{h} = \mathbf{0}$, where A is the matrix of equation coefficients built from the matrix rows A_i contributed from each correspondence, and $\mathbf{h}$ is the vector of unknown entries of H". With eight equations, and nine unknowns, the solution $\mathbf{h}$ can only be determined up to a non-zero scale factor.

The inhomogeneous case of the DLT algorithm

"An alternative to solving for $\mathbf{h}$ directly as a homogeneous vector is to turn the set of equations B.8 into a inhomogeneous set of linear equations by imposing a condition $h_j = 1$ for some entry of the vector $\mathbf{h}$" [45]. This is justified by recalling from Section B.1.3 that the matrix H is homogeneous and therefore we can impose that one of the 9 elements (say, H_{33}) be 1, leaving eight independent ratios amongst the nine elements of H to be found.

Thus, if h_9, corresponding to H_{33}, is set to one, then certain terms of the matrix product $A\mathbf{h}$ will not contain any of the unknown elements of $\mathbf{h}$ and can be brought

to the other side of the equals sign. This results in:

$$\begin{bmatrix} 0 & 0 & 0 & x_i w_i' & y_i w_i' & w_i w_i' & -x_i y_i' & -y_i y_i' \\ x_i w_i' & y_i w_i' & w_i w_i' & 0 & 0 & 0 & -x_i x_i' & -y_i x_i' \end{bmatrix} \tilde{\mathbf{h}} = \begin{pmatrix} w_i y_i' \\ w_i x_i' \end{pmatrix} \quad \text{(B.9)}$$

The following section describes how to determine $\mathbf{h}$, in both Equations B.8 and B.9.

The over-determined solution

If more than four point correspondences $\mathbf{x}_i \leftrightarrow \mathbf{x}_i'$ are given, then the set of equations $A\mathbf{h} = \mathbf{0}$ (given in Equation B.8) or $A\mathbf{h} = \mathbf{b}$ (given in Equation B.9) is over-determined. If the position of the points is exact then the matrix A will still have rank 8, and there would be an exact solution for $\mathbf{h}$. This will not be the case if the measurement of image coordinates is inexact; that is, if there is noise. There will not be an exact solution to the over-determined system apart from the zero solution [45]. Instead of demanding an exact solution, one can find an *approximate* solution, namely a vector $\mathbf{h}$ that minimizes a suitable cost function [45].

For example, in the inhomogeneous DLT algorithm, $\mathbf{h}$ minimizes $\| (A * \mathbf{h} - \mathbf{b}) \|$, the length of the vector $A\mathbf{h} - \mathbf{b}$. $\mathbf{h}$ is known as the least-squares solution to the over-determined system. The least-squares solution is conveniently found using the Singular Value Decomposition (SVD) [45]. Using the Matlab operator mldivide, $\mathbf{h} = A \backslash \mathbf{b}$ is the solution in the least squares sense to the over-determined system of equations $A\mathbf{h} = \mathbf{b}$ using SVD. Alternatively, the *normal equations* may be used to find the pseudo inverse of A and thus the LS solution. This approach is computationally simpler than SVD when n is small compared to m.

Appendix C

Background on Elliptical Fourier Descriptors

Two methods which can be used to model shape contours are conventional Fourier series and the comparatively recent development, Elliptical Fourier Functions. A brief description of the use of Fourier series for fitting a curve is presented in Section C.1. Elliptical Fourier Functions (EFFs) are a parametric approach to the same problem, which allow for the analysis of contours that cannot be simply represented as single-valued functions (a constraint with conventional Fourier series analysis). EFFs are described in more detail in Section C.2. Within the context of using Fourier series and Elliptical Fourier Functions for fitting curves to biological contours, the author of [84] states that "a particularly attractive property of both of these approaches is that they are *information preserving contour representations* in the sense that they not only allow for the precise reconstruction of a biological outline, but also allow for its re-creation, at any time, in the absence of the original specimen" [84].

C.1 Conventional Fourier Series and Descriptors

A periodic function $f(t)$, of period 2π, can be written as an infinite sum of oscillating sine and cosine terms as follows:

$$f(t) = a_0 + \sum_{n=1}^{\infty} a_n \cos nt + \sum_{n=1}^{\infty} b_n \sin nt, \tag{C.1}$$

where

$$a_0 = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(t)dt, \tag{C.2}$$

$$a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(t) \cos nt dt, \tag{C.3}$$

$$b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(t) \sin nt dt. \tag{C.4}$$

Equation C.1 is referred to as the Fourier series (FS) expansion of $f(t)$; a_n and b_n are thus referred to as the Fourier coefficients for the n_{th} harmonic. The decomposition of a time series into these separate components, or harmonics, is termed harmonic or spectral analysis.

C.1.1 Approximating a curve with FS

An approximation to the function $f(t)$ can be found by truncating Equation C.1 and calculating the expression for only a finite number of harmonics, N, as shown in Equation C.5:

$$f(t) = a_0 + \sum_{n=1}^{N} a_n \cos nt + \sum_{n=1}^{N} b_n \sin nt. \tag{C.5}$$

This expression is now a Fourier Descriptor (FD) of the original function. Of course, the coefficients a_0, a_n, and b_n can only be calculated if the function $f(t)$ is known. If one wants to use Fourier series to describe a curve for an unknown function,

than such an analytical solution cannot be employed. This is the case for the scenario in this thesis; the data represent a tabulated list of (x_i, y_i) point coordinates obtained through measurement. This nonperiodic tabulated function can still be represented with a Fourier series by making it "periodic"; that is, by considering the set of data points to be a repeating pattern over a certain period. "With tabulated functions, harmonic analysis consists of fitting, by a least squares estimation procedure, the regression Equation C.5 to the set of tabulated data, where the Fourier coefficients a_0, a_n, and b_n for all k points and n harmonics, are regression coefficients" [84].

A common approach to dealing with planar tabulated data is to represent it in polar coordinates, such that the Fourier series becomes:

$$f(\theta) = a_0 + \sum_{n=1}^{N} a_n \cos n\theta + \sum_{n=1}^{N} b_n \sin n\theta. \qquad (\text{C.6})$$

where the period is defined over a 2π interval and θ is in radians. In this case, the points on the boundary must be equally spaced (to avoid a weighted analysis); they must be described by distances from a predetermined center within the closed contour; and the contour cannot be double valued (that is, a vector extending from the center cannot intersect the contour in more than one location). The coefficients are found by least-squares estimation methods as follows:

$$a_0 = k^{-1} \sum_{i=0}^{k-1} r_i, \qquad (\text{C.7})$$

$$a_n = \frac{2}{k} \sum_{i=0}^{k-1} r_i \cos n\theta, n = 0, 1, 2, ... \frac{k-1}{2} \qquad (\text{C.8})$$

$$b_n = \frac{2}{k} \sum_{i=0}^{k-1} r_i \sin n\theta, n = 1, 2, ..., \frac{k-1}{2}, \qquad (\text{C.9})$$

where r_i are the distances from the predetermined origin and k is the total number

of measurements. The most logical choice of the origin is the centroid of the contour; other locations may be chosen if they represent some biological landmark of interest, but may introduce increased error in the Fourier coefficients.

The above discussion has described how conventional FDs can be a model for representing the boundary of a morphological form. "The FD approach confers two advantages: (1) that with the advent of computers, the analysis is fairly simple and straightforward; and (2) under certain circumstances, it is possible to attach biological meaning to the Fourier coefficients" [84]. Some limitations with conventional FDs, however, preclude their use in all situations. Limitations include: (1) the tabulated function, $f(x)$, must be single-valued; (2) the data must be set up in polar form; (3) the angles between vectors must be equal (to avoid the complications of a weighted analysis); and (4) integration is sometimes required for evaluation of the Fourier coefficients [84]. These limitations effectively limit FD analysis to forms with relatively simple contours. "Complex morphological forms that contain outlines that cross back onto themselves (producing intersections) for example, would be inadmissible" [84], as would be the presence of multiple values for a single vector. An alternative approach that largely circumvents these constraints, Elliptical Fourier Functions, was developed by Kuhl and Giardina [59] in 1982.

C.2 Elliptical Fourier Functions and Descriptors

According to Kuhl and Giardina, a continuous, closed contour in two dimensions can be represented parametrically as a vector function of time, $V(t)$ [83]. "Parametric refers to a set of equations set up as functions of a common variable; for example, the circle $(x^2 + y^2 = r^2)$ can also be presented with parametric equations $x = r \cos \theta$ and $y = r \sin \theta$ in terms of the parameter θ" [84]. Elliptical Fourier Functions are a parametric formulation whereby the projections of the vector function $V(t)$ on the x

and y axes are separately set up as functions of a third variable, t; that is, $x(t)$ and $y(t)$. This is assuming that the contour is traced at a constant "speed" and that the functions are periodic with period T, where T is the total time required to trace the whole contour. Since these functions are periodic, they can be represented by Fourier series expansions as follows:

$$x(t) = A_0 + \sum_{n=1}^{\infty} a_n \cos \frac{2\pi nt}{T} + \sum_{n=1}^{\infty} b_n \sin \frac{2\pi nt}{T}, \qquad \text{(C.10)}$$

and

$$y(t) = C_0 + \sum_{n=1}^{\infty} c_n \cos \frac{2\pi nt}{T} + \sum_{n=1}^{\infty} d_n \sin \frac{2\pi nt}{T}, \qquad \text{(C.11)}$$

where

$$A_0 = \frac{1}{T} \int_0^T x(t)dt, \qquad \text{(C.12)}$$

$$a_n = \frac{2}{T} \int_0^T x(t) \cos \frac{2\pi nt}{T} dt, \qquad \text{(C.13)}$$

$$b_n = \frac{2}{T} \int_0^T x(t) \sin \frac{2\pi nt}{T} dt. \qquad \text{(C.14)}$$

Similar expressions can be derived for C_0, c_n, and d_n.

This approach overcomes most of the limitations of conventional FDs as discussed above, albeit numerical complexity is increased. "The requirement of equal intervals along the outline is now relaxed and multi-valued functions are no longer a problem" [84], allowing for the numerical characterization of a much larger class of two-dimensional shapes than previously possible.

As is, these expressions still require integrations in order to compute the DC components A_0 and C_0, and the coefficients a_n, b_n, c_n, and d_n. Kuhl and Giardina showed that if $x(t)$ and $y(t)$ are piecewise-linear representations of a contour, continuous for

all time, expressions for the Fourier coefficients can be derived such that no integral needs to be calculated, irrespective of the incremental changes in Δx and Δy. "The derivation of these coefficients involves the time derivatives $\dot{x}(t)$ and $\dot{y}(t)$, which consist of the sequence of piecewise constant derivatives $\frac{\Delta x_p}{\Delta t_p}$ and $\frac{\Delta y_p}{\Delta t_p}$ associated with the time intervals $t_{p-i} < t < t_p$ for values of p in the range of $1 \leq p \leq K$" [59]. Here Δt_p is the time between points p and $p-1$ and K is the number of points along the contour. The time derivatives are periodic with period T and can themselves be represented by a Fourier series, as shown for the $\dot{x}(t)$ case below:

$$\dot{x}(t) = \sum_{n=1}^{\infty} \alpha_n \cos \frac{2\pi nt}{T} + \sum_{n=1}^{\infty} \beta_n \sin \frac{2\pi nt}{T}, \tag{C.15}$$

where

$$\alpha_n = \frac{2}{T} \int_0^T \dot{x}(t) \cos \frac{2\pi nt}{T} dt, \tag{C.16}$$

$$\beta_n = \frac{2}{T} \int_0^T \dot{x}(t) \sin \frac{2\pi nt}{T} dt. \tag{C.17}$$

These coefficients can be expressed as:

$$\alpha_n = \frac{2}{T} \sum_{p=1}^{K} \frac{\Delta x_p}{\Delta t_p} \int_{t_{p-1}}^{t_p} \cos \frac{2\pi nt}{T} dt \tag{C.18}$$

$$= \frac{2}{T} \sum_{p=1}^{K} \frac{\Delta x_p}{\Delta t_p} \left(\sin \frac{2\pi nt_p}{T} - \sin \frac{2\pi nt_{p-1}}{T} \right) \tag{C.19}$$

and

$$\beta_n = \frac{2}{T} \sum_{p=1}^{K} \frac{\Delta x_p}{\Delta t_p} \int_{t_{p-1}}^{t_p} \sin \frac{2\pi nt}{T} dt \tag{C.20}$$

$$= -\frac{2}{T} \sum_{p=1}^{K} \frac{\Delta x_p}{\Delta t_p} \left(\cos \frac{2\pi nt_p}{T} - \cos \frac{2\pi nt_{p-1}}{T} \right). \tag{C.21}$$

But $\dot{x}(t)$ can also be obtained directly from its definition as:

$$\dot{x}(t) = \sum_{n=1}^{\infty} -\frac{2\pi n}{T} a_n \sin \frac{2\pi n t}{T} + \sum_{n=1}^{\infty} \frac{2\pi n t}{T} b_n \cos \frac{2\pi n t}{T}. \tag{C.22}$$

Equating coefficients from the two expressions of $\dot{x}(t)$ given in Equations C.15 and C.22 gives the following expressions for a_n and b_n:

$$a_n = \frac{T}{2\pi^2 n^2} \sum_{p=1}^{K} \frac{\Delta x_p}{\Delta t_p} \left[\cos \frac{2\pi n t_p}{T} - \cos \frac{2\pi n t_{p-1}}{T} \right], \tag{C.23}$$

$$b_n = \frac{T}{2\pi^2 n^2} \sum_{p=1}^{K} \frac{\Delta x_p}{\Delta t_p} \left[\sin \frac{2\pi n t_p}{T} - \sin \frac{2\pi n t_{p-1}}{T} \right]. \tag{C.24}$$

Similarly, the integral formulations for the coefficients of the Fourier series expansion for the y projection of the function $V(t)$ (which would be similar to those given in Equations C.13 and C.14) can be expressed as:

$$c_n = \frac{T}{2\pi^2 n^2} \sum_{p=1}^{K} \frac{\Delta y_p}{\Delta t_p} \left[\cos \frac{2\pi n t_p}{T} - \cos \frac{2\pi n t_{p-1}}{T} \right], \tag{C.25}$$

$$d_n = \frac{T}{2\pi^2 n^2} \sum_{p=1}^{K} \frac{\Delta y_p}{\Delta t_p} \left[\sin \frac{2\pi n t_p}{T} - \sin \frac{2\pi n t_{p-1}}{T} \right]. \tag{C.26}$$

Finally, as for the DC components A_0 and C_0 they are:

$$A_0 = \frac{1}{T} \sum_{p=1}^{K} \frac{\Delta x_p}{2\Delta t_p} (t_p^2 - t_{p-1}^2) + \xi_p (t_p - t_{p-1}), \tag{C.27}$$

$$C_0 = \frac{1}{T} \sum_{p=1}^{K} \frac{\Delta y_p}{2\Delta t_p} (t_p^2 - t_{p-1}^2) + \delta_p (t_p - t_{p-1}), \tag{C.28}$$

where

$$\xi_p = \sum_{j=1}^{p-1} \Delta x_j - \frac{\Delta x_p}{\Delta t_p} \sum_{j=1}^{p-1} \Delta t_j, \tag{C.29}$$

and

$$\delta_p = \sum_{j=1}^{p-1} \Delta y_j - \frac{\Delta y_p}{\Delta t_p} \sum_{j=1}^{p-1} \Delta t_j,$$ (C.30)

and

$$\xi_1 = \delta_1 = 0.$$ (C.31)

The reader is referred to [59] for further explanation of these DC components.

Since the parametric EFF coefficients are now generated using an algebraic approach, instead of the integral solutions required previously, computation is simpler and much faster.

C.2.1 Approximating a curve with EFFs

As with the conventional Fourier series case presented in Section C.1, different levels of approximation to a closed contour can be obtained by using different numbers of harmonics from the Elliptical Fourier Functions. In general, the truncated EFF approximation (or the EFD) of a closed contour based on the first N harmonics can be written as:

$$x_N(t) = A_0 + \sum_{n=1}^{N} a_n \cos \frac{2\pi n t}{T} + \sum_{n=1}^{N} b_n \sin \frac{2\pi n t}{T},$$ (C.32)

and

$$y_N(t) = C_0 + \sum_{n=1}^{N} c_n \cos \frac{2\pi n t}{T} + \sum_{n=1}^{N} d_n \sin \frac{2\pi n t}{T},$$ (C.33)

where n equals the harmonic number, N equals the maximum harmonic number, and the coefficients are as specified in Equations C.23, C.24, C.25, C.26, C.27 and C.28.