MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY

A.I. Technical Report No. 1544 January 1995

Enhanced Reality Visualization in a Surgical Environment

J.P. Mellor
jpmellor@ai.mit.edu

This publication can be retrieved by anonymous ftp to publications.ai.mit.edu.

Abstract

Enhanced reality visualization is the process of enhancing an image by adding to it information which is not present in the original image. A wide variety of information can be added to an image ranging from hidden lines or surfaces to textual or iconic data about a particular part of the image. Enhanced reality visualization is particularly well suited to neurosurgery. By rendering brain structures which are not visible, at the correct location in an image of a patient's head, the surgeon is essentially provided with X-ray vision. He can visualize the spatial relationship between brain structures before he performs a craniotomy and during the surgery he can see what's under the next layer before he cuts through. Given a video image of the patient and a three dimensional model of the patient's brain, the problem enhanced reality visualization faces is to render the model from the correct viewpoint and overlay it on the original image. The relationship between the coordinate frames of the patient, the patient's internal anatomy scans and the image plane of the camera observing the patient must be established. This problem is closely related to the camera calibration problem. This report presents a new approach to finding this relationship and develops a system for performing enhanced reality visualization in a surgical environment. Immediately prior to surgery a few circular fiducials are placed near the surgical site. An initial registration of video and internal data is performed using a laser scanner. Following this, our method is fully automatic, runs in nearly real-time, is accurate to within a pixel, allows both patient and camera motion, automatically corrects for changes to the internal camera parameters (focal length, focus, aperture, etc.) and requires only a single image.

Acknowledgments

This work would not have been possible without the love and support of my family, particularly my wife Kathryn who has been incredibly understanding through it all.

Thanks are due to my thesis supervisor Tomás Lozano-Pérez. His thoughtful advice and confidence in my abilities have helped make this work enjoyable. Thanks are also due to Gil Ettinger, William Wells and Steve White for answering my numerous questions about their work and helping me with the laser scanner, to Greg Klanderman for providing the edge detection code used in Chapter 7 and to Gideon Stein for providing the implementation of the plumb-line method for radial distortion calibration used in Appendix A.

I also wish to thank all of the individuals who make the Artificial Intelligence Laboratory an outstanding research environment. The suggestions and feedback provided by fellow lab members significantly improved this work.

This report is a revised version of a thesis submitted to the Department of Electrical Engineering and Computer Science on 13 January 1995, in partial fulfillment of the requirements for the degree of Master of Science.

This report describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology, and was funded by the Advanced Research Projects Agency of the Department of Defense under Rome Laboratory contract F3060-94-C-0204 and under Office of Naval Research contract N00014-91-J-4038.

To Kathryn, Phillip and Patrick

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Computers in Medicine

The use of computers in medicine has increased dramatically over the last decade [Bemmel *et al.*, 1985, Reggia and Tuhrim, 1985, Miller, 1990, Chang, 1993]. As a result, nearly all aspects of medical care have seen improvement from the introduction of computer-based tools. These tools range from automated patient record keeping to three dimensional visualization of internal anatomy and from computer assisted diagnosis to automatic drug interaction and allergy screening. The use of computers to assist physicians in the planning and execution of surgical procedures is also growing [Lemoine *et al.*, 1991, Smith *et al.*, 1991, Pieper *et al.*, 1992, Verbeeck *et al.*, 1993]. One area which could benefit greatly from more sophisticated computer-based tools is neurosurgery. The need to minimize collateral damage while removing diseased tissue requires extreme precision. In addition, damage to certain critical brain regions, such as the motor strip, must be avoided if at all possible. Planning a surgical approach meeting all of the criteria is difficult and tedious. Identification of specific brain structures and modification of the planned approach are often difficult during the surgical procedure, placing additional emphasis on planning. Traditionally, precision neurosurgery requires the use of a stereotactic frame which is rigidly attached to the patient's skull. Figure 1-1 shows some typical stereotactic frames. The frame is attached prior to and is visible in presurgical imaging such as magnetic resonance (MR) or computed tomography (CT) imaging. This allows surgical plans based on presurgical internal anatomy scans to be transferred to the patient using the stereotactic frame as a reference. Frequently the patient must wear the frame for several days between imaging and surgery. The frames are a significant discomfort to the patient and are cumbersome to the surgeon, possibly limiting his flexibility during the procedure.

A system which improves surgical precision, enables identification of brain structures, allows modification of the planned approach during the surgical procedure and does not require the use of a stereotactic frame would be a vast improvement over traditional neurosurgical procedures.

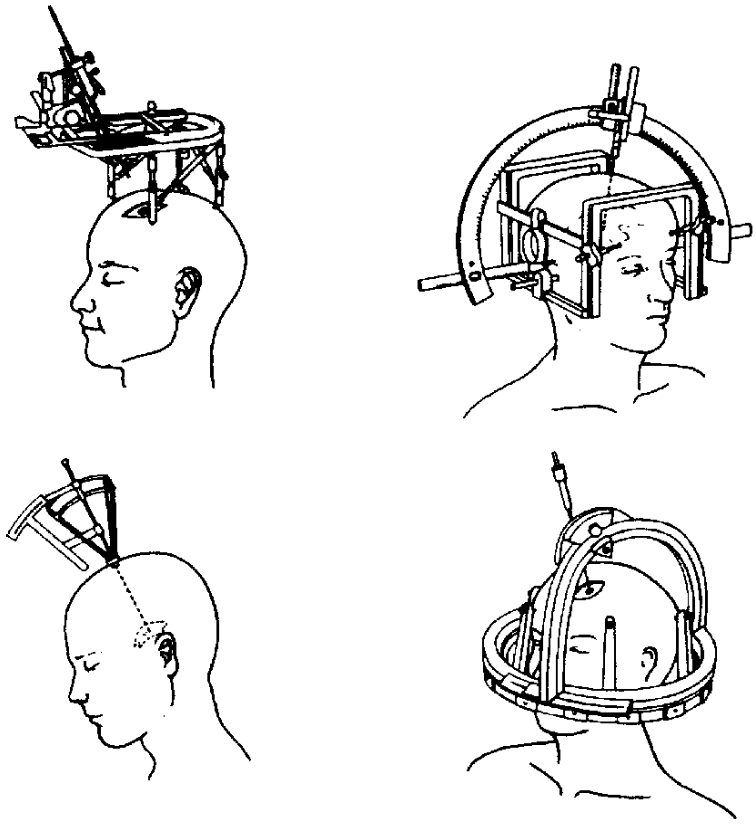

Figure 1-1: Some typical stereotactic frames.

1.2 What is Enhanced Reality Visualization?

Computer assisted surgery is a relatively new development which attempts to provide the surgeon with a tool to assist in the planning and execution of surgical procedures [Adams *et al.*, 1990, Lavallee and Cinquin, 1990]. *Image guided surgery* is a specific type of computer assisted surgery which uses advanced three dimensional visualization techniques to provide the surgeon with a wealth of valuable information not normally available in the operating room [Pelizzari *et al.*, 1991, Wells *et al.*, 1993, Black *et al.*, 1993, Grimson *et al.*, 1994]. In essence, a complex surgical procedure can be navigated visually with great precision by overlaying on an image of the patient a color coded preoperative plan specifying details such as the locations of incisions, areas to be avoided and the diseased tissue. The process of aligning the preoperative plans with and overlaying them on the patient is known as *enhanced reality visualization*. Enhanced reality visualization is the process of enhancing an image by adding information to it. The information added can be anything from text to icons or color coding to three dimensional surfaces. Figure 1-2 shows an enhanced reality visualization of a patient about to undergo neurosurgery. The area shown in ▩ is the tumor to be removed and the ventricles are shown in ▧.

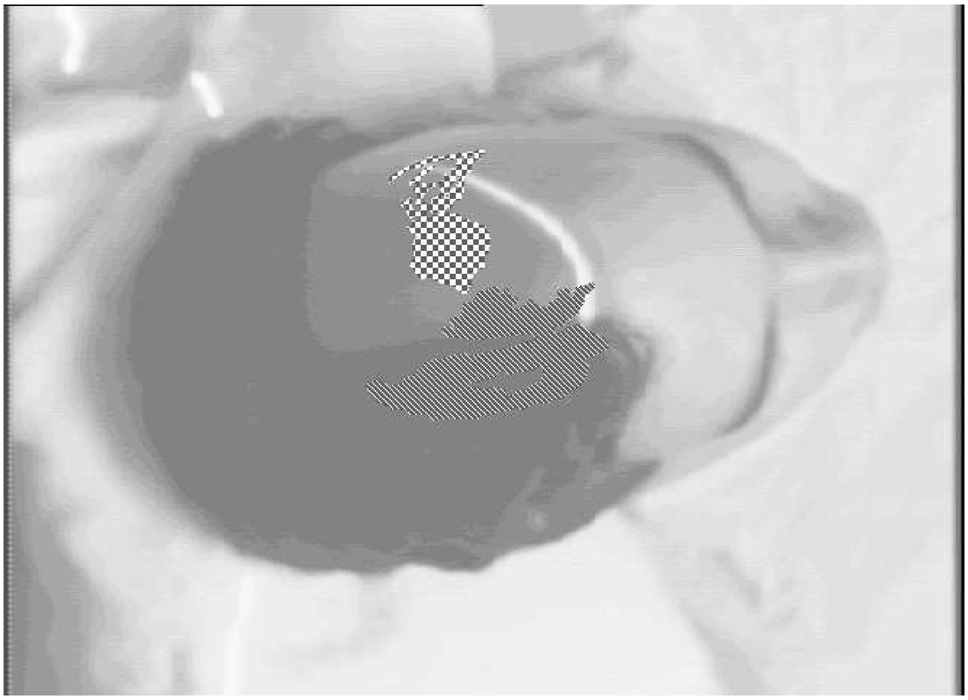

Figure 1-2: Enhanced reality visualization showing a tumor and ventricles.

1.3 A Scenario Which Utilizes Enhanced Reality Visualization

1. A patient needing neurosurgery is scanned by one or more three dimensional, high resolution, internal anatomy scanners, such as magnetic resonance (MR) or computed tomography (CT).

2. Each internal anatomy scan is segmented by tissue type (white matter, gray matter, bone, etc). The various scans of the patient are also registered with one another. The result is a three dimensional model of the patient's brain.

3. Tools which identify, classify and label brain structures such as motor strip and speech centers are used to add the required detail to the model of the patient's brain.

4. Once the model of the patient's brain has been constructed, enhanced reality visualization is possible. Enhanced reality visualization can be used to help plan the surgical procedure. Live video of the patient is mixed with information generated from the brain model allowing the surgeon to view the internal anatomy of the patient in a non-invasive manner. This capability allows the surgeon to test the feasibility of various possible surgical approaches on the actual patient. The enhanced reality visualization may be presented to the surgeon using one of several methods, such as a head-mounted display, a transparent projection screen or a surgical microscope. Details regarding the surgical approach and procedure can be added to the brain model.

5. The surgical procedure is performed using enhanced reality visualization. Enhanced reality enables the surgical site to be precisely located and facilitates accurate transfer of surgical plans to the patient.

1.4 The Problem and Our Approach

Enhanced reality visualization is an integral part of the image guided surgery paradigm, however compared with other aspects, little effort has been expended on this area [Adams *et al.*, 1990, Lavallee and Cinquin, 1990, Pelizzari *et al.*, 1991, Wells *et al.*, 1993, Grimson *et al.*, 1994]. In order to produce enhanced reality visualizations we must be able to quickly and accurately align information such as a brain model with an image. There are several other issues which must be addressed before a full function enhanced reality visualization system can be created. Some of these challenges are listed below.

Display method The displays currently available for enhanced reality visualization are less than optimal. Head mounted displays are still heavy, awkward and have relatively low resolution. Conventional CRT's have better resolution but limit the applications of enhanced reality visualization.

Rendering The complexity of information and the detail and realism with which it can be rendered while updating at a reasonable frame rate are limited.

Information acquisition Acquiring information and converting it to a form suitable for use in enhanced reality visualization can be a difficult and time consuming process.

Virtual reality faces many of the same issues as enhanced reality visualization. There already exists a significant research effort in virtual reality examining these problems. While there are many similarities, enhanced reality differs fundamentally from virtual reality in that it is anchored in the real world. Enhanced reality visualization must align the enhancement with a *real* image quickly and accurately. The ability to perform this alignment quickly and accurately is fundamental to enhanced reality visualization and will be the focus of this report. Given a video image and a three dimensional model, the problem is to render the model from the correct viewpoint and overlay it on the original image. The relationship between the coordinate frames of the world, the model and the image plane of the camera must be established. This problem is closely related to the camera calibration problem. Stated more precisely the problem is to:

> *Determine the perspective transformation which maps model coordinates to image coordinates in "real-time", allowing the information from the model to be added to an image in the correct location.*

An overview of the problem is shown in Figure 1-3. We solve for the transformation which maps model coordinates to image coordinates directly. An alternate approach solves for several transformations and then composes them into a single transformation from model to image coordinates. For example, a reference coordinate system could be defined for the physical object(s). The first step might be to find the transformation which aligns the model with the reference coordinate system. Next, the transformation from reference coordinates to image coordinates must be determined. Solving for intermediate transformations can introduce error into the solution and is computationally more expensive. Unless this data is needed there is no reason to break the problem into several pieces.

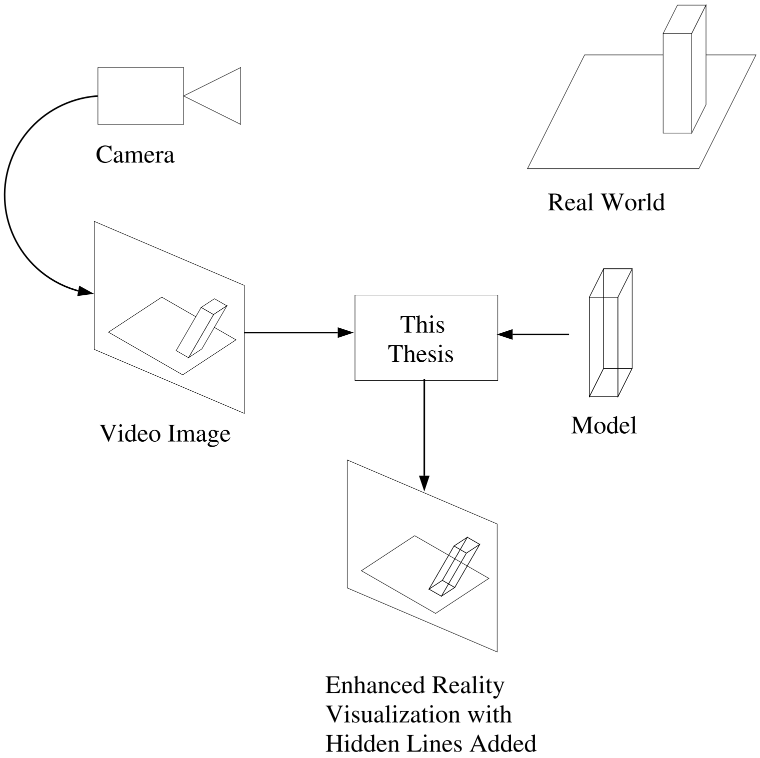

Figure 1-3: Overview of this report.

We will define several terms that will be used throughout this report. An enhanced reality visualization is composed of a virtual image overlayed on a raw image. A **raw image** is an image of the real world prior to any *enhancement*. The coordinates of the raw image are referred to as **image coordinates**. The information which will be added to the raw image to produce the enhancement is referred to as the **model**. The coordinates of the model are referred to as **model coordinates**. A **virtual image** is generated by rendering the model from a particular view point. The **view point** captures the relative placement and orientation between model and viewer. Finally, the term **world coordinates** is used to refer to an arbitrary coordinate system attached to an object visible in the raw image.

Figure 1-4 shows an overview of our method in the context of neurosurgery. A novel formulation of the camera calibration problem allows us to quickly and easily obtain the perspective transformation mapping model (MR or CT) coordinates to image coordinates. The perspective transformation is then used to generate the enhanced reality visualization. Our approach utilizes several circular fiducials placed near the surgical site.

The circular fiducials enable us to recover a scale factor at each fiducial (the focal length of the camera divided by the depth of the fiducial). Given the scale factor as well as the image and model (MR or CT) coordinates for each fiducial,

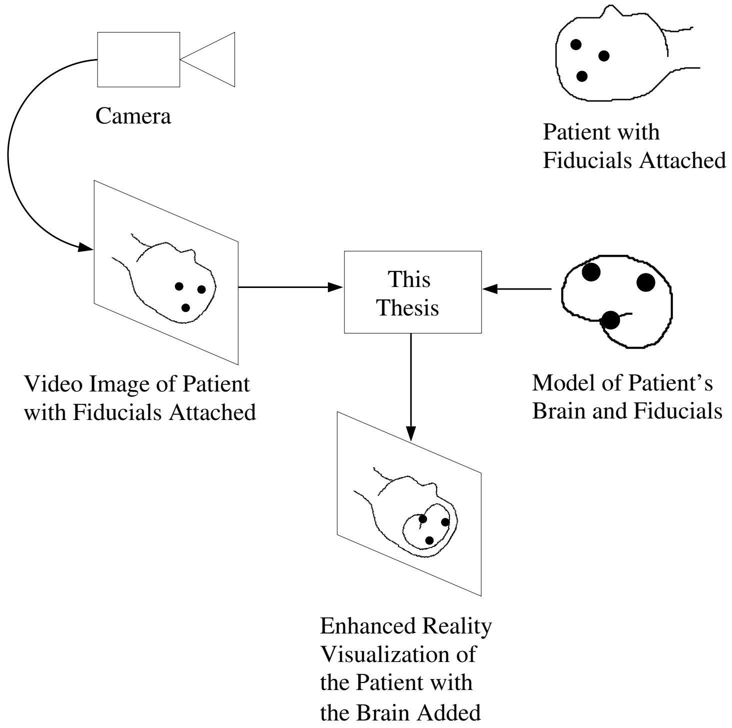

Figure 1-4: Overview of this report showing the circular fiducials.

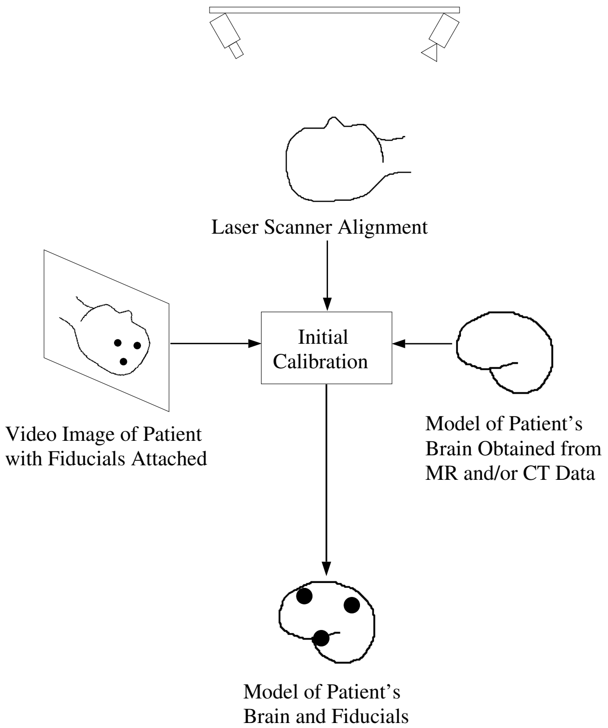

Figure 1-5: Determining the model (MR) coordinates of the fiducials.

the problem of determining the perspective transformation which maps model coordinates to image coordinates reduces to three sets of linear equations. In general, the model coordinates of the fiducials are not known a priori, and they must be calibrated. Figure 1-5 depicts the process of determining the model (MR or CT) coordinates of the fiducials. During an initial calibration phase a laser scanner is used to register the world coordinates of the fiducials with the model coordinates of the MR or CT data. Once the initial calibration is complete, our method is fully automatic and uses visual information exclusively. Some of the additional characteristics of our approach which make it particularly well suited to enhanced reality visualization are:

1. Requires only a single image with a few fiducials
2. Does not require internal calibration or known focal length
3. Accurate to within a pixel
4. Solution is non-iterative

The remaining chapters of this report are organized as follows: Chapter 2 reviews current enhanced reality visualization techniques. Chapter 3 develops a basic camera model and contains a brief discussion of current camera calibration techniques. Chapter 4 presents our method and an overview of its implementation. Chapter 5 provides the details (theory, error analysis and empirical results) associated with circular fiducials. Chapter 6 discusses one method of calibrating fiducials. Chapter 7 shows the results of our method for a test object and a plastic skull. Finally, Chapter 8 presents our conclusions.

Chapter 2

Related Work

Several groups of researchers have recently been investigating enhanced reality. The proposed applications for enhanced reality range from laser printer repair to aircraft manufacture. Current research efforts in enhanced reality visualization differ in many implementation details. The one thing they all have in common is the requirement to align a model with an image of the real world. In this chapter we will examine several different approaches.

2.1 Medical Applications

2.1.1 Aachen University of Technology

A group at the Aachen University of Technology in Germany has developed a "Computer Assisted Surgery" module for use in ENT surgical procedures [Adams *et al.*, 1990]. A model of the patient is produced from presurgical CT scans. Radiopaque markings are attached to the patient's skull prior to the presurgical scans for use as reference points. The system is calibrated using a hand-guided electro-mechanical three dimensional coordinate digitizer. The digitizer is used to measure the positions of several reference points. With correspondence between digitizer and CT points the transformation from the CT coordinates to digitizer coordinates can be calculated using 3D/3D matching. During an operation the surgeon can use the digitizer to point at an unidentified structure. Three perpendicular views of the CT data corresponding to the location of the digitizer are then displayed on a nearby CRT. The system must be recalibrated every time the patient moves with respect to the digitizer. The reported accuracy is better than ± 1mm.

2.1.2 TIMB

A group at TIMB in Grenoble, France has developed a "Computer Assisted Medical Intervention" module [Lavallee and Cinquin, 1990]. A model of the patient's internal anatomy is produced from presurgical imaging. This system uses a surgical robot or guidance system with modes that range from passive to semi-autonomous. In passive mode the robot provides visual feedback by overlaying the position of an instrumented probe on the presurgical scans. In semi-autonomous mode some portions of the surgical procedure are performed by the robot under the supervision of the surgeon. The system is calibrated using a special calibration cage made of four Plexiglas planes containing metallic balls. The calibration cage is placed near the patient and a pair of X-ray images are taken. The relationship between the presurgical imaging, the X-ray device and the surgical robot can be established using 3D/3D matching. Again if the patient moves relative to the robot (or the X-ray device) the system must be recalibrated. Accuracy for the instrumented probe is reported as ~5mm. Accuracies for other modes are not reported.

2.1.3 University of Chicago

A group at the University of Chicago has developed a method for "Interactive 3D Patient - Image Registration" [Pelizzari *et al.*, 1991]. The method is used to position patients for radiation therapy. Again, a model of the patient's internal anatomy is produced from presurgical imaging. The model is used to plan the geometry of radiation therapy beams. Because of the non-invasive nature of radiation therapy it is difficult to transfer the beam geometry planned using the model to the actual patient. A Polhemus 3Space tracker and localizer are used as a magnetic 3D digitizer to measure the surface of the patient. The model and the measured surface are then registered using 3D/3D surface fitting. Once the registration has been performed, the magnetic digitizer is used again to mark the patient for setup. The intersections of the three principle planes with the patient are traced. These marks are then used as reference for positioning the therapy machine. The therapy machine must be aligned manually. If the patient moves the entire calibration need not be reperformed, however the therapy machine must be realigned with the reference marks on the patient. Accuracies of ~1mm and ~1° are reported.

2.1.4 Massachusetts Institute of Technology

A group at MIT's Artificial Intelligence Laboratory has developed "An Automatic Registration Method for Frameless Stereotaxy, Image Guided Surgery, and Enhanced Reality Visualization" [Grimson *et al.*, 1994]. As in the previous work described, a model of the patient's internal anatomy is produced from presurgical imaging. A Technical Arts laser range scanner is used to collect a set of 3D data points from the patient's skin surface. The model and the laser data are registered using 3D/3D surface matching. A special calibration object is used to calibrate the laser scanner and calibrate the location of a camera on the laser scanner. Once the model is registered with the laser data and the location of the laser scanner camera is calibrated, the model can be overlayed with video of the patient. The calibration must be reperformed if the patient or camera move and the overlay can only be generated for a single viewpoint. The reported accuracy of this method is $\sim$1mm.

2.1.5 Stanford

A group at Stanford University has developed "Treatment Planning for a Radiosurgical System with General Kinematics" [Schweikard *et al.*, 1994]. The method is used to plan and perform radiosurgery. A model of the patient's internal anatomy is produced from presurgical imaging. The radiosurgery is planned using the model. In addition, the model is used to synthesize radiographs from different view points. A total of over 400 such radiographs are produced. During the radiosurgery, two nearly orthogonal X-ray images of the patient are taken and compared with the precomputed radiographs to determine the patient's position and orientation with respect to the treatment machine. The patient's position and orientation can be determined about twice per second. The treatment machine (X-ray system, radiation source, etc.) must be calibrated separately using a special calibration routine. The accuracy of this method is not reported.

2.1.6 University of North Carolina

A group at the University of North Carolina has developed a method for "Merging Virtual Objects with the Real World" [Bajura *et al.*, 1992]. This system allows the user to see ultrasound imagery overlaid on a patient in near real-time. A six degrees of freedom (DOF) Polhemus 3Space tracker is mounted on the probe used to acquire ultrasound images. A second 6DOF tracker is attached to the head-mounted display (HMD) used to view the overlay. Images from a camera also mounted on the HMD are combined with the ultrasound images to produce the overlay. Since both trackers report position and orientation it is a simple matter to transform between ultrasound tracker coordinates and HMD tracker coordinates. In order to transform ultrasound images into the coordinate system of the HMD camera the relationships between ultrasound images and the ultrasound tracker and the HMD camera and the HMD tracker must be established. A special calibration jig is used to determine these transformations periodically. This system allows for motion of both the user and the patient. The accuracy of the overlay is not reported.

2.2 Other Applications

2.2.1 Boeing

A group at Boeing has developed "An Application of Heads-Up Display Technology to Manual Manufacturing Processes" [Caudell and Mizell, 1992]. The goal of this work is to overlay manufacturing instructions on images of the manufacturing process and display them to a worker using an HMD. The instructions to be overlayed are derived from a CAD model. The system uses a 6DOF Polhemus 3D Isotrack magnetic tracking system attached to the HMD to generate the overlay. A calibration jig is used to establish the relationship between the HMD and the work site. Given the relationship between the HMD and the work site an overlay can easily be generated. The accuracy of this system is not reported.

2.2.2 Columbia University

A group at Columbia University has developed a method for "Knowledge-Based Augmented Reality" [Feiner *et al.*, 1993]. The goal of this work is to overlay instructions for repairing a laser printer with images of the laser printer. The instructions are derived from a knowledge-based system. A Logitech 3D ultrasonic tracking system and an Ascension Technology magnetic tracking system are used to determine the position and orientation of an HMD and several key parts of the laser printer. Using the position and orientation information from the tracking system an instruction overlay is generated and displayed to the user via the HMD. Frame rates of about 15hz are reported. Accuracy is not reported.

2.3 Discussion

As discussed in Section 1.4 the transformation which maps model coordinates to image coordinates can be divided into several pieces. All of the methods discussed above take this approach and all of them calculate a transformation which registers the model with some world (reference) coordinate system. Initially, our discussion will consider only this piece of the larger transformation.

Current methods of registering the model with the world coordinate system are somewhat limited. Many of the approaches use magnetic trackers to determine the transformation which will align the two coordinate frames. Magnetic trackers have several significant shortcomings which limit their effectiveness for enhanced reality visualization. Magnetic trackers have a very short range, typically a few feet. Accuracies are limited to ~6mm and ~1.5°.[1] Perhaps the most significant limitation is that magnetic and metallic objects can severely degrade the performance of magnetic trackers. This is a significant limitation for surgical applications as most operating rooms are loaded with metallic and magnetic objects. In addition, current advances in intra-operative imaging have made it possible to obtain MR images of a patient's internal anatomy during surgery. This environment precludes the use of magnetic trackers.

Several other types of sensors are also used to determine the transformation which will register the model with the world coordinate system. These sensors also have limitations which make them less than desirable for enhanced reality visualization. Ultrasonic trackers have range and accuracy limitations similar to those of magnetic trackers. While they are not susceptible to magnetic interference they are limited to line of sight operation. Mechanical digitizers have good accuracy but are cumbersome and have limited range. The laser scanner provides very accurate position information but also has a short range and is limited to line of sight operation.

All of the methods cited above use active sensors to collect the data required to register the model with the world coordinate system. This requires either a special environment (mechanical digitizers, magnetic trackers and ultrasonic trackers) or a cumbersome piece of equipment (laser scanner and X-ray). There are also many situations where active emissions are not desirable.

The registration produced by most of the medical applications (the exceptions are the work being done by the groups at the University of North Carolina and Stanford University), is limited to a fixed patient and sensor configuration. They do not allow for patient motion. This is because the alignment between the patient and the world coordinate system is implicitly determined during the initial calibration phase and is not monitored. It is not clear that it is possible to extend these methods to allow for motion. [Grimson *et al.*, 1994] claim that their method is extensible to cover patient motion, however it is not clear that it is practical or possible to do so using a laser scanner. In a surgical environment, with all but the surgical site hidden under sterile drapes, it is doubtful that enough patient surface area will be visible to the scanner to allow registration of the patient and model. In addition, using a laser in the operating room might be distracting to the surgeons.

These methods also require a high degree of operator action to register the model with the patient. Typically the operator must measure data points by hand or edit data that was semi-automatically collected. At least one of the methods requires about half an hour to produce a single registration.

While all of the medical applications register a model obtained from presurgical imaging to a world coordinate system, only two of the applications (Massachusetts Institute of Technology and University of North Carolina) actually

[1] These accuracies are for a sensor located between 10 and 70cm from the source.

produce enhanced reality visualizations. Neither of these methods can handle dynamic changes in focus, zoom or aperture of the camera used to obtain the raw image for the enhanced reality visualization. And only the method produced by the group at the University of North Carolina allows the surgeon to change the viewpoint of the enhanced reality visualization.

None of the current approaches to enhanced reality visualization are particularly well suited to a surgical environment. The sensors used are active and are frequently cumbersome. A good solution for a surgical environment should allow for patient motion and should allow the surgeon to see enhanced reality visualizations from different view points. It should be fully automatic following a straight forward initial calibration. And the method should allow for dynamic changes in the camera parameters. We will consider optical sensors for a number of reasons. Small, light weight and inexpensive video cameras are readily available. Very accurate results can be achieved with these cameras which can operate over long ranges. Further, we will obtain the information required to register the model with the raw image entirely from the raw image. This has the advantage of ensuring that registration information is available exactly when raw images are available to produce an enhanced reality visualization. Since the goal of enhanced reality visualization is to *enhance* an image, the raw image will likely contain a lot of information valuable to performing the enhancement. Almost all of the current approaches ignore the information contained in the raw image opting for what is essentially a closed loop solution.

Recently [Wells *et al.*, 1993] proposed a method of enhanced reality visualization using video information exclusively, however the method requires manual alignment of the model with the video image and in some cases requires markers to appear in both the MR image and video image. An optical tracker has also been proposed by a group at the University of North Carolina [Wang *et al.*, 1990, Ward *et al.*, 1992, Gottschalk and Hughes, 1993, Azuma and Biship, 1994]. This method is essentially another active sensor not unlike a laser scanner. It uses a "sea-of-lights" consisting of nearly 1000 LED's mounted in the ceiling tiles of 10' by 12' room. Three cameras mounted on an HMD are aimed at the ceiling while the LED's are flashed sequentially. This "optical tracker" requires a special ceiling which must be calibrated and a significant amount of additional hardware (3 extra cameras, LED control, etc). Neither of these, proposals are suitable in our application for the reasons cited above.

Chapter 3

Camera Calibration

3.1 Camera Model

The pin-hole camera is frequently used to model the transformation from world coordinates to image plane. The pin-hole model uses the perspective projection model of image formation. Orthographic projection with scale or weak perspective is used in many computer vision applications, however it is not accurate enough across the entire image for our application. For example consider an object with 5cm of depth located 100cm away from the camera and 5cm off-axis, see Figure 3-1. Under orthographic projection points a and b are collocated, however in a real image (using a 25mm lens) the points are 5 pixels apart. The effect is significant even with the object only slightly off center. The left side of Figure 3-2 shows a camera centered Cartesian coordinate system. The optic axis of the camera is coincident with the z axis. The image plane is parallel to the xy plane and located a distance f from the origin. Even though the image plane is not required to be parallel to the xy plane, most camera models do not explicitly consider this possibility. Image plane pitch θ_{x} and tilt θ_{y} are usually reflected in pose. We will start with the assumption that $\theta_{\mathrm{x}} = \theta_{\mathrm{y}} = 0$ and then in Chapter 4 we will show how our method implicitly models image plane pitch and tilt. The point where the image plane and the optic axis intersect is known as the principal point. Under perspective projection a point $P_{\mathrm{c}} = [x_{\mathrm{c}}\ y_{\mathrm{c}}\ z_{\mathrm{c}}]$ projects to point $p = [x\ y]$ on the image plane by the following equations[1]:

$$x = f\frac{x_{\mathrm{c}}}{z_{\mathrm{c}}} \tag{3.1}$$

$$y = f\frac{y_{\mathrm{c}}}{z_{\mathrm{c}}} \tag{3.2}$$

Unfortunately, we are not able to directly access the image plane coordinates. Instead we have access to an array of pixels in a frame buffer or computer

[1] We represent points as row vectors rather than column vectors. This means that points will be premultiplied instead of postmultiplied when applying a transformation. This is the exact opposite of what is typically used in computer vision, however it is the notation that the author was first exposed to and what has stuck.

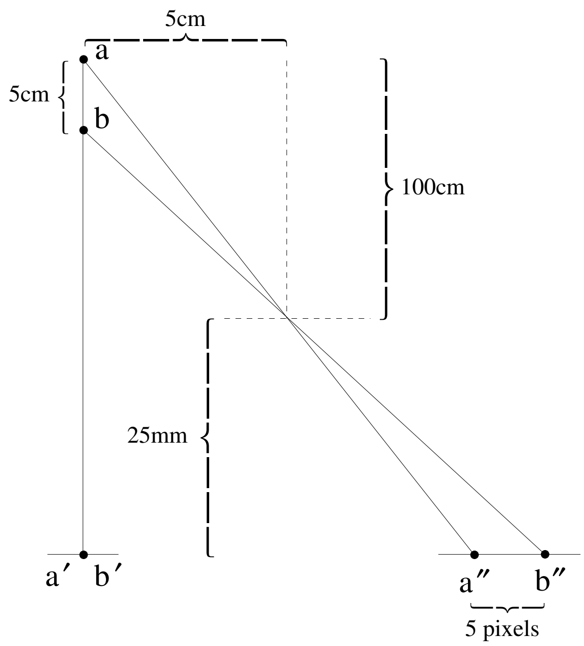

Figure 3-1: Effect of orthographic projection assumption.

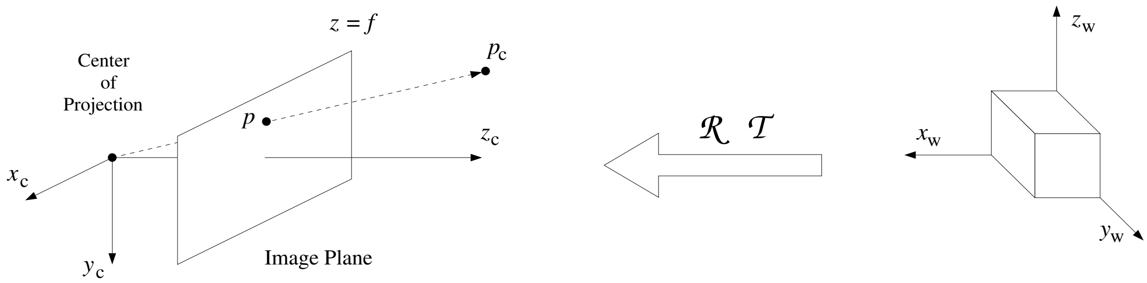

Figure 3-2: Transformation from world coordinates to the image plane of an ideal pin-hole camera.

memory. In order to understand the relationship between image plane coordinates and the array of pixels, we must examine the imaging process. The image plane of a CCD camera is a rectangular array of discrete light sensitive elements. The output of each of these elements is proportional to the amount of light which falls on it. The values for each of these elements are read out one element at a time row after row until the entire sensor array has been read. This analog signal is processed by a frame grabber which converts the camera's output to digital values and stores them in the frame buffer. The result is an array of digital values in the memory of the computer. The rows of the array correspond to the rows of the image sensor. In general, this is not the case for the columns. A synchronization signal is provided between rows, however the frame grabber samples the signal within a row asynchronously and at an independent frequency which may result in a different number of columns per row than were present in the camera. Synchronization errors can also cause the rows to not line up. This is known as pixel jitter and in extreme cases can cause the x and y axes to appear non-orthogonal or skewed [Lenz and Tsai, 1988]. Most camera models omit skew angle θ_{xy} (the angle between the x and y axes minus 90°). We will start with the assumption that $\theta_{\mathrm{xy}} = 0$ for simplicity and then in Chapter 4 we will show how our method implicitly models skew angle. A single element of the array in memory is commonly called a pixel[2]. We will refer to the row and column number of a given pixel as y' and x' respectively. Several parameters are defined to quantify the relationship between the array in memory and the coordinate system of the image plane. x_0 and y_0 are the pixel coordinates of the principal point. s_{x} and s_{y} are the number of pixels in memory per unit distance in the x and y direction of the image plane. These parameters along with f, θ_{x}, θ_{y} and θ_{xy} are intrinsic or internal camera calibration parameters. The projection of point P_{c} to point $p' = [x'\ y']$ in memory is described by the following equations:

$$x' - x_0 = f s_{\mathrm{x}} \frac{x_{\mathrm{c}}}{z_{\mathrm{c}}} \tag{3.3}$$

$$y' - y_0 = f s_{\mathrm{y}} \frac{y_{\mathrm{c}}}{z_{\mathrm{c}}} \tag{3.4}$$

The right half of Figure 3-2 shows an arbitrary Cartesian coordinate system which we will refer to as the world coordinate system. A point P_{w} in the world coordinate system is transformed into the camera centered coordinate system

[2] As noted above pixels in memory can differ in size from the underlying image sensing elements. Many of the measures of accuracy cited both in this work and in the literature should actually be made relative to the image sensing elements. This is straight forward for a calibrated camera. We are working with uncalibrated cameras and the relationship between image sensing elements and pixels in memory is frequently not known. Thus for simplicity and in spite of the preceding, we will express our measures in terms of pixels in memory.

by the following equation:

$$P_c = P_w \mathcal{R} + \mathcal{T} \tag{3.5}$$

where $\mathcal{R}$ is an orthonormal rotation matrix and $\mathcal{T}$ is a translation vector. $\mathcal{R}$ and $\mathcal{T}$ together are commonly referred to as the extrinsic or external camera parameters.

The pin-hole model assumes an ideal camera. Because of lens distortion, real cameras deviate from ideal. The major categories of lens distortion are:

1. Radial distortion - the path of a light ray traveling from the object to the image plane through the lens is not always a straight line.

2. Decentering distortion - the optic axis of individual lens components are not always collinear.

3. Thin prism distortion - the optic axis of the lens assembly is not always perpendicular to the image plane.

When it is necessary to explicitly model lens distortion it is typically sufficient to model only radial distortion. Our method, developed in Chapter 4, implicitly models a linear approximation to lens distortion.[3] Using a 25mm lens of average quality we have not found it necessary to explicitly model lens distortion. The maximum radial distortion at the extreme edge of the image for our configuration is just a few pixels.

3.2 Current Camera Calibration Techniques

Research into the camera calibration problem has a long history originating in the field of photogrammetry. For a more complete discussion of camera calibration techniques see [Slama, 1980, Tsai, 1987]. Camera calibration techniques can be divided into three different categories:

1. Methods which recover only intrinsic parameters. These methods generally require a special calibration object or stand to allow the internal parameter(s) to be measured independent of other parameters. Also, these methods assume that the intrinsic parameters do not change. Unless extreme care is taken to ensure otherwise, it is almost certain that the intrinsic parameters will change perhaps by a significant amount following calibration. Examples of these methods include [Brown, 1965, Lenz and Tsai, 1988, Maybank and Faugeras, 1992].

[3] A discussion of this characteristic can be found in Appendix A.

2. Methods which recover only extrinsic parameters (also known as geometric methods). These methods assume that the intrinsic camera parameters are precisely known. This is not always possible for the reasons mentioned above. Examples of these methods include [Church, 1945, Fischler and Bolles, 1981].

3. Methods which recover both intrinsic and extrinsic parameters. These methods can be further divided into two categories:

   (a) Nonlinear optimization methods. These methods are both nonlinear and iterative. These methods typically produce the most accurate results but require a large number of features and a significant amount of time. Further they are frequently not automatic and need a good initial solution to ensure convergence. Finding an initial solution can be a difficult problem. Examples of these methods include [Faig, 1975, Sobel, 1974].

   (b) Linearization methods. These methods linearize the nonlinear projection equations by introducing additional constraints. The basic difference between members of this category is how the problem is linearized. If care is not taken when the equations are linearized significant bias can be introduced. Many of these methods are iterative and under certain conditions fail to converge. These methods tend to be significantly faster than the nonlinear optimization methods. They frequently do not require an initial solution or can calculate one with relative ease. These methods still require a large number of points for good results. Examples of these methods include [Faugeras and Toscani, 1987, Grosky and Tamburino, 1987, Tsai, 1987, Goshtasby, 1987, Ganapathy, 1984].

None of the current solutions to the camera calibration problem are ideally suited to enhanced reality visualization. The linearization methods come closest to meeting the requirements of enhanced reality visualization, however there is room for improvement.

Chapter 4

Our Solution

We have developed a novel method for determining the relationship between model and image coordinates. Figure 4-1 provides an overview of our method. Two key insights lead to a significant simplification of the problem. These insights are:

1. It is not necessary to separate the intrinsic and extrinsic parameters for enhanced reality visualization.

2. It is possible to recover depth information from a single 2D image.

Utilizing these insights produces a solution that is particularly well suited to enhanced reality visualization and meets the requirements discussed in Chapters 1 and 2. Our solution is most closely related to the linearization methods described in Chapter 3.

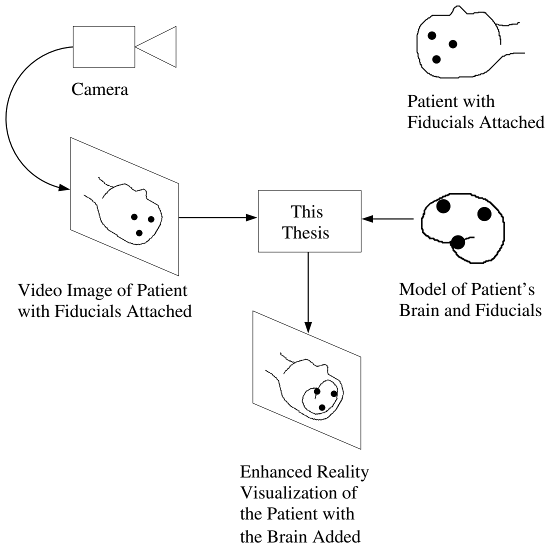

Figure 4-1: Overview of this report showing the circular fiducials.

4.1 A Perspective Transformation is Enough

The camera calibration problem is typically posed as follows:

$$I = M\mathcal{X}\mathcal{C} \tag{4.1}$$

where:

$$\begin{aligned}
\mathcal{X} &= \begin{bmatrix} \mathcal{R} \\ \mathcal{T} \end{bmatrix} \\
&= \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \\ t_{\mathrm{x}} & t_{\mathrm{y}} & t_{\mathrm{z}} \end{bmatrix} \\
\mathcal{C} &= \begin{bmatrix} s_{\mathrm{x}}f & 0 & 0 \\ 0 & s_{\mathrm{y}}f & 0 \\ x_0 & y_0 & 1 \end{bmatrix}
\end{aligned}$$

M is a matrix of model points of the form $[X\ Y\ Z\ 1]$, I is a matrix of image points of the form $[x\ y\ z]$ resulting from the projection of M onto the image plane, $\mathcal{X}$ is an external camera calibration matrix, $\mathcal{R}$ is an orthonormal rotation matrix, $\mathcal{T}$ is a translation vector, and $\mathcal{C}$ is an internal camera calibration matrix. Image points are expressed in homogeneous coordinates to allow the perspective projection to be captured using linear equations [Duda and Hart, 1973]. The pixel coordinates of an image point $[x'\ y']$ are determined by the following relationships: $x' = x/z$ and $y' = y/z$. These relationships are very similar to (3.3) and (3.4). In fact, x, y and z are analogous to the camera centered coordinates x_{c}, y_{c} and z_{c}.

The ultimate goal of most camera calibration is to enable the recovery of metric 3D information, such as the pose (position and orientation) of an object, from its two dimensional image. Clearly, to recover the pose of an object it is necessary to separate the intrinsic and extrinsic parameters. Separating the parameters is difficult [Ganapathy, 1984]. The problem is nonlinear and several of the parameters are closely coupled. In the presence of noise a single solution to the camera calibration problem does not exist, rather there exists a set of solutions. These solutions can differ significantly and yet give rise to nearly identical images. For example, in the presence of noise significant trade offs can be made between t_{z} and f. This can result in a solution which looks good from one view point but where neither the intrinsic nor extrinsic parameters are correct. The fact that the optimal solution for one view point may not be the globally optimal solution is at the heart of what makes camera calibration hard.

Camera calibration is often performed as a preliminary step in many applications. A set of camera calibration parameters is recovered and the intrinsic parameters are used for future images. This assumes that a globally optimal set of intrinsic parameters has been recovered and that the parameters are fixed. By using multiple images or multiple calibration planes the solution can be improved in a global sense. However, in the presence of noise small errors can lead to large errors for view points significantly different than those used during calibration. Further, the intrinsic camera calibration parameters are not fixed. They change with the focus and aperture settings. For example, the principle point can shift by 8 pixels or more with adjustments to focus [Willson and Shafer, 1993]. The effective focal length f also varies with focus and aperture settings. Zoom lenses take this variability to an extreme, enabling large changes to f. Lens distortion also varies with changes to focus and aperture [Brown, 1965].

In enhanced reality visualization we are interested in the total transformation from model to image coordinates. We do not need to separate intrinsic and extrinsic parameters to generate an enhanced reality image. All of the parameters comprising all of the intrinsic and extrinsic calibration parameters can be composed into a single 3×4 matrix. This insight is not new, but how we apply it is. The following matrix equation is equivalent to (4.1) and the combination of (3.3), (3.4) and (3.5).

$$I = M\mathcal{P} \tag{4.2}$$

where:

$$\begin{aligned}\mathcal{P} &= \mathcal{X}\mathcal{C} \\ &= \begin{bmatrix} r_{11}s_\mathrm{x}f + r_{13}x_0 & r_{12}s_\mathrm{y}f + r_{13}y_0 & r_{13} \\ r_{21}s_\mathrm{x}f + r_{23}x_0 & r_{22}s_\mathrm{y}f + r_{23}y_0 & r_{23} \\ r_{31}s_\mathrm{x}f + r_{33}x_0 & r_{32}s_\mathrm{y}f + r_{33}y_0 & r_{33} \\ t_\mathrm{x}s_\mathrm{x}f + t_\mathrm{z}x_0 & t_\mathrm{y}s_\mathrm{y}f + t_\mathrm{z}y_0 & t_\mathrm{z} \end{bmatrix}\end{aligned} \tag{4.3}$$

For our purposes finding values for the elements of $\mathcal{P}$ is sufficient.

A more general internal calibration matrix can be defined as follows:

$$\mathcal{C} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \tag{4.4}$$

This new definition of $\mathcal{C}$ has 9 degrees of freedom. These degrees of freedom correspond to x_0, y_0, s_x, s_y, f, θ_xy, θ_x, θ_y and θ_z. Only 5 of these 9 degrees of freedom are unambiguous. s_x, s_y and f actually constitute 2 degrees of freedom. This is equivalent to saying that $\mathcal{C}$ is only defined up to a scale factor.

θ_x, θ_y and θ_z are redundant degrees of rotational freedom which are captured in $\mathcal{X}$. Therefore the internal calibration matrix used in (4.1) needs only slight modification: the addition of a skew parameter θ_{xy}. The skew parameter is added in the 1st column, 2nd row of $\mathcal{C}$. The value of the skew parameter is equal to $\tan\theta_{xy}$ or change in the x coordinate based on the y coordinate. With the addition of this parameter to **C** the perspective transformation ($\mathcal{XC}$) matrix becomes:

$$\mathcal{P} = \begin{bmatrix} r_{11}s_x f + r_{12}\tan\theta_{xy} + r_{13}x_0 & r_{12}s_y f + r_{13}y_0 & r_{13} \\ r_{21}s_x f + r_{22}\tan\theta_{xy} + r_{23}x_0 & r_{22}s_y f + r_{23}y_0 & r_{23} \\ r_{31}s_x f + r_{32}\tan\theta_{xy} + r_{33}x_0 & r_{32}s_y f + r_{33}y_0 & r_{33} \\ t_x s_x f + t_y\tan\theta_{xy} + t_z x_0 & t_y s_y f + t_z y_0 & t_z \end{bmatrix} \tag{4.5}$$

$\mathcal{P}$ can exactly model non-orthogonal frame buffer axes ($\theta_{xy} \neq 0$) and perspective projection with the image plane not perpendicular to the optic axis (θ_x and/or $\theta_y \neq 0$). A linear approximation of radial distortion can also be obtained[1]. Notice that the revised definition of $\mathcal{C}$ has 5 degrees of freedom and when combined with the 6 degrees of freedom contained in $\mathcal{X}$ accounts for all 11 degrees of freedom in $\mathcal{P}$. Thus $\mathcal{P}$ implicitly models 5 intrinsic and 6 extrinsic parameters. The modeling is implicit because the underlying physical parameters are never actually computed. By formulating the problem in this manner, we avoid the difficulties associated with decomposing the intrinsic and extrinsic camera parameters. We solve (4.2) for each image we obtain. Thus we do not need to worry about finding a globally optimal solution, optimal for this view point is sufficient. Further, changes to the intrinsic camera parameters are inherently captured.

4.2 Depth Information From a Single 2D Image

Even with the simplifications made so far, the problem of solving for the perspective transformation which maps model coordinates to image coordinates is still nonlinear. While it is possible to solve for the elements of $\mathcal{P}$ using a minimum of 6 point features and an iterative method, we can do better. Expanding (4.2) produces:

$$\begin{aligned} x' &= x/z \\ &= \frac{p_{11}X + p_{21}Y + p_{31}Z + p_{41}}{p_{13}X + p_{23}Y + p_{33}Z + p_{43}} \end{aligned} \tag{4.6}$$

[1] A discussion of this characteristic can be found in Appendix A.

$$\begin{aligned} y' &= y/z \\ &= \frac{p_{12}X + p_{22}Y + p_{32}Z + p_{42}}{p_{13}X + p_{23}Y + p_{33}Z + p_{43}} \end{aligned} \tag{4.7}$$

If we knew the value of $z = p_{13}X + p_{23}Y + p_{33}Z + p_{43}$ then solving for the elements of $\mathcal{P}$ would reduce to 3 sets of linear equations. Using spatial features, rather than point features, enables us to measure a quantity that is proportional to z for each feature. We call this quantity the local scale factor. The local scale factor is equal to the focal length of the camera divided by the depth of the feature ($s_y f/z$). We use a circular fiducial as our spatial features [Landau, 1987, Thomas and Chan, 1989, Hussain and Kabuka, 1990, Chaudhuri and Samanta, 1991, Safaee-Rad *et al.*, 1992]. The exact nature of these fiducials will be discussed in Chapter 5. In essence, by using a spatial feature we are able to recover $2\frac{1}{2}$D information from a single video image. The idea of using spatial features is not new, however our use of the information provided by spatial features is. We will modify our matrix equation slightly by multiplying I and $\mathcal{P}$ by $\frac{1}{s_y f}$. Since I is expressed in homogeneous coordinates, multiplying I and/or $\mathcal{P}$ by an arbitrary constant has no effect on the solution ($\frac{1}{s_y f}\mathcal{P}$ and $\mathcal{P}$ represent the same solution). We will refer to the elements of $\frac{1}{s_y f}\mathcal{P}$ as p_{ij} and define $\mathcal{I}' = \frac{1}{s_y f}\mathcal{I}$. $\mathcal{I}'$ is a matrix of image points of the form $[x^\star\ y^\star\ 1/s]$. s is the local scale factor at the image point. The pixel coordinates of an image point $[x'\ y']$ are determined by $x' = sx^\star$ and $y' = sy^\star$. $x^\star$, $y^\star$ and $1/s$ are similar to the homogeneous image coordinates defined in (4.1). The elements of $\mathcal{P}$ can be solved for using the following three sets of linear equations and as few as four spatial features.

$$x_i^\star = (p_{11}X_i + p_{21}Y_i + p_{31}Z_i + p_{41}) \tag{4.8}$$

$$y_i^\star = (p_{12}X_i + p_{22}Y_i + p_{32}Z_i + p_{42}) \tag{4.9}$$

$$1/s_i = (p_{13}X_i + p_{23}Y_i + p_{33}Z_i + p_{43}) \tag{4.10}$$

Where $x_i^\star$, $y_i^\star$ and $1/s_i$ are the components of the i^{th} image point and X_i, Y_i and Z_i are the components of the i^{th} model point.

4.3 Implementation

It should be noted that by using a spatial feature the problem of determining the relationship between model coordinates and image coordinates becomes linear and the solution can be found using as few as four features. Also, since all of the calibration parameters (both intrinsic and extrinsic) are bundled into $\mathcal{P}$ we are not required to make any assumptions about the stability of the intrinsic parameters. This is important in dynamic environments because changes to the focus, aperture and/or zoom will likely be needed during the enhanced reality visualization.

Given a model and an image containing at least four fiducials with known model coordinates it is a relatively straightforward task to solve for $\mathcal{P}$ and generate an enhanced reality image. The basic steps are as follows:

1. Grab an image
2. Locate the fiducials in the image and calculate the local scale factor
3. Establish correspondence between fiducials in the image and fiducials in the model
4. Solve for $\mathcal{P}$
5. Using $\mathcal{P}$, project the model into the image
6. Display the enhanced reality image
7. Repeat

4.3.1 Image Acquisition

720×480 pixel images are acquired using a Pulnix TMC-50 color CCD camera or a Panasonic WV-CD50 monochrome CCD camera with either a 25mm or 16mm lens or a CIDTEK monochrome CID camera with a 29mm lens and a Sun VideoPix frame grabber. Both the cameras and frame grabber are relatively inexpensive commodity items. VideoPix is only capable of grabbing $\sim$4 monochrome or $\sim$1 color frame per second. This severely limits the update rate of the enhanced reality visualization. Furthermore, VideoPix only provides 7^- bits of luminance information. Even though pixel values range from 0 to 255 the actual resolution is less than half of this range. We intend to upgrade to a better camera/frame grabber combination in the future, however the current combination is sufficient to demonstrate our method.

4.3.2 Fiducial Location

The location (centroid) and local scale factor (semi-major axis of the fiducial's image divided by the radius of the fiducial) are calculated using moments. Chapter 5 describes these calculations in detail.

4.3.3 Correspondence

Once an initial correspondence has been established, it should be possible to maintain correspondence by tracking the fiducials. The idea is that if images can be processed fast enough the locations of the fiducials should not change very much. Given the correspondence from the last image, we look for fiducials in the new image within a small region around each fiducial's last location. If exactly one fiducial is found in that region then the correspondence for that fiducial is maintained. If at least some minimum number of correspondences (≥ 4) are maintained, then the fiducials have been successfully tracked. If correspondence is lost or no previous correspondence exists than an initial correspondence must be established. The initial correspondence is performed using a modified version of the alignment method [Huttenlocher, 1988]. The alignment method is modified to use scale information as well as some orientation constraints to significantly prune the search space. The three major constraints used are listed below:

- Each fiducial is visible from only one side. Specifically, the dot product of fiducial's normal and the viewing direction must be negative or the fiducial is definitely not visible.

- The local scale factor s_i establishes the relative depth of the fiducials up to the accuracy of the measurement.

- The local scale factor s_i is used to effectively *unproject* the image point $[x'_i \; y'_i]$. Recall that $x'_i = x^\star_i s_i$ and $y'_i = y^\star_i s_i$. If $\mathcal{C}$ is close to a diagonal matrix or if a reasonable guess exists for at least some of the intrinsic camera parameters,[2] then $x^\star_i$ and $y^\star_i$ can be treated as the x and y components of the camera centered coordinates for the i^{th} point. Since scaling is not allowed between world coordinates and camera centered coordinates any transformation between the two must have a scale factor close to unity.

The alignment method uses triples of model and image points to generate possible transformations from model to image coordinates. There are a total of $\binom{m}{3}\binom{i}{3}3!$ different sets of triples where m is the number of model points and i is the number of image points. In general the alignment method produces 2 solutions for every set of model and image points. This is because the alignment method assumes orthographic projection and is therefore unable to resolve reflections about the xy plane. For an image and a model both containing 7 points the alignment method generates $\approx 15,000$ possible solutions. Utilizing the constraints listed above significantly reduces this number. For 20 random views of an object with 7 fiducials, the number of possible solutions was reduced from 15,000 to an average of 100. For one of the views the constraints reduced the number of possible solutions to 3. Some of these views are shown in Chapter 7. The constraints are applied using only the set of three image and model points and before the remaining model points are transformed or global constraints are checked. This reduces the computational cost of establishing correspondence for an object with 7 fiducials by over 2 orders of magnitude.

[2] In our experience only x_0 and y_0 need to be estimated and it is sufficient to use the geometric center of the image plane as a fixed estimate of their values.

Pseudo code for establishing correspondence follows:

1. If correspondences $\geq$ minimum $\Rightarrow$ done.
2. If $0 <$ correspondences $<$ minimum $\Rightarrow$ establish the required additional correspondences.
3. Otherwise $\Rightarrow$ establish correspondences.
   (a) Find candidate transformations from model to image points.
      i. Take each possible triple of model points and pair it with each permutation of three image points.
      ii. For each group of model and image points check that the model points are consistent and determine from which side they are visible. P_i and n_i are the location and normal of the i^{th} model point in the triple.
         - Find the normal to the triple

$$n_{\text{p}} = (P_2 - P_1) \times (P_3 - P_1)$$

         - Verify that the model points can be visible simultaneously

$$\text{SIGN}(n_{\text{p}} \times n_1) = \text{SIGN}(n_{\text{p}} \times n_2) = \text{SIGN}(n_{\text{p}} \times n_3)$$

      iii. Transform (rotate and translate) the model points of the triple so that the first point is at the origin and the points lie in the xy plane. There are two transformations which will accomplish this, choose the one which will make the model points right side up as determined in Step 3(a)ii. Call the transformed model points $M^\star$.
      iv. Unproject the image points of the triple and translate so that the first point is at the origin. Call the transformed image points $I^\star$.
      v. Calculate the transformation(s) $\mathcal{X}$ which maps $M^\star$ to $I^\star$ using the alignment method.
      vi. Check that the solution computed in Step 3(a)v is consistent.
         - Use relative depth constraints to eliminate one of the two solutions.
         - Verify that the solution does not turn the model points upside down. Step 3(a)ii ensured that the model points were right side up. As long as the transformation calculated in Step 3(a)v does not rotate more than 90° about any axis in the xy plane the model points will remain right side up. $\vec{z}$ is the unit vector in the z direction.

$$\text{SIGN}(\|\vec{z}\mathcal{X}\|) > 0$$

- Verify that the scale factor is close to unity

vii. If the transformation computed in Step 3(a)v is consistent, compose the total transformation using the transformations from Steps 3(a)iii and 3(a)v and the translation from Step 3(a)iv.

(b) For each consistent transformation determine correspondences between the remaining image and model points and calculate the cost.

i. Transform the remaining model points into camera centered coordinates.

ii. For each transformed model point

A. Find the closest image point

B. If the closest image point is close enough, add the correspondence to the match and add the Euclidean distance squared to the total cost.

iii. Return a match consisting of a list of correspondences and the total cost.

(c) Consolidate matches and find the best one.

i. For each unique list of correspondences create a consolidated match consisting of:

- The list of correspondences.
- The number of matches containing the list of correspondences.
- The minimum cost among the matches containing the list of correspondences.

ii. Return the best consolidated match which is the one with the largest number of correspondences or the largest number of matches or the lowest cost, in that order.

The algorithm used in Step 2 to establish partial correspondences is very similar to that described in Step 3. If less then three correspondences exist, additional pairs of model and image points are combined with the existing correspondences to form sets of three model and three image points. If three or more correspondences exist then triples of the established correspondences are used to form the sets. The rest of the algorithm is unchanged. The ability to perform partial correspondence greatly simplifies the problem of reestablishing correspondence when most but not all of the fiducials are temporarily occluded. If at least the minimum number of correspondences are maintained partial correspondence is not used to find correspondences for fiducials that were occluded but are now visible. This case is handled nicely as part of locating the fiducials described in Chapter 5.

4.3.4 Solving for $\mathcal{P}$

Equations (4.8), (4.9) and (4.10) are used to solve for the elements of $\mathcal{P}$. If correspondences have been established for more than four fiducials than an over-determined system of linear equations exists. We solve this problem in a least-squares fashion by minimizing the following error terms using Householder's QR decomposition [Watkins, 1991].

$$\|r_1\|_2 = \sum_{i=1}^{n} |x_i^\star - (p_{11}X_i + p_{21}Y_i + p_{31}Z_i + p_{41})|^2 \tag{4.11}$$

$$\|r_2\|_2 = \sum_{i=1}^{n} |y_i^\star - (p_{12}X_i + p_{22}Y_i + p_{32}Z_i + p_{42})|^2 \tag{4.12}$$

$$\|r_3\|_2 = \sum_{i=1}^{n} \left|\frac{1}{s_i} - (p_{13}X_i + p_{23}Y_i + p_{33}Z_i + p_{43})\right|^2 \tag{4.13}$$

Householder's method exhibits good stability and is relatively efficient. Computing the QR decomposition requires approximately $nm^2 - m^3/3$ flops and using the decomposition to solve for the unknowns requires approximately $2nm - m^2/2$ flops where n is the number of unknowns and m is the number of equations. Splitting the problem into three sets of equations has a significant computational advantage. Equations (4.8), (4.9) and (4.10) can be rewritten in matrix form as follows:

$$X^\star = M\mathcal{P}_1 \tag{4.14}$$

$$Y^\star = M\mathcal{P}_2 \tag{4.15}$$

$$S = M\mathcal{P}_3 \tag{4.16}$$

$X^\star$, $Y^\star$ and S are column vectors whose i^{th} components are $x_i^\star$, $y_i^\star$ and $1/s_i$ respectively. M is a $m \times 4$ matrix whose i^{th} row is the i^{th} model point $[X_i\ Y_i\ Z_i\ 1]$. $\mathcal{P}_i$ is the i^{th} column of $\mathcal{P}$. Matrix M is decomposed into the matrices Q and R. Since M is common to all three equations we need only perform the decomposition once. We can simply reuse the decomposition for the remaining equations. In essence, you pay for solving (4.14) and you get the solutions to (4.15) and (4.16) for very little. Formulating the problem as three sets of linear equations each with 4 unknowns reduces the complexity of computing the decomposition by an order of magnitude and solving for the unknowns by half an order of magnitude compared to solving a single set of linear equations with 12 unknowns. In fact the QR decomposition need not be recomputed until the correspondences change, further increasing the computational savings.

4.3.5 Creating the Enhanced Reality Image

Creating the enhanced reality image consists of two steps: making a virtual image (rendering the model) and combining the virtual image and the raw image. $\mathcal{P}$ is used to project the model into an initially empty virtual image. Currently, models consist of either a collection of points or a collection of line segments. Points are projected by multiplying by $\mathcal{P}$ and then rounding off to the nearest pixel. Line segments are handled by using $\mathcal{P}$ to project the endpoints into the virtual image and then drawing a line between them. No anti-aliasing or z-buffering is performed. As a result, some edges are jagged and some points which perhaps should not be visible are. Once the virtual image has been generated it must be combined with the raw image to form the enhanced reality image. Pixels in the virtual image have precedence over pixels in the raw image. If a pixel in the virtual image is nonzero (zero being no information) then its value is placed in the corresponding location in the enhanced reality image. If a pixel in the virtual image is zero then the corresponding pixel in the raw image is used. Both the rendering and combination routines are a bit simplistic, but are sufficient to demonstrate our method. Rendering and combination are important problems, however they are not the focus of this work.

4.3.6 Displaying the Enhanced Reality Image

The enhanced reality image is displayed as a single image on a high resolution CRT using the X window system. The size of the image to be displayed and whether it is to be displayed on the local machine greatly affects the time it takes to display an image. In the current implementation, about 20% of the computation time is spent simply putting a half sized enhanced reality image on the screen. Creating a stereo display or using a video see-through HMD are straightforward extensions of our method.

4.3.7 Discussion

The current system is implemented in Lucid Common Lisp and runs on a SparcStation 2. Currently, the two most limiting components are the frame grabber and the rendering/display system. Depending on the complexity of the model, the renderer may require a minute or more to perform the rendering. Using a simple model, frame rates of $\sim$2hz can be achieved. Table 4.1 shows a break down of the computational time required for the major functions. Low end SGI machines such as the Indy are capable of grabbing a full size color image and displaying it on the screen at $>$ 30hz. The SGI machines are also capable of rendering a fairly complex model and displaying it at $>$ 30hz. If these times are substituted for frame grabbing, rendering and displaying a frame rate of $\sim$10hz results. The frame rate would be $\sim$20hz if the time requirements for frame grabbing, rendering and displaying could be eliminated entirely. Little effort has been put into optimizing the current implementation for speed. Recoding

Function	Time Required	
Grab Frame	0.25s	50.3%
Find Fiducials and Calculate Centroid and Local Scale Factor	0.03s	6.1%
Check Correspondences	0.007s	1.4%
Calculate Perspective Transformation	0.01s	2.0%
Render Model and Create Enhanced Reality Image	0.1s	20.1%
Display Image	0.1s	20.1%

Table 4.1: Time required for major functions running on a SparcStation 2 in Common Lisp using a simple model and displaying a half sized enhanced reality image.

some portions and using a C-30 digital signal processing board to grab and process images should produce significant improvements in speed. We believe frame rates of $>$30hz should be achievable with this modest hardware.

Assuming that we are given a model to be used in creating the enhanced reality image is not unreasonable. In fact, it is a fundamental assumption of enhanced reality visualization. The basic idea is to add information to an image. This information in most cases is not visible from the current view point and must come from some source other than the raw image (typically the model). This is not to say that constructing a model is easy. Model building is simply not the focus of this work. Assuming that we know the model coordinates of the fiducials in some cases is unreasonable. This amounts to assuming that the fiducials are part of the model. In Chapter 7 we present enhanced reality visualizations of a test object and a plastic skull. The model for the test object is a CAD-like model and the fiducials are part of it. This is not the case for the skull. Here, the model is a CT scan of the skull. The fiducials are not present in the CT data and are not part of the model. In this case, we need a method of determining the model coordinates of the fiducials. Chapter 6 presents the details of determining the model coordinates for the fiducials used on the skull.

Chapter 5

Feature Detection and Localization

In the last chapter we described the theory behind our method. The success of any method for enhanced reality visualization is inseparably tied to the accuracy of the data used to determine to transformation which maps model coordinates to image coordinates. In this chapter we will discuss the practical details of finding fiducials and the accuracy with which their position and local scale factor can be determined.

5.1 Details

The circular fiducials used in our method are detected using pattern matching and are localized using moment calculations. An actual size fiducial is shown in Figure 5-1. A chord passing near the center of the fiducial will exhibit transitions from light to dark, dark to light, light to dark and dark to light. Figure 5-2 shows a blow-up of an image of a fiducial and the intensity profile of a chord line. Constraints such as the steepness of the transition, the length of the transition and the separation between transitions eliminate nearly all detections which do not come from actual fiducials. By checking the rows and columns of an image for collocated occurrences of this transition pattern the presence of an fiducial can be further validated and a rough position can be efficiently found. The time required is linear in the size of the image. In addition, a bounding box (x_1, x_2, y_1 and y_2) and an upper and lower threshold (t_{upper} and t_{lower}) for each fiducial can be readily obtained from this process. The bounding box, with vertices $[x_1\ y_1]$, $[x_1\ y_2]$, $[x_2\ y_1]$ and $[x_2\ y_2]$, is slightly larger than the smallest rectangle aligned with the axes which can contain the fiducial. The upper and lower thresholds are used to rescale the pixel values. These values are needed because we use grey scale moments to find the centroid and local scale factor of the fiducial. The local scale factor is the semi-major axis of the fiducial's image divided by the radius of the fiducial. The bounding box and thresholds for one fiducial are completely independent from those of the other fiducials. This produces an extremely robust detection and localization

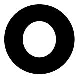

Figure 5-1: Actual size fiducial.

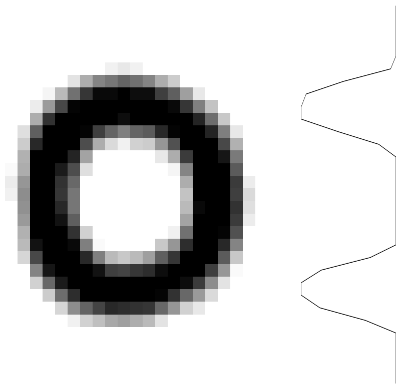

Figure 5-2: Enlarged image of a fiducial with pixel values for a chord shown to the side.

algorithm. For example, large gradients in average image intensity, such as might be caused by shadows, have little effect on detecting and localizing the fiducials.

Fiducials are detected by looking for occurrences of intensity profiles such as the one shown in Figure 5-2. The location and the maximum and minimum values of these profiles are used to determine a bounding box and an upper and lower threshold for the fiducial. Once a fiducial has been detected moments are used to calculate the centroid and local scale factor of the fiducial. Detailed pseudo code used to locate fiducials follows:

1. Grab a fresh image.
2. For each fiducial present in the last image:
   (a) Define a window centered at the last location $[x'_{\text{old}}\ y'_{\text{old}}]$ with dimensions equal to $2kr$. k is a window size scale factor and r is equal to the fiducial's local scale factor s_i in the last image times the fiducial's radius in model coordinates.[1]
   (b) Scan the window for fiducials.
      i. For each horizontal and vertical scan line in the region collect possible fiducial detects.
         A. Initialize the following parameters: l_1 through l_4 (location of transition 1–4), $l_{\min}$ and $l_{\max}$ (the minimum and maximum

[1] $s_{x/y}$ is the ratio of pixel spacing in the x and y directions (s_x/s_y). This quantity is used to correct for the fact that pixels generally are not square. The x dimension of the window must be scaled by $1/s_{x/y}$.

separation between transitions), w_1 through w_4 (width of transition 1–4), $w_{\max}$ (the maximum width of a transition), s_1 through s_4 (slope of transition 1–4), $s_{\min}$ (the minimum intensity change to be considered above noise), t_{upper} and t_{lower} (upper and lower thresholds).

B. Scan across or down the scan line until the change in pixel intensity $< -s_{\min}$. Scan back several pixels, take the minimum value and if it is less than t_{upper} store it in t_{upper}. Update l_1, w_1 and s_1.

C. Continue scanning until the change in pixel intensity is no longer $< -s_{\min}$. Scan ahead several pixels, take the maximum value and if it is less than t_{lower} store it in t_{lower}. Update l_1, w_1 and s_1. If $w_1 > w_{\max}$ go to Step 2(b)iA.

D. Continue scanning until the change in pixel intensity is $> s_{\min}$. Scan back several pixels, take the maximum value and if it is greater than t_{lower} store it in t_{lower}. Update l_2, w_2 and s_2.

E. Continue scanning until the change in pixel intensity is no longer $> s_{\min}$. Scan ahead several pixels, take the minimum and if it is less than t_{upper} store it in t_{upper}. Update l_2, w_2 and s_2. If $l_{\min} \leq l_2 - l_1 \leq l_{\max}$ or $s_1 \not\approx s_2$ go to Step 2(b)iA.

F. Repeat Steps 2(b)iB and 2(b)iC except update l_3, w_3 and s_3. If $w_3 > w_{\max}$ or $2(l_2 - l_1) \not\approx (l_3 - l_2)$ or $s_3 \not\approx s_2 \approx s_1$ go to Step 2(b)iA.

G. Repeat Steps 2(b)iD and 2(b)iE except update l_4, w_4 and s_4. If $w_4 > w_{\max}$ or $(l_4 - l_3) \not\approx 2(l_2 - l_1) \approx (l_3 - l_2)$ or $s_4 \not\approx s_3 \approx s_2 \approx s_1$ go to Step 2(b)iA.

H. Add the detection (location, size and thresholds) to a list of detections.

I. Go to Step 2(b)iA

ii. Consolidate detections. Detections from adjacent scan lines are combined if they overlap by at least 50%. Detections from orthogonal scan lines are combined if the detections intersect. All consolidations retain the maximum bounding box, the minimum t_{upper} and the maximum t_{lower}.

iii. Expand the bounding box as necessary to ensure the fiducial is fully enclosed. This is required because if the fiducial is elliptical in shape and is at an angle to the x and y axes, the bounding box generated by the detections may under estimate the size of the fiducial.

(c) Calculate the 0^{th}, 1^{st} and 2^{nd} moments of inertia as well as the Euler number of the window using grey scale values.

(d) If the Euler number is zero return the the location and local scale factor $[x' \; y' \; 1/s]$ for the fiducial.

3. If there was a valid solution for the last image, predict the location and size of each model point for which a correspondence did not exist using this solution. Perform Steps 2a through 2d.

4. If the number of correspondences maintained in Step 2 plus the number established in Step 3 is less than the minimum, scan the entire image for fiducials and establish correspondences for any new fiducials found using the algorithm presented in Section 4.3.3.

The zeroth, first and second order moments of a region bounded by x_1, x_2, y_1 and y_2 can easily be calculated using the following formulas:

$$m_0 = \sum_{x=x_1}^{x_2} \sum_{y=y_1}^{y_2} \rho(x,y) \tag{5.1}$$

$$m_{\mathrm{x}} = \sum_{x=x_1}^{x_2} \sum_{y=y_1}^{y_2} \rho(x,y)x \tag{5.2}$$

$$m_{\mathrm{y}} = \sum_{x=x_1}^{x_2} \sum_{y=y_1}^{y_2} \rho(x,y)y \tag{5.3}$$

$$m_{\mathrm{x}^2} = \sum_{x=x_1}^{x_2} \sum_{y=y_1}^{y_2} \rho(x,y)\left(y^2 + i_{\mathrm{x}}\right) \tag{5.4}$$

$$m_{\mathrm{xy}} = \sum_{x=x_1}^{x_2} \sum_{y=y_1}^{y_2} \rho(x,y)\left(xy + i_{\mathrm{xy}}\right) \tag{5.5}$$

$$m_{\mathrm{y}^2} = \sum_{x=x_1}^{x_2} \sum_{y=y_1}^{y_2} \rho(x,y)\left(x^2 + i_{\mathrm{y}}\right) \tag{5.6}$$

x and y are image coordinates and $\rho(x,y)$ is a weight based on the value $v(x,y)$ of the pixel at image coordinates x, y. i_{x}, i_{xy} and i_{y} are the moments of inertia for an individual pixel about the center of the pixel. The weights are determined by the following function.

$$\rho(x,y) = \begin{cases} 0 & \text{if } v(x,y) \geq t_{\mathrm{upper}} \\ 1 & \text{if } v(x,y) \leq t_{\mathrm{lower}} \\ \frac{t_{\mathrm{upper}} - v(x,y)}{t_{\mathrm{upper}} - t_{\mathrm{lower}}} & \text{otherwise} \end{cases} \tag{5.7}$$

The Euler number of the region can be used to verify that only a single object with a single hole is present. By using an upper and lower threshold we can ensure that noisy pixels which are not on the fiducial do not contribute to the moment calculations and that noisy pixels which are entirely on the fiducial contribute fully. Euler numbers can also be used to verify that the thresholds have been properly chosen. From (5.1) through (5.6) the centroid of the region and the moment about the axis of greatest inertia can be calculated.[2]

$$\begin{aligned}
\bar{x} &= s_{\mathrm{x/y}} \frac{m_{\mathrm{x}}}{m_0} && (5.8)\\
\bar{y} &= \frac{m_{\mathrm{y}}}{m_0} && (5.9)\\
I_{\max} &= \left(m_{\mathrm{x}^2} - m_0 \bar{y}^2\right) \sin^2\theta + 2\left(s_{\mathrm{x/y}} m_{\mathrm{xy}} - m_0 \bar{x}\bar{y}\right) \cos\theta \sin\theta + \\
&\quad \left(s^2_{\mathrm{x/y}} m_{\mathrm{y}^2} - m_0 \bar{x}^2\right) \cos^2\theta && (5.10)\\
\theta &= \frac{1}{2} \arctan\left(\frac{2 m_{\mathrm{xy}}}{s_{\mathrm{x/y}} m_{\mathrm{y}^2} - m_{\mathrm{x}^2}/s_{\mathrm{x/y}}}\right)
\end{aligned}$$

It is well known that the perspective projection of a circle is an ellipse. The semi-major axis of the ellipse can be calculated using the following equation:[3]

$$a = \sqrt{\frac{4 I_{\max}}{m_0 \left(1 + r_{\mathrm{i}}^2/r_{\mathrm{o}}^2\right)}} \tag{5.11}$$

For now we will assume that orthographic projection is a reasonable model for the area immediately surrounding a fiducial, see Figure 5-3. Later we will consider the error introduced by this assumption, Figure 5-4. This error is sometimes referred to as perspective distortion. Given this assumption, the centroid of the circle C_{c} projects onto the centroid of the ellipse C_{e} and the diameter d' projects onto the major axis of of the ellipse, a'. The diameter d' is parallel to the image plane so it is not foreshortened. The following equations relate the fiducial to its projection.

$$\begin{aligned}
a' &= \frac{f}{z^\star} d' \Longrightarrow s = \frac{a'}{d'} = \frac{f}{z^\star} && (5.12)\\
\bar{x} &= s x^\star = x' && (5.13)\\
\bar{y} &= s y^\star = y' && (5.14)
\end{aligned}$$

It should be noted that s, x', y', $x^\star$ and $y^\star$ are the same parameters as in (4.8) through (4.10). x', y' and s can be easily calculated requiring time linear in the number fiducials and their size. At first, the need to use $s_{\mathrm{x/y}}$ in recovering s

[2]The quantity shown for $I_{\max}$ should actually be multiplied by an additional factor of $s_{\mathrm{x/y}}$. We have omitted it for simplicity sake because it cancels with the same factor for m_0 in (5.11).

[3]Our fiducials have holes in them and the additional factor of $(1 + r_{\mathrm{i}}^2/r_{\mathrm{o}}^2)$ in the denominator corrects for this. r_{i} is the inner radius and r_{o} is the outer radius.

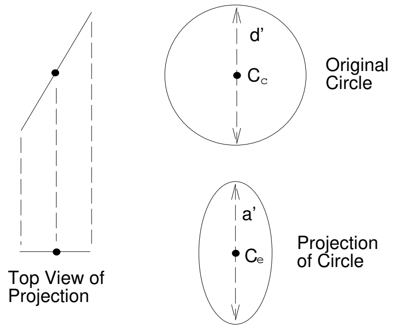

Figure 5-3: Orthographic projection of a circle.

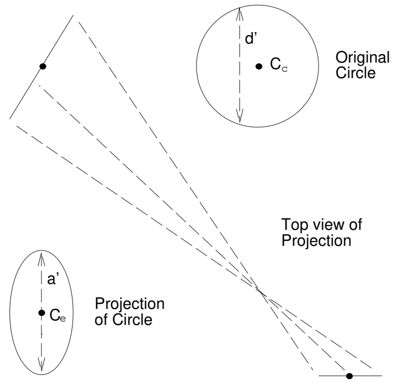

Figure 5-4: Perspective projection of a circle.

would appear to a significant limitation. This is not the case because $s_{x/y}$ is easy to calibrate and it does not change [Lenz and Tsai, 1988, Penna, 1991]. $s_{x/y}$ is a function of the aspect ratio of the image sensor and the ratio of camera and frame grabber clock frequencies. The physical properties of the image sensor cannot change and modern clocks have extremely stable frequencies. Therefore it is very reasonable to calibrate $s_{x/y}$ once and then forget about it. $s_{x/y}$ could also be determined via self-calibration removing any burden to the user.

5.2 Error Analysis

There are several sources of error associated with processing digital images. One of the more significant sources is quantization errors [Kamgar-Parsi and Kamgar-Parsi, 1989]. These errors are the result of taking a continuous signal and converting it to digital values. First, we will examine errors caused by the fact that pixel values are only available at discrete locations in a lattice. Consider a row of pixels such as those shown in Figure 5-5. The grid represents the pixel lattice. Pixels have just two states, on and off, with shaded squares representing on pixels. Using the centroid calculation described above both rows have the same centroid. The maximum error in the horizontal position of the centroid is 0.5 pixels. In the vertical direction the maximum is also 0.5 pixels so the maximum distance between the calculated centroid and the actual centroid is $1/\sqrt{2}$.

The maximum error can be reduced significantly by using a circular shape [Bose and Amir, 1990, Efrat and Gotsman, 1993]. Figure 5-7 shows a digital approximation of a circle. The improvement which results from using a circular shape is caused by the fact that the error for a given row or column is dependent

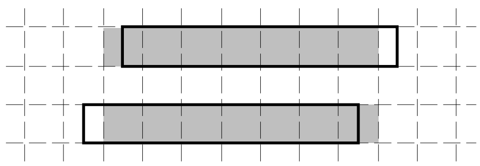

Figure 5-5: The effect of quantization errors on the centroid of a row of pixels.

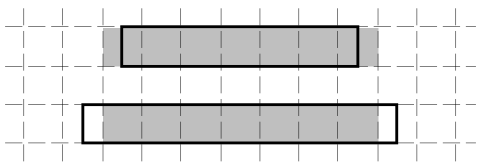

Figure 5-6: The effect of quantization errors on the length of a row of pixels.

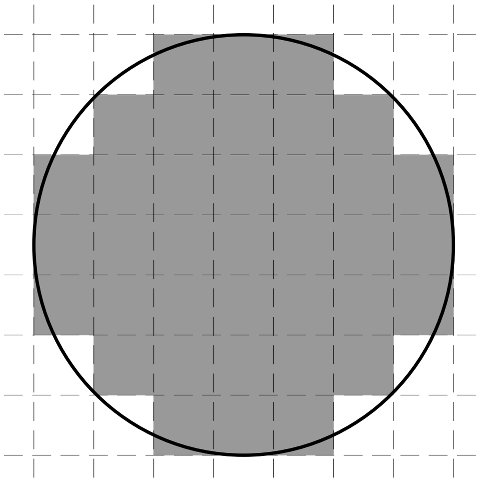

Figure 5-7: A digital approximation of a circle.

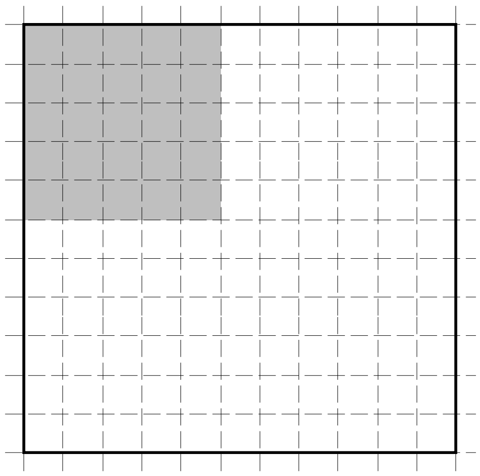

Figure 5-8: Model for grey scale pixel values.

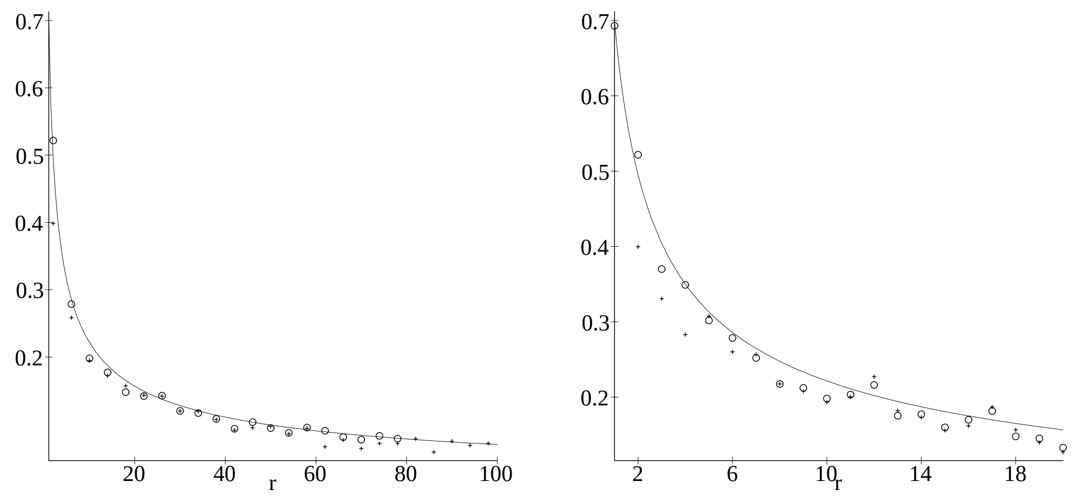

Figure 5-9: Error in the centroid of a circular fiducial. Error is the distance in pixels between the actual centroid and that of a digital approximation. The radius is also expressed in pixels.

upon the errors of the other rows or columns. In short, the errors tend to cancel out. For a circle of radius r and centroid $[x_0\ y_0]$ the maximum error in the calculated centroid $\mu(r)$ is given by the following expression.

$$\mu(r) = \max_{x_0\ y_0\ dr} \left(\| [x_0\ y_0] - \text{CENTROID}\,(x_0, y_0, r, dr) \| \right) \tag{5.15}$$

CENTROID() is a function which calculates the centroid of a digital approximation of a circle using (5.1) through (5.3), (5.8) and (5.9). $\rho(x, y)$ is replaced with INSIDE?() which returns a 1 if $(x - x_0)^2 + (y - y_0)^2 \leq (r + dr)^2$ otherwise it returns 0. Figure 5-9 shows the maximum error in the centroid calculation $\mu(r)$. The circles are the result of evaluating (5.15) for $0.0 \leq x_0, y_0, dr \leq 1.0$ with 0.01 increments. The crosses are the result of a stochastic sampling method to find the maximum over the same region. $1/\sqrt{2r}$ is also plotted on the axes. The curve fits the data well and is a good estimate of $\mu(r)$.

The radius calculations described above are subject to errors similar to those seen for the centroid. The maximum error in the length is 1 pixel, see Figure 5-6. Using a circular shape will also reduce the error in the calculated radius for the same reasons as above. For a circle of radius r and centroid $[x_0\ y_0]$ the maximum error in the calculated radius $\nu(r)$ is given by the following expression.

$$\nu(r) = \max_{x_0\ y_0\ dr} \left(\| r - \text{RADIUS}\,(x_0, y_0, r, dr) \| \right) \tag{5.16}$$

RADIUS() is a function which calculates the radius of a digital approximation of a circle using (5.1) through (5.6), (5.8) through (5.10) and (5.11). Again,

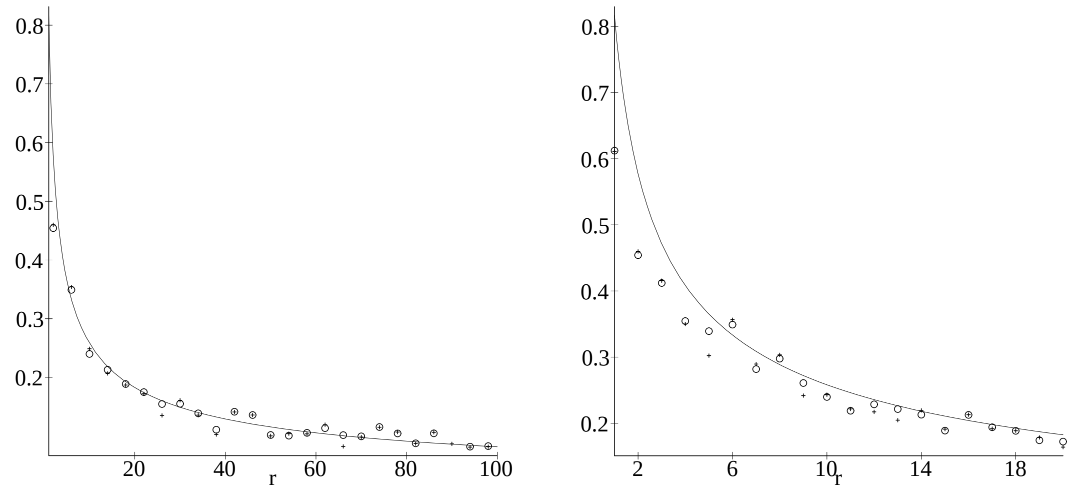

Figure 5-10: Error in the radius of a circular fiducial. Error is the difference in pixels between the actual radius and that of a digital approximation. The radius is also expressed in pixels.

$\rho(x, y)$ is replaced with INSIDE?() which returns a 1 if $(x - x_0)^2 + (y - y_0)^2 \leq (r + dr)^2$ otherwise it returns 0. Figure 5-10 shows the maximum error in the radius calculation $\nu(r)$. The circles are the result of evaluating (5.16) for $0.0 \leq x_0, y_0, dr \leq 1.0$ with 0.01 increments. The crosses are the result of a stochastic sampling method to find the maximum over the same region. $\sqrt{2/3r}$ is also plotted on the axes.[4] The curve fits the data well and is a good estimate of $\nu(r)$.

The maximum error for both the centroid and radius can be further reduced by using grey scale values rather than binary values [Chiorboli and Vecchi, 1993]. Grey scale values can be modeled as the sum of a number of binary sub-pixels. Figure 5-8 shows a pixel with a dynamic range of 122 and a value of 25. This effectively increases the pixel resolution by the square root of the dynamic range, $\sqrt{t_{\text{upper}} - t_{\text{lower}} + 1}$. This increase in the resolution increases the effective radius of the circle.

Another class of quantization error is caused by the fact that pixel values are the result of a spatial process. The value of a particular pixel is not the intensity at some infinitesimal point, rather it is the average intensity within the area of the pixel. Figure 5-8 shows a model of a grey scale pixel. If the shaded and unshaded portions of the pixel represents maximum and minimum intensity

[4]The factor of $\sqrt{2/3}$ results because (5.11) assumes a elliptical shape and our digital approximations are not truly ellipses. This error is most pronounced at small radii however some error will always be present.

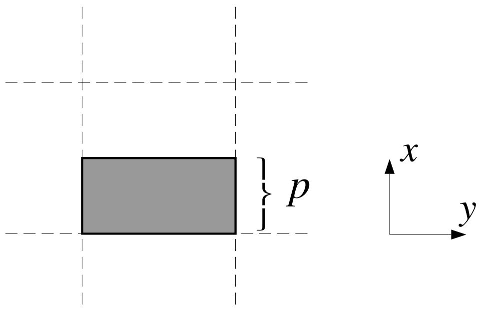

Figure 5-11: A pixel partially covered by a larger figure.

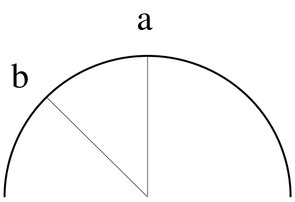

Figure 5-12: Effect on a circular figure.

respectively, then the pixel has a value of 0.2 or 25 on a scale from 0 to 121. If the shaded portion is produced by a larger rectangular figure for which we are calculating moments, the correct values for x and y to use in (5.2) through (5.6) are the x and y components of the centroid of the shaded region. The centroid of the shaded region is not the center of the pixel, however (5.2) through (5.6) assume that it is, in essence treating the pixel as if it were homogeneous. The fact that the centroid of the region which produces a pixel's value may not be the center of the pixel introduces error into the moment calculations. Figure 5-11 shows a pixel only partially covered by a larger figure. p is fraction of the pixel covered by the larger figure. The calculated and actual contribution to the first moment, $m_{1_{\text{calculated}}}$ and $m_{1_{\text{actual}}}$, as well as their difference, $\triangle m_1$, are given by the following equations:

$$m_{1_{\text{calculated}}} = py \tag{5.17}$$

$$m_{1_{\text{actual}}} = p\left(y - \frac{1-p}{2}\right) \tag{5.18}$$

$$\begin{aligned}\triangle m_1 &= m_{1_{\text{calculated}}} - m_{1_{\text{actual}}} \\ &= p\,(1-p)\,/2 \end{aligned} \tag{5.19}$$

A maximum $\triangle m_1$ of 1/8 occurs when $p = 1/2$. $\triangle m_1$ cannot be negative therefore the maximum error results when $\triangle m_1 = 1/8$ along one side of the figure and $\triangle m_1 = 0$ along the other. We will assume that $\triangle m_1 = 1/8$ along the half circle shown in Figure 5-12.[5] The circle has a radius of r and $\triangle m_1$ is towards the

[5]This assumption overestimates the error on two counts. First $\triangle m_1$ cannot equal 1/8 everywhere along the hemi-circle. Second $\triangle m_1$ assumes that the partial figure is aligned with the pixel grid. If it is not, the maximum error is reduced.

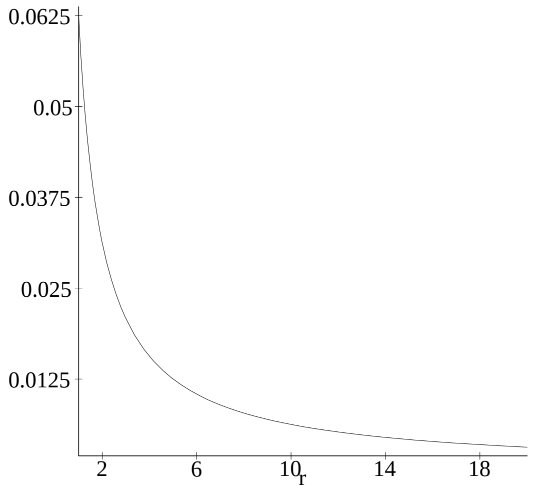

Figure 5-13: Error in the centroid resulting from the homogeneous assumption. Both the error and radius are expressed in pixels

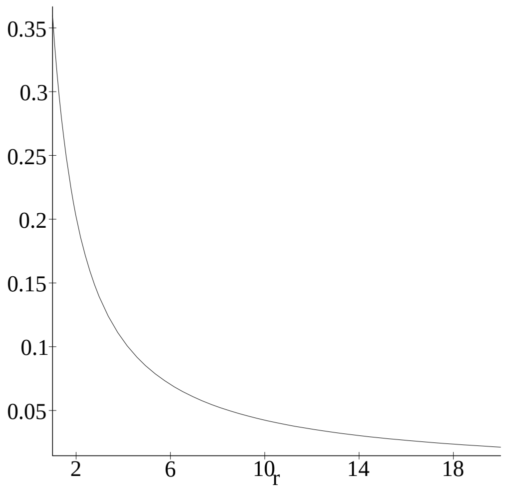

Figure 5-14: Error in the radius resulting from the homogeneous assumption. Both the error and radius are expressed in pixels.

center of the circle producing the following expression for the contribution to the y component of the centroid:

$$\triangle m_{1_y}(x) = \frac{\sqrt{r^2 - x^2}}{8r}. \tag{5.20}$$

Integrating this expression from $-r$ to r and dividing by the area results in a maximum error in the centroid of $\triangle \bar{y}_{\max} = 1/16r$ as shown in Figure 5-13.

An analysis of the error in the second moment is similar. The calculated and actual contribution to the second moment, $m_{2_{\text{calculated}}}$ and $m_{2_{\text{actual}}}$, as well as their difference, $\triangle m_2$, are given by the following equations:

$$m_{2_{\text{calculated}}} = p\left(y^2 + 1/12\right) \tag{5.21}$$

$$m_{2_{\text{actual}}} = p\left(y - \frac{1-p}{2}\right)^2 + p^3/12 \tag{5.22}$$

$$\begin{aligned} \triangle m_2 &= m_{2_{\text{calculated}}} - m_{2_{\text{actual}}} \\ &= py\,(1-p) - \frac{2p}{12}\left(1 - 3p + 2p^2\right) \end{aligned} \tag{5.23}$$

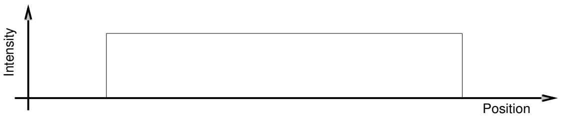

Figure 5-15: The ideal intensity profile for a cross section of a circular disk.

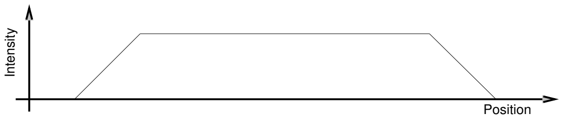

Figure 5-16: The effect of bleeding on the disk in the figure to the left.

The maximum $\triangle m_2$ and the value of p at which it occurs are functions of y. We will assume that $p_{\max} = 1/2$ and $\triangle m_{2_{\max}} = y/4$.[6] The maximum error results when $\triangle m_2 = y/4$ around the entire circumference of the circle. Doubling $\triangle m_2$ and correcting for the orientation produces

$$\triangle m_2(x) \quad = \quad \frac{r^2 - x^2}{2r}. \tag{5.24}$$

Integrating this expression from $-r$ to r and substituting into (5.11) results in a maximum error in the radius of $\triangle r_{\max} = r - \sqrt{r^2 + 8/3\pi}$ as shown in Figure 5-14.

Next we will consider image formation errors. These errors include noise and nonlinearities in the image sensor [Dinstein *et al.*, 1984, Healey and Kondepudy, 1994]. For our purposes the most significant phenomenon is the smoothing of high contrast edges. We will refer to this as bleeding. Figure 5-15 shows the ideal intensity profile for a cross section of a circular disk and Figure 5-16 shows the effect of bleeding on the same disk. We have shown the transition from maximum intensity to minimum intensity as linear. This is almost certainly not the case, however it makes little difference for our analysis. As long as the transition has the same shape all along the circumference of the circle, bleeding has no effect on the centroid. The inertia of the two disks shown in Figures 5-15 and 5-16, however are not the same, therefore the radius calculation is effected by bleeding. In order to explore this effect we will consider the ellipse produced by the following function

$$f(x, y) \quad = \quad \min\left(\max\left(v, 0\right), 1\right) \tag{5.25}$$

$$v \quad = \quad \frac{1 - x^2/a^2 - y^2/b^2}{1 - (a - w)^2/a^2}. \tag{5.26}$$

a and b are the semi-major and semi-minor axis of the ellipse and w is the length of the transition. v was chosen because it is a good approximation to a linear transition and is easily integrable. The ellipse is shown in Figure 5-17. The

[6]Actually $p_{\max} = 1/2 - y \pm \sqrt{y^2 + 1/12}$. This function rapidly approaches an asymptotic value of 1/2. For $y = 4.2$ the actual $\triangle m_{2_{\max}}$ is within 2% of 1/2.

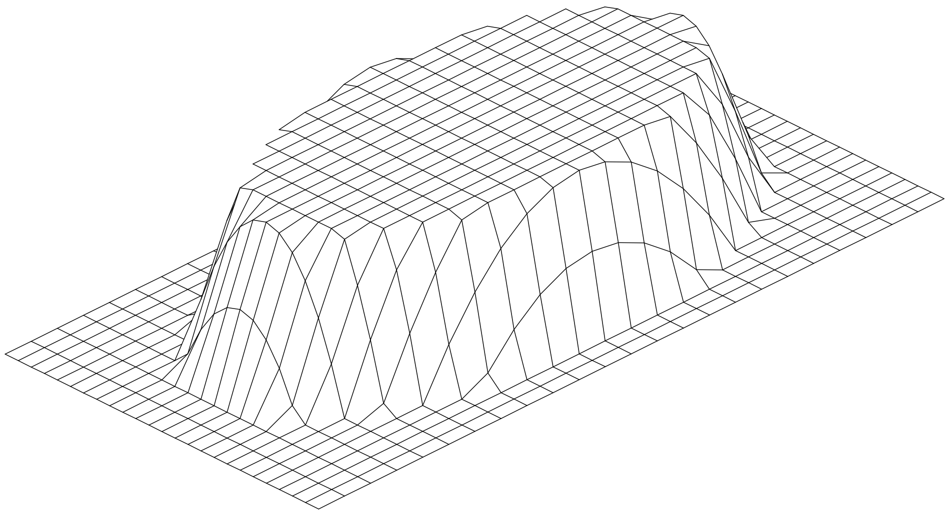

Figure 5-17: Intensity profile for an ellipse.

zeroth and second moments (about the axis of greatest inertia) for $f(x, y)$ are as follows.

$$m_0 = \frac{\pi b}{2a}\left(w^2 - 2aw + 2a^2\right) \tag{5.27}$$

$$m_{2_{\max}} = \frac{\pi b}{12a}\left(w^4 - 4w^3a + 7w^2a^2 - 6a^3w + 3a^4\right) \tag{5.28}$$

We will assume that the actual edge occurs at $a - w/2$ along the major axis. The calculated semi-major axis can be found by substituting m_0 and $m_{2_{\max}}$ into (5.11). The ratio of the actual edge location to the calculated semi-major axis $\eta(a, b, w)$ is a measure of the error introduced by bleeding and is shown below.

$$\eta(a, b, w) = (a - w/2)\sqrt{\frac{3\left(w^2 - 2aw + 2a^2\right)}{2\left(w^4 - 4w^3a + 7w^2a^2 - 6a^3w + 3a^4\right)}} \tag{5.29}$$

Figure 5-22 shows a plot of $\eta(a, b, w)$. The transition lengths we have encountered are typically less than one pixel.

So far in our discussion of errors (with the exception of bleeding) we have considered circles not ellipses. The analysis extends easily to cover ellipses. Two effects are seen as the figure becomes an ellipse. First the effective radius

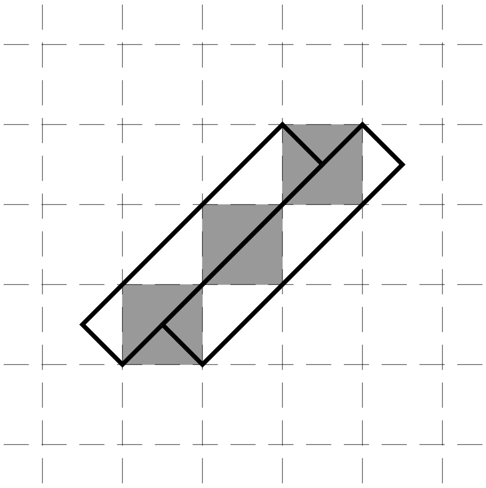

Figure 5-18: The effect of quantization errors on the centroid for a rectangular figure at an angle to the pixel lattice.

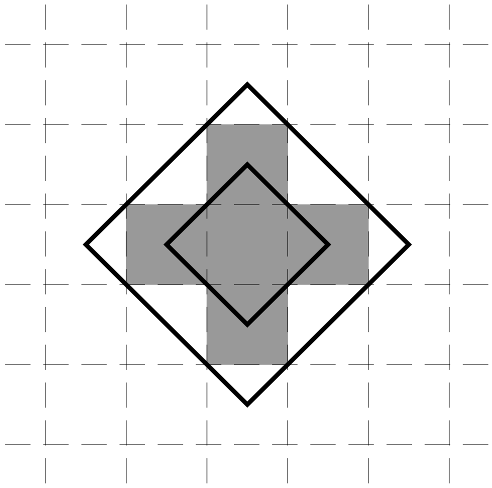

Figure 5-19: The effect of quantization errors on the radius for a rectangular figure at an angle to the pixel lattice.

is the semi-minor axis b. Second, additional error is introduced when the major axis is not aligned with the pixel lattice. Figures 5-18 and 5-19 show two examples of the latter effect. By modifying (5.15) slightly we obtain:

$$\mu(r, b/a) = \max_{x_0\ y_0\ dr\ \theta} (\|[x_0\ y_0] - \text{CENTROID}\,(x_0, y_0, r, dr, b/a, \theta)\|) \tag{5.30}$$

b/a is the ratio of the minor axis to the major axis. CENTROID() and INSIDE?() are modified appropriately to handle ellipses at any angle θ relative to the x axis. Figure 5-20 shows the maximum error in the centroid calculation $\mu(r, b/a)$. A stochastic sampling method was used to find the maximum of (5.30) over the region $0.0 \leq x_0, y_0, dr \leq 1.0$, $0 \leq \theta \leq \pi$ and $r = 10$. $\frac{1+(\sqrt{2}-1)(1-b/a)}{\sqrt{2b}}$ is also plotted on the axes. The curve fits the data well and is a good estimate of $\mu(r, b/a)$. By modifying (5.16) slightly we obtain:

$$\nu(r, b/a) = \max_{x_0\ y_0\ dr\ \theta} (\|r - \text{RADIUS}\,(x_0, y_0, r, dr, b/a, \theta)\|) \tag{5.31}$$

RADIUS() is modified appropriately to handle ellipses at any angle θ relative to the x axis. Figure 5-21 shows the maximum error in the radius calculation $\nu(r, b/a)$. A stochastic sampling method was used to find the maximum of (5.31) over the region $0.0 \leq x_0, y_0, dr \leq 1.0$, $0 \leq \theta \leq \pi$ and $r = 10$. $\frac{1+(\sqrt{2}-1)\sqrt{1-b/a}}{\sqrt{3b/2}}$ is also plotted on the axes. The curve fits the data well and is a good estimate of $\nu(r, b/a)$.

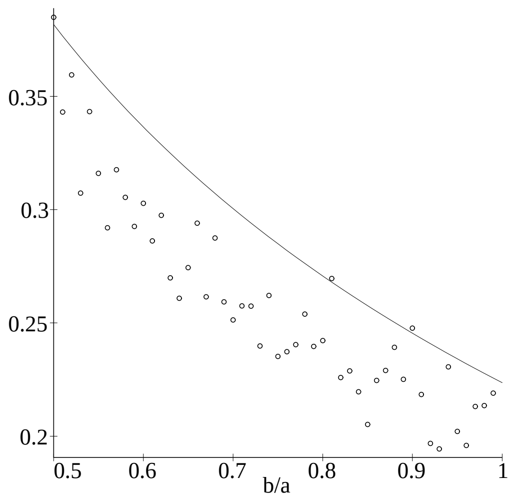

Figure 5-20: Centroid error for an elliptical fiducial with a radius of 10. The error is expressed in pixels and b/a is the ratio of minor and major axis.

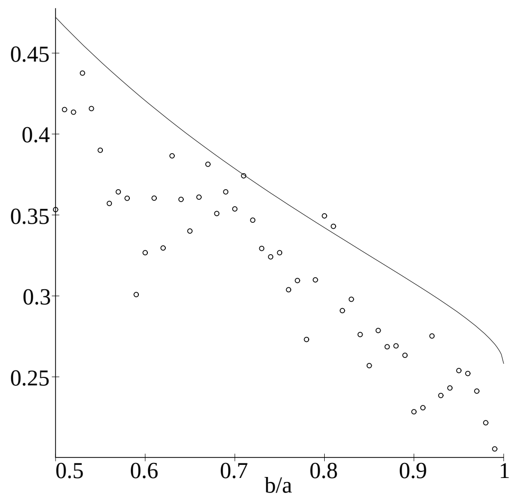

Figure 5-21: Radius error for an elliptical fiducial with a radius of 10. The error is expressed in pixels and b/a is the ratio of minor and major axis.

Finally we will consider the errors introduced by our assumption of orthographic projection for a fiducial. As shown in Figure 5-4, some error is introduced in the centroid as well as the semi-major axis. Consider a plane rotated by an angle of ϕ about an axis passing through $[X_0\ Y_0\ Z_0]$ in camera centered coordinates which is parallel to the x axis. Let $[X\ Y]$ represent a point on the plane and let the origin of the plane be $[X_0\ Y_0\ Z_0]$. Points on the plane project on to the image plane by the following relationships

$$x' = \frac{f\,(X + X_0)}{Y \sin\phi + Z_0} \tag{5.32}$$

$$y' = \frac{f\,(Y\cos\phi + Y_0)}{Y \sin\phi + Z_0}. \tag{5.33}$$

Next, consider a circle in the plane and centered at the origin with a radius of r. We can determine the effects of perspective distortion on the centroid and radius calculation by evaluating following continuous versions of (5.1) through (5.6).

$$m_0 = \int_{-r}^{r} \int_{-\sqrt{r^2-y^2}}^{\sqrt{r^2-y^2}} \delta x' \delta y' \tag{5.34}$$

$$m_{\mathrm{x}} = \int_{-r}^{r} \int_{-\sqrt{r^2-y^2}}^{\sqrt{r^2-y^2}} x' \delta x' \delta y' \tag{5.35}$$

$$m_{\mathrm{y}} = \int_{-r}^{r} \int_{-\sqrt{r^2-y^2}}^{\sqrt{r^2-y^2}} y' \delta x' \delta y' \tag{5.36}$$

$$m_{\mathrm{x}^2} = \int_{-r}^{r} \int_{-\sqrt{r^2-y^2}}^{\sqrt{r^2-y^2}} y'^2 \delta x' \delta y' \tag{5.37}$$

$$m_{\mathrm{xy}} = \int_{-r}^{r} \int_{-\sqrt{r^2-y^2}}^{\sqrt{r^2-y^2}} x' y' \delta x' \delta y' \tag{5.38}$$

$$m_{\mathrm{y}^2} = \int_{-r}^{r} \int_{-\sqrt{r^2-y^2}}^{\sqrt{r^2-y^2}} x'^2 \delta x' \delta y' \tag{5.39}$$

The error in the x component of the centroid α is the difference between the projection of X_0 and m_{x}/m_0. The error in the y component β is defined similarly

$$\alpha = \frac{f Z_0 X_0}{Z_0^2 - r^2 \sin^2\phi} - \frac{f X_0}{Z_0} \tag{5.40}$$

$$\beta = \frac{f\left(Z_0^3 r^2 \sin^3\phi + Y_0 r^2 \cos\phi \sin^2\phi - Z_0 r^2 \sin\phi - Y_0^2 Z_0 \sin\phi + Z_0^2 Y_0 \cos\phi\right)}{\left(Z_0^2 - r^2 \sin^2\phi\right)\left(Z_0 \cos s\phi - Y_0 \sin\phi\right)} - \frac{f Y_0}{Z_0}.$$

The equation quantifying the effect of our orthographic projection assumption on the semi-major axis is as follows.

$$\gamma = \frac{a Z_0}{f r} \tag{5.41}$$

The full version is much too messy to include here. a is the semi-major axis of the projection and can be solved for using (5.34) through (5.39), (5.10) and (5.11). Figures 5-23 through 5-27 show plots of α, β and γ for typical values: $R_0 = 10$cm, $Z_0 = 100$cm, $r = 0.5$cm and $f = 2000$ pixels. X_0 and Y_0 are converted to polar coordinates such that $R_0 = \sqrt{X_0^2 + Y_0^2}$. R_0 is fixed and θ is one of the axes plotted. Although we have not found it necessary, estimates of the transition length and the minor axis length can be easily obtained from the image and used to improve the calculated centroid and semi-major axis.

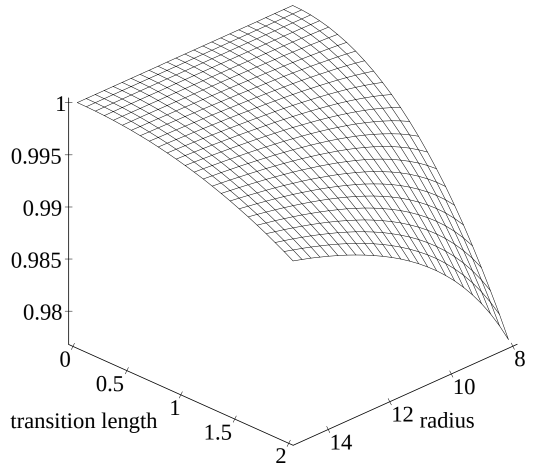

Figure 5-22: The effect of bleeding on the radius calculation. Error is expressed as the ratio of the actual radius and the calculated semi-major axis, d'/a'. The transition length and radius are expressed in pixels.

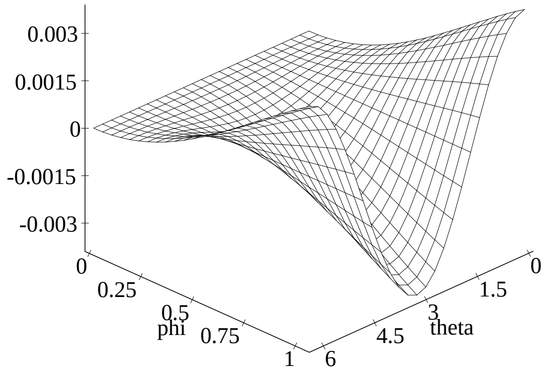

Figure 5-23: Perspective error in the centroid parallel to the major axis. The error is expressed in pixels.

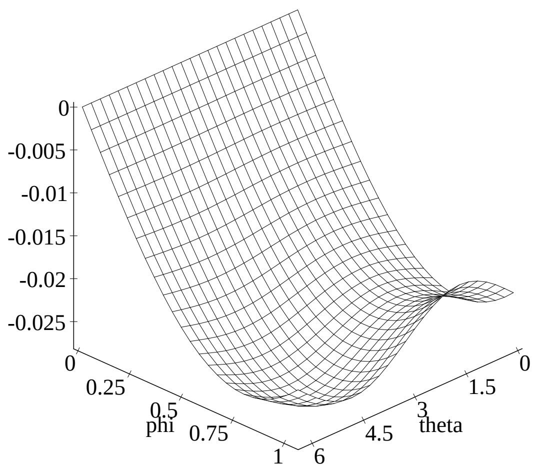

Figure 5-24: Perspective error in the centroid perpendicular to the major axis. The error is expressed in pixels.

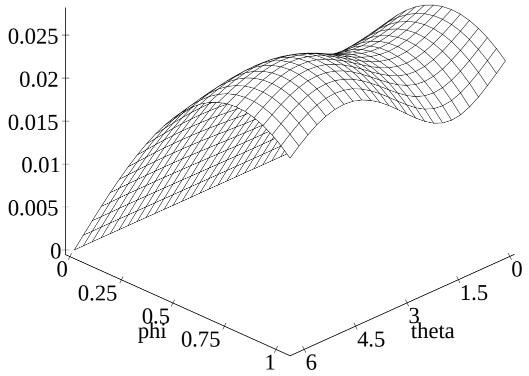

Figure 5-25: Total perspective error in the centroid. The error is expressed in pixels.

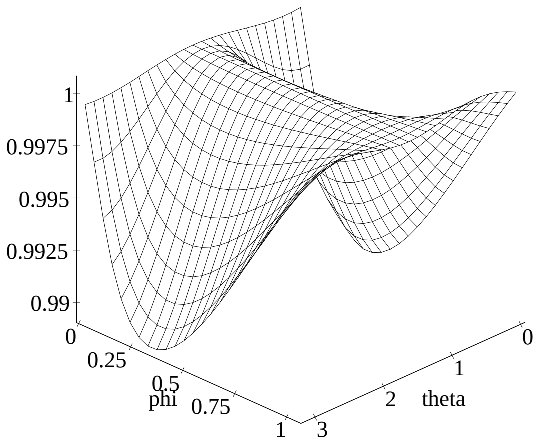

Figure 5-26: Perspective error in the semi-major axis for $R_0 = 10$cm. Error is expressed as the ratio of the calculated value and the actual radius.

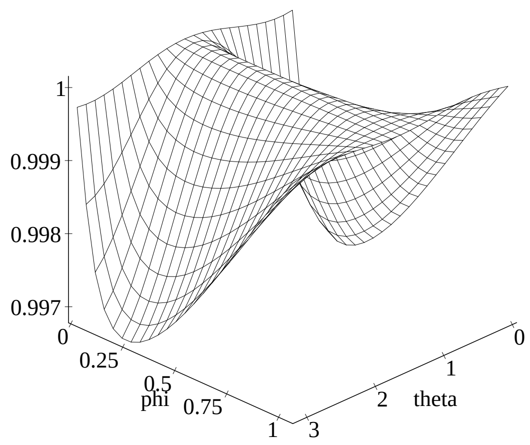

Figure 5-27: Perspective error in the semi-major axis for $R_0 = 10$cm. Error is expressed as the ratio of the calculated value and the actual radius.

5.3 Experiments

In the absence of noise, two images of a fiducial taken from the same position should produce the same centroid and semi-major axis. If our calculations were exact, a series of images taken from positions displaced only in a direction perpendicular to the optic axis should produce centroids that vary linearly with position. Similarly, a series of images taken from positions displaced only in a direction parallel to the optic axis should produce semi-major axes that vary linearly with the inverse of position. The degree to which real data deviate from these ideals is an empirical measure of the accuracy of our calculations.

Two sets of experiments were conducted using an optical bench. The first set of experiments examined the centroid calculations, the second set the semi-major axis calculations. A rail with a precision positioner runs along one side of the optical bench. For the first set of experiments, a camera was mounted on the positioner with its optic axis perpendicular to the rail. This setup allows camera motion only in the x direction. Motion in the y direction is achieved by rotating the camera $90°$ in its mounting. A Klinger DCS-750 motor controller with UE-72CC positioner was used to precisely position the camera. The controller/positioner combination is accurate to a few microns. On the optical bench roughly a meter away a fiducial was mounted so that it appeared near the center of the camera's field of view, see Figures 5-28 and 5-29. Experiments were performed for the following cases:

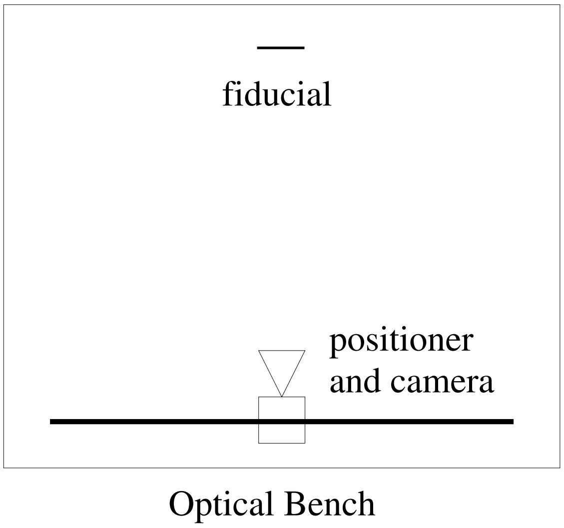

Figure 5-28: Top view of experimental setup for centroid.

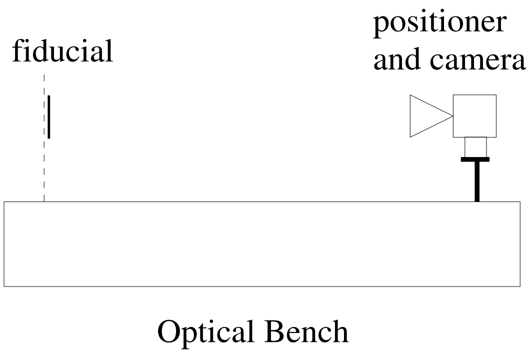

Figure 5-29: Side view of experimental setup for centroid.

- 10 micron and 100 micron steps
- Motion in both the x and y directions
- Fiducial parallel to the image plane and at a 45° angle
- Light and dark images

For each experiment, data was collected at 26 camera positions along the rail. At each position, 100 images were collected. The mean and standard deviation were calculated for each position. The results are shown in Figures 5-30 through 5-41. The mean at each position is marked by an "x". The error bars are 1 standard deviation above and below the mean. The line is the least squares best fit to the means. Table 5.1 shows both the theoretical and empirical accuracy of the centroid calculation for three conditions. The theoretical and empirical results are in good agreements.

Condition	Dynamic Range	Major Axis	Empirical Accuracy	Theoretical Accuracy
Bright, flat	80	17.5	0.065	0.087
Dark, flat	6	17.5	0.15	0.16
45° Angle	40	17.5	0.22	0.16

Table 5.1: Empirical and theoretical accuracy for centroid calculations.

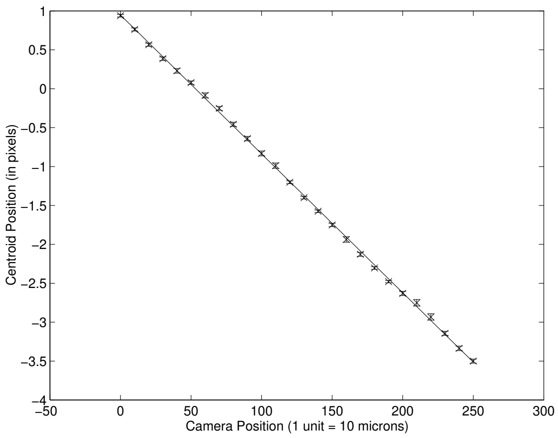

Figure 5-30: Data for bright image, camera motion in the x direction with 100 micron steps and fiducial parallel to the image plane.

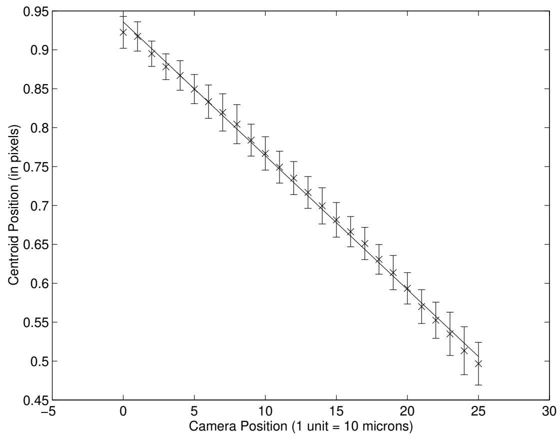

Figure 5-31: Data for bright image, camera motion in the x direction with 10 micron steps and fiducial parallel to the image plane.

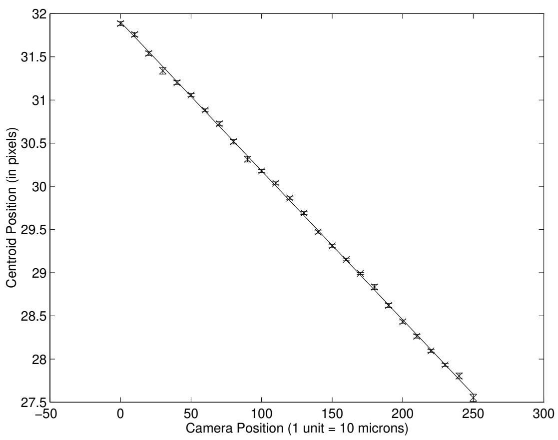

Figure 5-32: Data for bright image, camera motion in the y direction with 100 micron steps and fiducial parallel to the image plane.

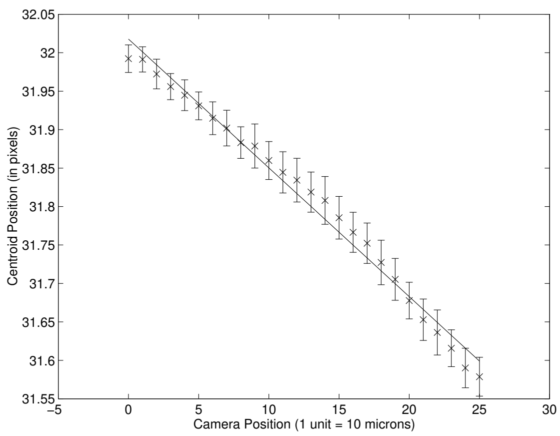

Figure 5-33: Data for bright image, camera motion in the y direction with 10 micron steps and fiducial parallel to the image plane.

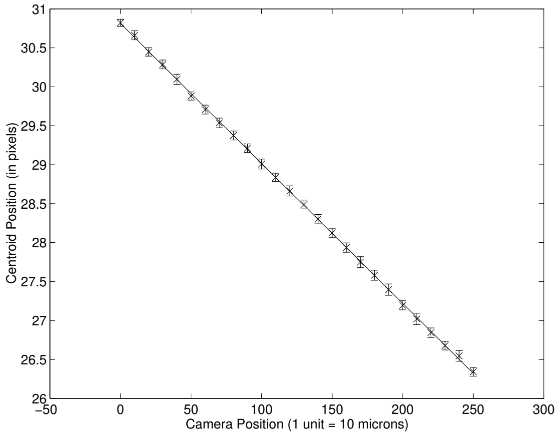

Figure 5-34: Data for dark image, camera motion in the x direction with 100 micron steps and fiducial parallel to the image plane.

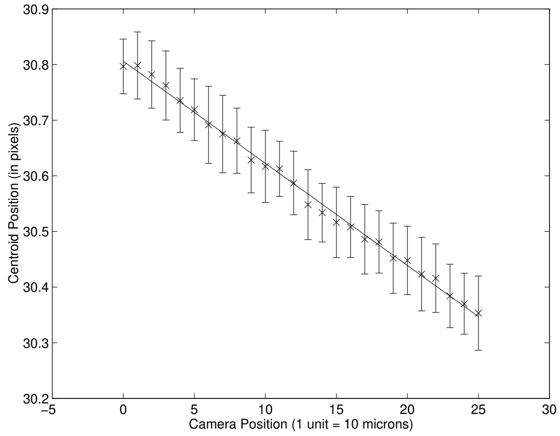

Figure 5-35: Data for dark image, camera motion in the x direction with 10 micron steps and fiducial parallel to the image plane.

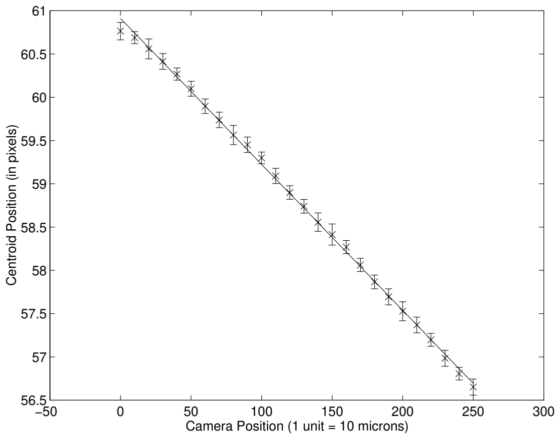

Figure 5-36: Data for dark image, camera motion in the y direction with 100 micron steps and fiducial parallel to the image plane.

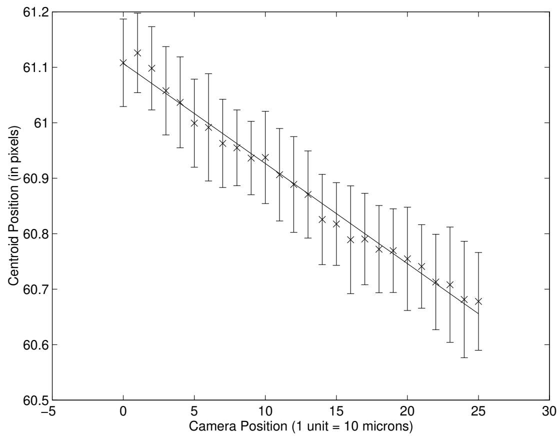

Figure 5-37: Data for dark image, camera motion in the y direction with 10 micron steps and fiducial parallel to the image plane.

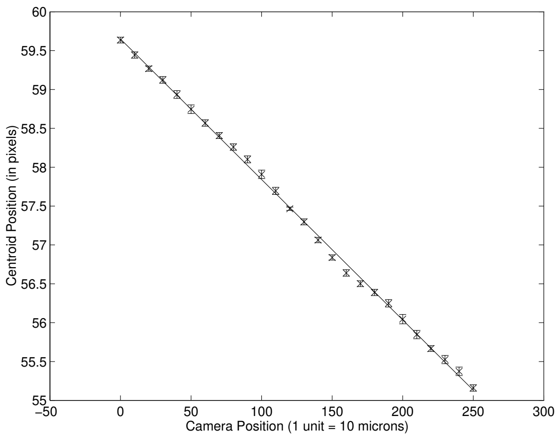

Figure 5-38: Data for bright image, camera motion in the x direction with 100 micron steps and fiducial 45° to the image plane.

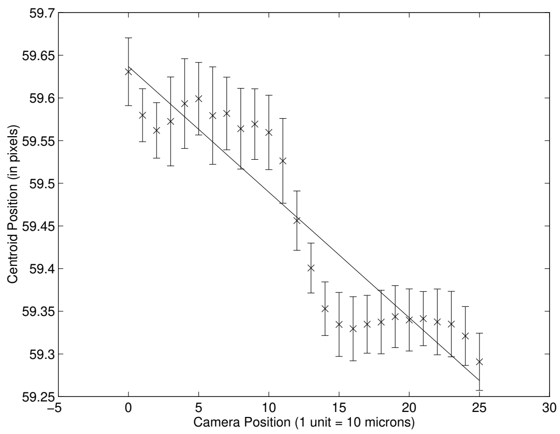

Figure 5-39: Data for bright image, camera motion in the x direction with 10 micron steps and fiducial 45° to the image plane.

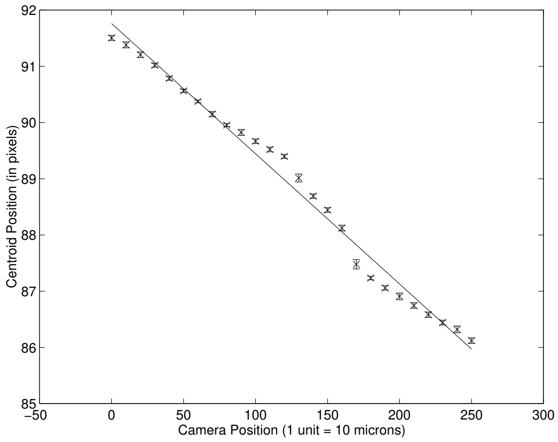

Figure 5-40: Data for bright image, camera motion in the y direction with 100 micron steps and fiducial 45° to the image plane.

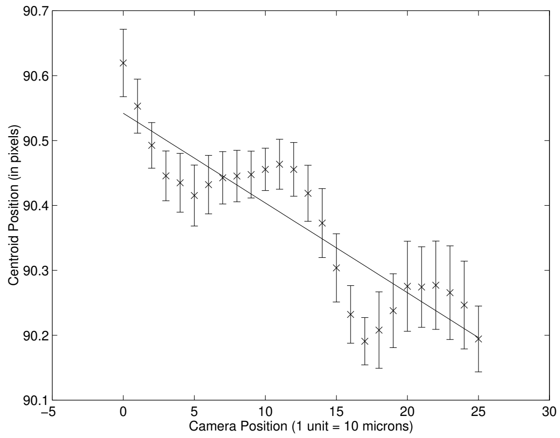

Figure 5-41: Data for bright image, camera motion in the y direction with 10 micron steps and fiducial 45° to the image plane.

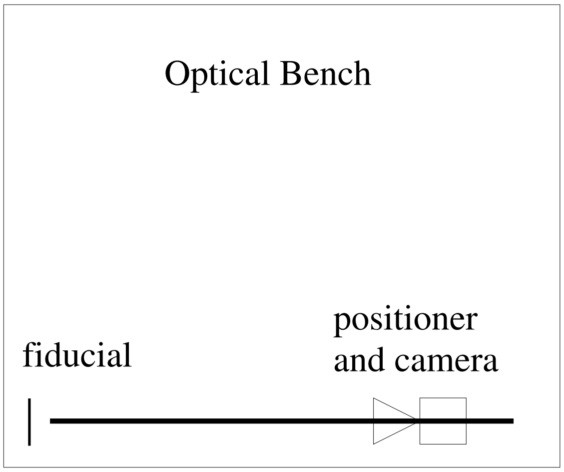

Figure 5-42: Top view of experimental setup for semi-major axis.

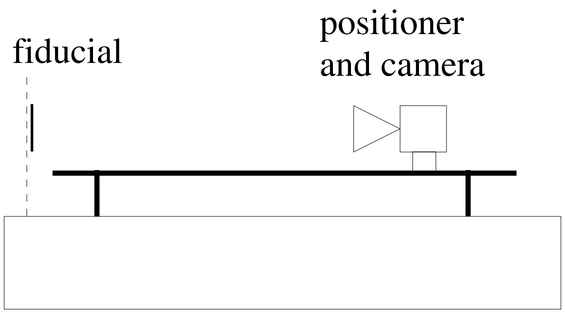

Figure 5-43: Side view of experimental setup for semi-major axis.

A similar set of experiments were conducted for the semi-major axis. For this set of experiments, the camera was mounted with its optic axis parallel to the rail. This setup allows camera motion only in the z direction. As before a fiducial was mounted on the optical bench roughly a meter away so that it appeared near the center of the camera's field of view, see Figures 5-42 and 5-43. Experiments were performed for the following cases:

- Fiducial parallel to the image plane and at a 45° angle
- Light and dark images

For each experiment, data was collected at 26 camera positions along the rail. At each position, 100 images were collected. The mean and standard deviation were calculated for the major axis at each position. The least squares best fit to the means was also found. The results are shown in Figures 5-44 through 5-46. Table 5.2 shows both the theoretical and empirical accuracy of the semi-major axis calculation for three conditions. The theoretical and empirical results are in good agreements.

Condition	Dynamic Range	Major Axis	Empirical Accuracy	Theoretical Accuracy
Bright, flat	40	17.5	0.15	0.16
Dark, flat	4	17.5	0.23	0.24
45° Angle	50	16.8	0.24	0.23

Table 5.2: Empirical and theoretical accuracy for semi-major axis calculations.

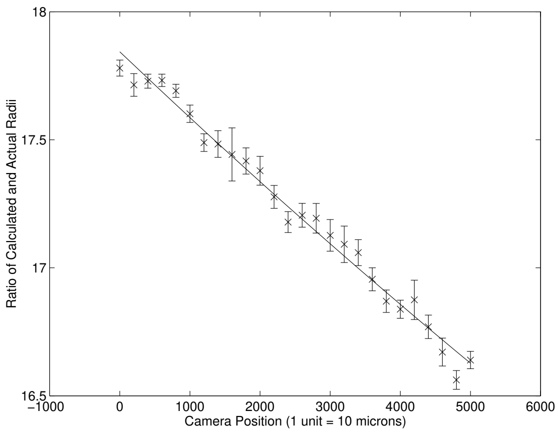

Figure 5-44: Data for bright image, camera motion in the z direction with 2000 micron steps and fiducial parallel to the image plane.

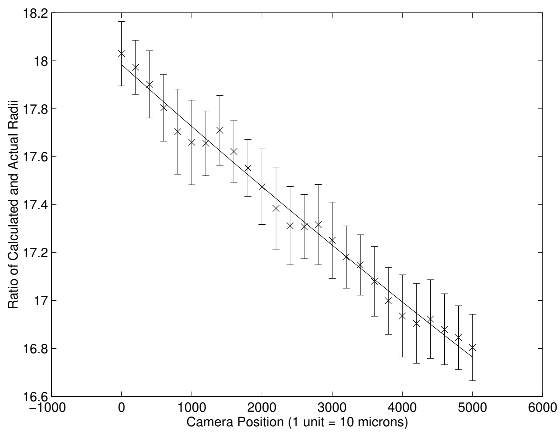

Figure 5-45: Data for dark image, camera motion in the z direction with 2000 micron steps and fiducial parallel to the image plane.

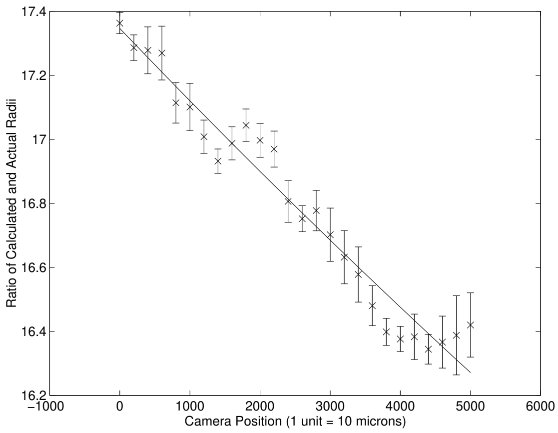

Figure 5-46: Data for bright image, camera motion in the z direction with 2000 micron steps and fiducial 45° to the image plane.

5.4 Discussion

An accurate, efficient and robust method of locating fiducials has been described. A thorough error analysis of the method has been provided including both theoretical and empirical data. This data confirms that in the worst case fiducials can be located to within 0.25 pixels and the local scale factor can be determined to within 1.5%.

Chapter 6

Initial Calibration

Before our fiducials can be used to perform enhanced reality visualizations, their model coordinates must be known. In cases were the model coordinates of the fiducials are not known a priori, an initial calibration must be performed. An initial alignment is performed using data from a laser scanner [Grimson *et al.*, 1994]. This initial alignment along with an image showing the fiducials is then used to *lookup* the model (MR or CT) coordinates of the fiducials. Figure 6-1 shows an overview of this process. Once the coordinates of the fiducials have been established the laser scanner is no longer needed and any camera which can view the fiducials can be used for enhanced reality visualization. In this chapter we provide a general discussion of how the laser scanner works and then give the details of fiducial calibration.

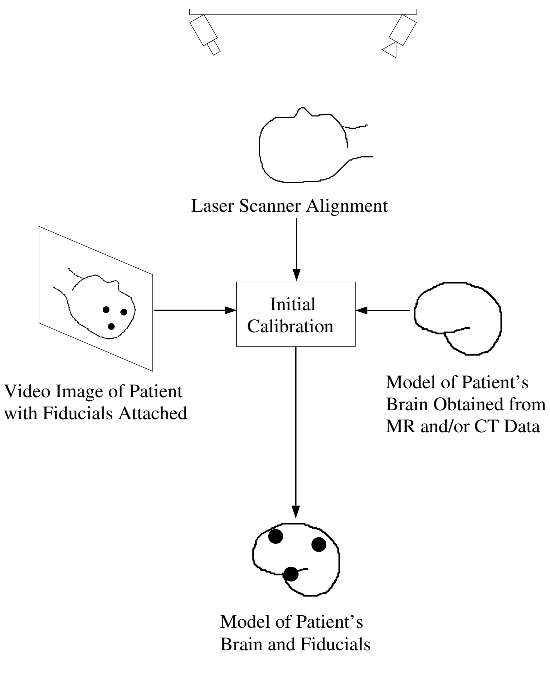

Figure 6-1: Determining the model (MR) coordinates of the fiducials.

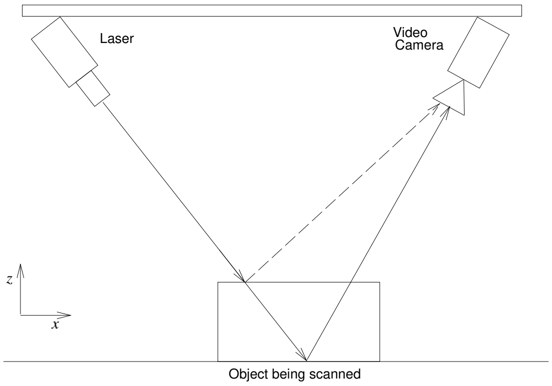

Figure 6-2: Side view of scanner.

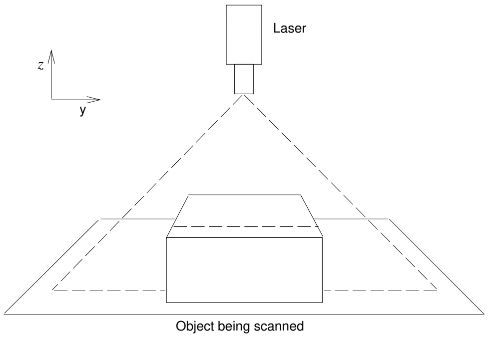

Figure 6-3: Object and laser light plane from video camera perspective.

6.1 Laser Scanner

For the skull data shown in Chapter 7, the initial calibration was performed with the aid of a laser range scanner produced by Technical Arts Corporation. It produces a planar sheet of light which is scanned using an oscillating mirror. A video camera is placed at an angle to the plane of light, see Figure 6-2. The x axis is parallel to the line segment joining the laser and video camera. The laser is oriented so that the line of illumination formed when the laser's plane of light strikes an object is perpendicular to the x axis. The y axis is parallel to the line of illumination, see Figure 6-3. The z axis is orthogonal to both the x and y axes. The y axis in Figure 6-2 is into the page and the x axis in Figure 6-3 is out of the page. The three dimensional coordinates of an object's surface can be determined if it is placed so that it can be simultaneously illuminated by the laser and viewed by the video camera. The y coordinate of a data point is determined directly from the horizontal displacement measured by the video camera. The x and z coordinates are recovered using the vertical displacement and the scan angle of the laser beam. A single scan produces 240 three dimensional measurements with accuracies up to 0.003".

Surface points on the patient's head near the surgical site are measured with the scanner. These data points are aligned with a model of the patient's head and brain obtained from previous MR and/or CT scans [Grimson *et al.*, 1994]. The registration is produced by minimizing the sum of the squared distance between the laser data and the model. The model is sampled at several resolutions to speed convergence and random perturbations are used to avoid

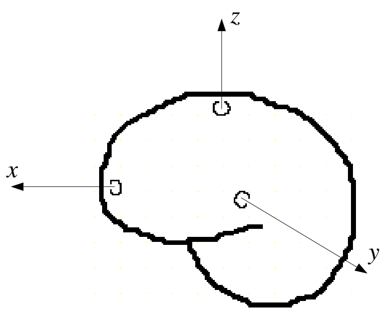

Figure 6-4: Model of patient's brain with coordinate axes.

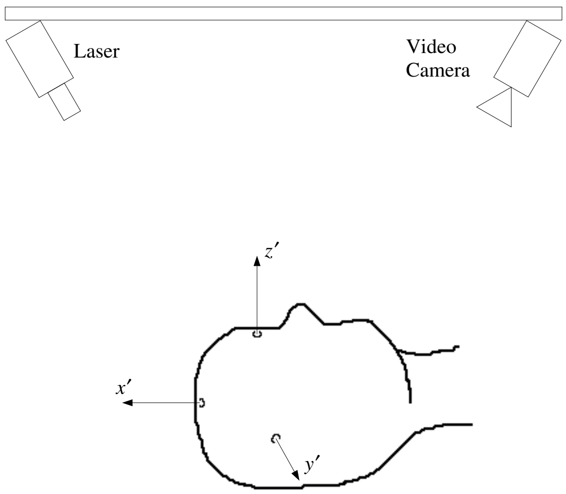

Figure 6-5: Patient, scanner and coordinate axes.

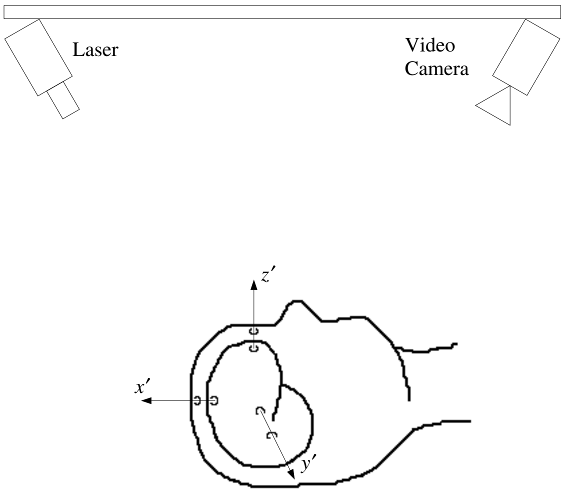

Figure 6-6: Model of patient's brain aligned with patient (valid only for the scanner).

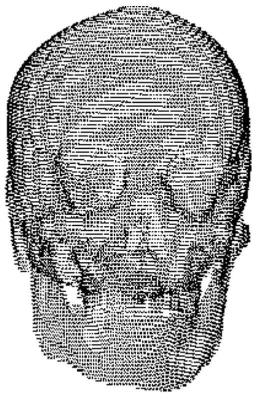

Figure 6-7: Model of a skull.

Top view Side view End view

Figure 6-8: Laser data from skull.

Figure 6-9: Video of skull being scanned.

Figure 6-10: Laser data aligned with skull.

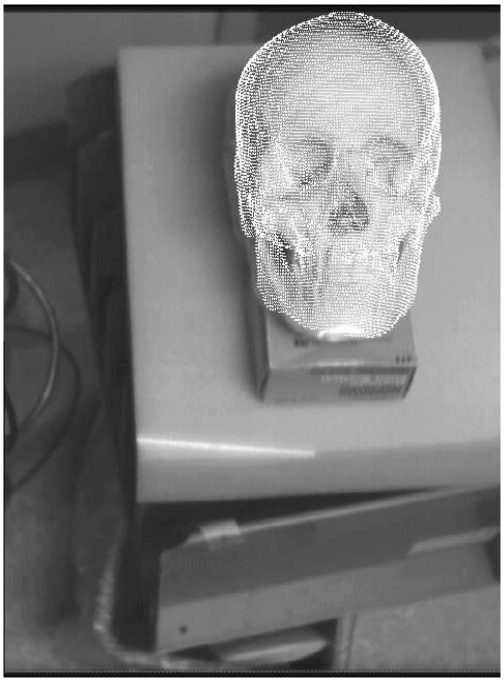

Figure 6-11: Model aligned with video of the same skull.

local minima. This produces a transformation from the coordinate frame of the model to that of the patient. Figure 6-4 shows a brain model and its coordinate system. Figure 6-5 shows the patient, scanner and their coordinate system. Figure 6-6 shows the result of aligning the model and the patient coordinate systems. It should be noted that Figure 6-6 is valid only from the perspective of the camera attached to the laser scanner.

Figure 6-7 shows a model obtained from CT imaging of a plastic skull. Figure 6-8 shows the three dimensional data obtained from laser scanning the same skull. Figure 6-9 shows two images of the skull containing laser scan lines. Figure 6-10 shows the laser data aligned with the skull. Figure 6-11 shows the model of the skull aligned with and superimposed upon video of the skull. The accuracy of this registration is believed to be on the order of the resolution of the MR or CT data ($\approx$ 1mm).

6.2 Calibration Routine

The transformation used to produce Figure 6-11 essentially maps model coordinates to image coordinates. This is exactly the inverse of what we need to calibrate our fiducials. What we would like to be able to do is measure the image coordinates of the fiducials and then use an inverse mapping to determine their model coordinates. The inverse mapping is not in general a one to one mapping. The image coordinates of a fiducial determine the ray in three space along which the model coordinates of the fiducial must lie. By intersecting this ray with the model and Z-buffering the result we can determine the model coordinate of the fiducial.

The initial calibration is limited by the accuracy of the registration performed using the laser scanner. The laser scanner registration produce results which are both repeatable and good for the single view in question. It is not clear how accurate the registration is in an absolute sense. Absolute accuracy is important for calibrating the fiducials. For example, small errors which are imperceptible from one point of view frequently lead to large errors from different view points. Perturbing the laser scanner solution by less than $1°$ can change a fiducial's location by over 6mm. The uncertain accuracy of the fiducial calibration bears significantly on the quality of enhanced reality visualizations which can be produced. In spite of this issue, the results shown in Chapter 7 are promising. The exact source and nature of the errors in the initial calibration needs to be explored further. The method of initial calibration presented here, while it requires the use of a laser scanner, has the advantage that it can be made fully automatic. Of course, other methods of initial calibration could be used. The only requirement is that the fiducial locations be determined accurately. How this information is obtained does not matter.

Chapter 7

Results

7.1 Test Object

To determine the accuracy of our method we performed several experiments using a special test object. A three dimensional object with seven fiducials was made. The relative positions of the fiducials were accurately measured. Figure 7-1 shows the basic test object. The large pillar near the center is used to measure the accuracy of the enhanced reality visualization. A wire frame corresponding to the edges of the pillar is displayed in the enhanced reality image. The difference between the actual edges and wire frame is a measure of the accuracy of the visualization. For comparison purposes a wire frame is also superimposed on a shorter pillar. Experiments were performed with four slightly different fiducial configurations. The first configuration consists of 6 coplanar fiducials plus a single fiducial 1cm above the plane. The enhanced reality visualizations produced from this configuration are not consistently accurate. Figures 7-2 through 7-5 are typical of the results for this configuration. Notice that the wire frame on the smaller pillar matches in all of the images. Errors occur only significantly away from the volume enclosed by the fiducials. The second configuration moves one of the 6 coplanar fiducials so that it is also 1cm above the plane. Figures 7-6 through 7-8 show typical results for this configuration. The accuracy is greatly improved with the addition of a second noncoplanar fiducial, however occasional slight errors do occur. The next configuration adds a third noncoplanar fiducial. Typical results for this configuration are shown in Figures 7-9 and 7-10. With this configuration, errors rarely occur and when they do they are small. The final configuration is similar to the first except that the noncoplanar fiducial is 4cm out of the plane. Results for this configuration are shown in Figures 7-11 and 7-12. The accuracy of this configuration is comparable to that of the last. Some depth in the model is required to accurately recover the third dimension. Once sufficient depth is present in the model, it appears that the solution is accurate over a large volume.

As shown in Figures 7-2 through 7-12, under reasonable conditions our method produces good results. These figures provide only a subjective measure of accuracy. Next we will attempt to provide a more quantitative measure.

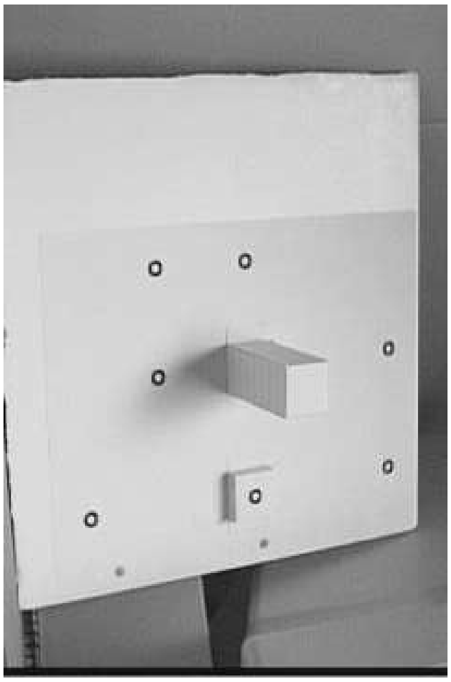

Figure 7-1: Test object.

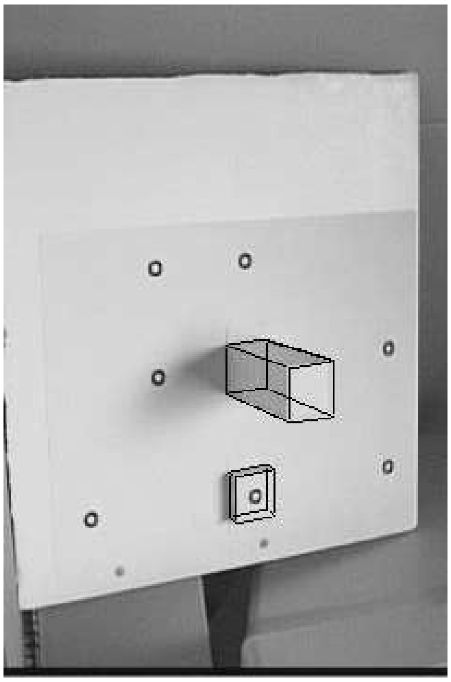

Figure 7-2: Test object view 1.

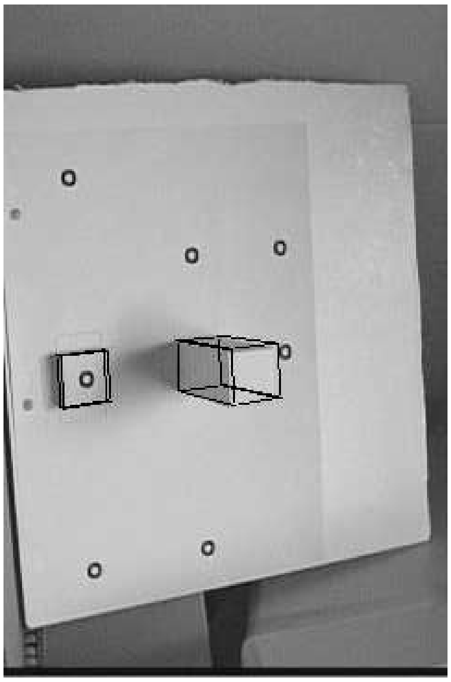

Figure 7-3: Test object view 2.

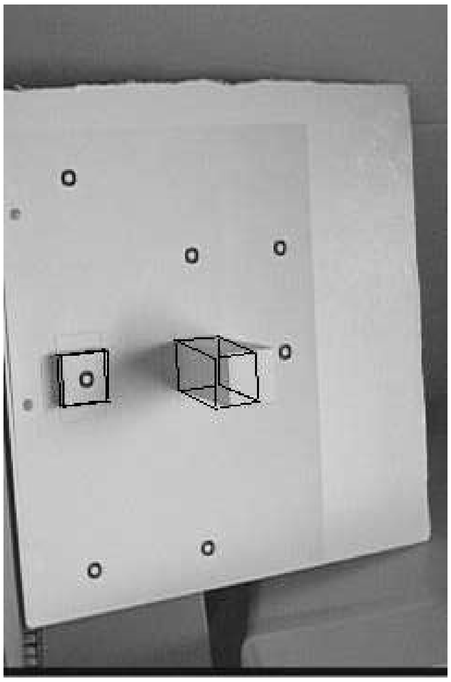

Figure 7-4: Test object view 3.

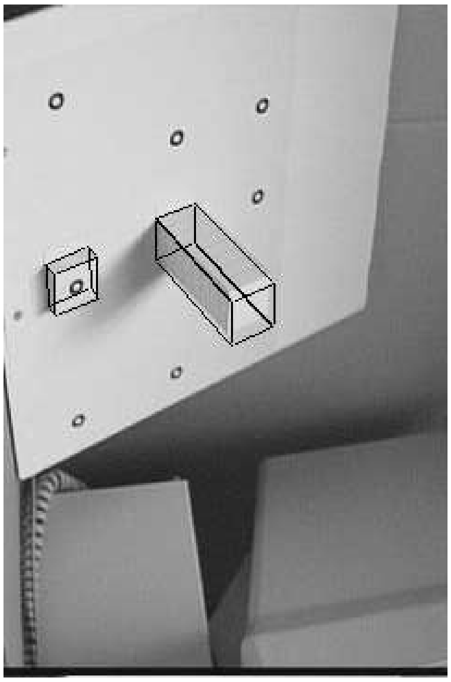

Figure 7-5: Test object view 4.

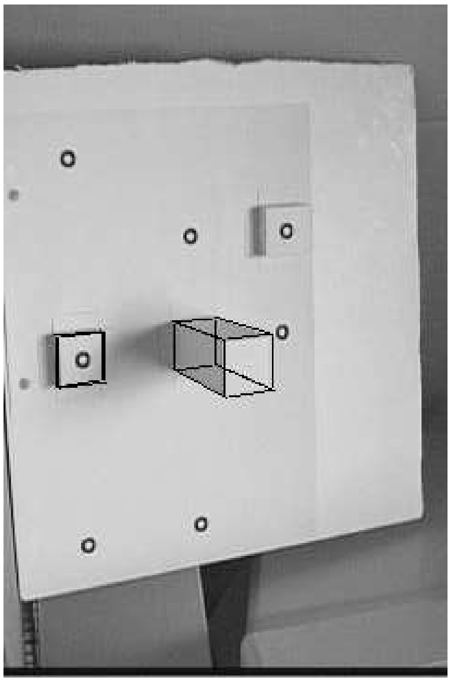

Figure 7-6: Test object view 5.

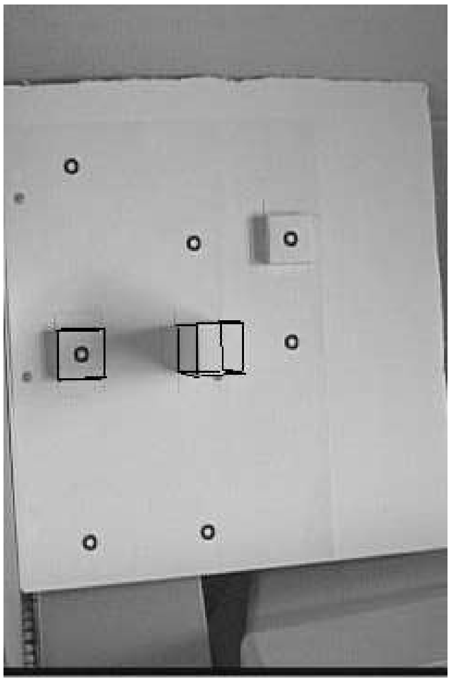

Figure 7-7: Test object view 6.

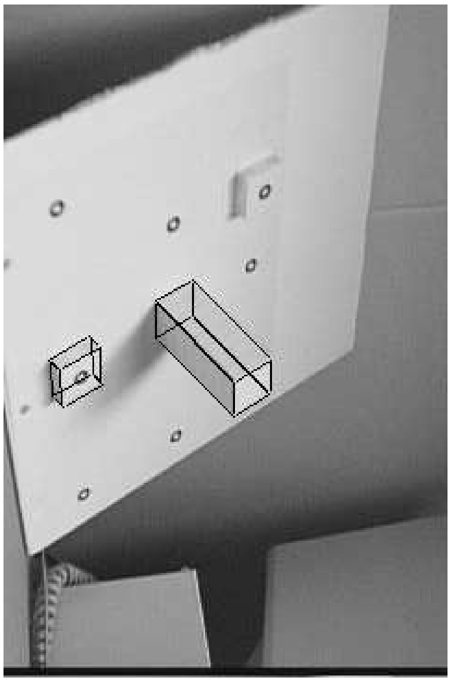

Figure 7-8: Test object view 7.

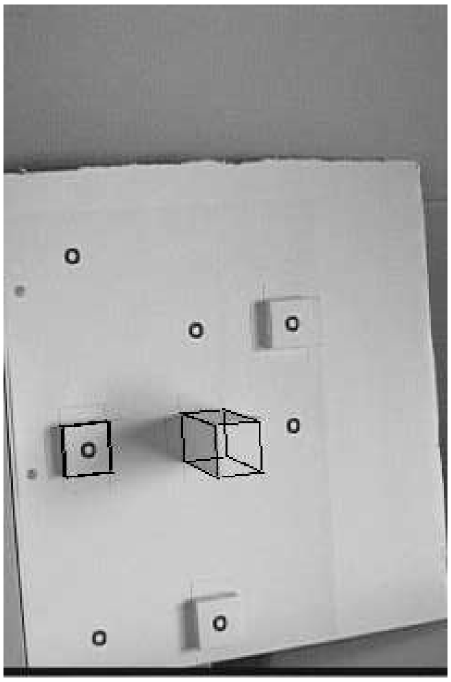

Figure 7-9: Test object view 8.

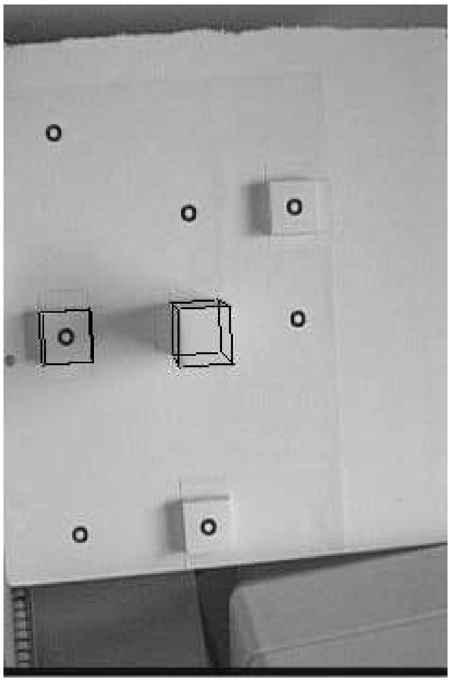

Figure 7-10: Test object view 9.

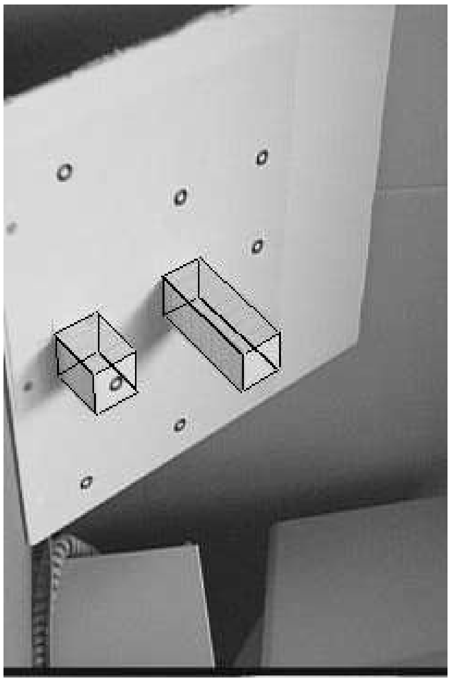

Figure 7-11: Test object view 10.

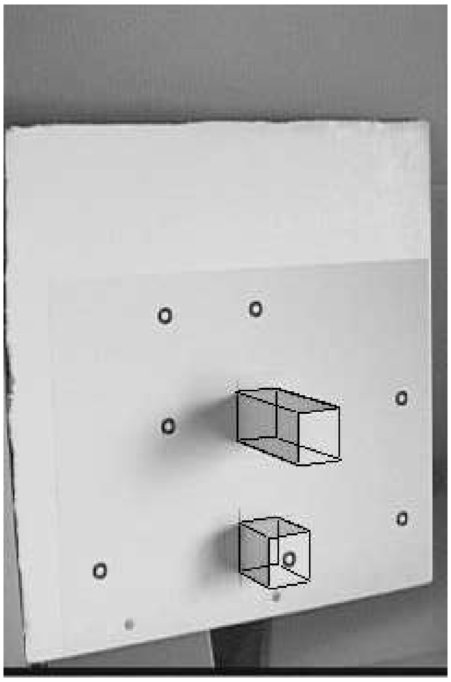

Figure 7-12: Test object view 11.

Average Distance	RMS Distance	Maximum Distance	Median Distance	Number of Points
1.35	1.46	2.94	1.38	848

Table 7.1: Distance between edge-based and enhanced reality vertices.

Average Misalignment	RMS Misalignment	Maximum Misalignment	Median Misalignment	Number of Points
0.88	0.91	1.77	0.87	212

Table 7.2: Misalignment between edge-based and enhanced reality polygons.

Given that the goal of enhanced reality visualization is to produce an *enhanced* image, the proper way to assess accuracy is to consider the deviation of the enhanced image from the *ideal* image. For various reasons we do not have access to the ideal image. However, we can compare the deviation between the image of a physical object and the enhancement produced from a model of the same physical object. For example, we could compare the wire frame to the edges of the test object's central pillar. This is essentially what we will do. Figure 7-13 shows the same test object used above with one change, the top of the central pillar had been darkened to provide high contrast edges. Edge pixels are extracted and chained together using a Canny edge detector. A line is fitted to the chains by minimizing the distance between the edge pixels and the line. These lines are used to compute two measures of accuracy. The first measure is the distance between the vertices of the darkened region as determined by our method (enhanced reality visualization) and those determined by intersecting the lines recovered above. The second measure is the area of misalignment divided by the perimeter or the average distance between the two polygons, see Figure 7-14.

A sequence of 212 images of the test object rotating through 360° was used. The first measure was computed for each of the four vertices in each image. The distance between the vertices in each image are shown in Figures 7-15 through 7-18. The second measure was computed for the darkened polygon in each image and the average distance between the polygons for each image is shown in Figure 7-19. Tables 7.1 and 7.2 summarize these results. The edge-based positions agree well with those produced by our method. It should be noted that edges and vertices recovered using the Canny edge detector are not without error. The most significant source of error is the fact that the implementation used only localizes the edge to the nearest pixel. As a result of

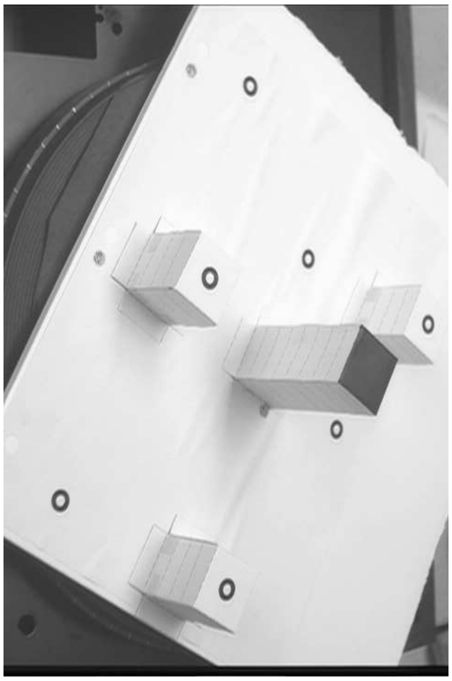

Figure 7-13: Test object used to quantify accuracy.

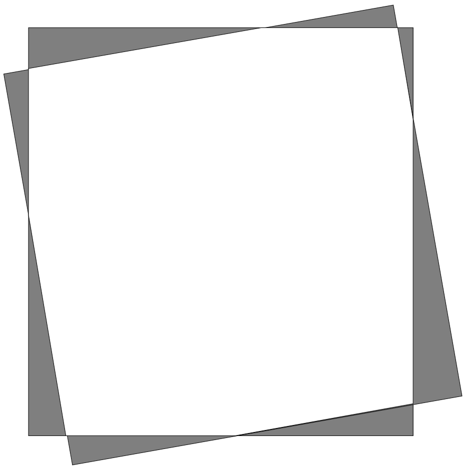

Figure 7-14: Misalignment of edge-based rectangle and enhanced reality rectangle.

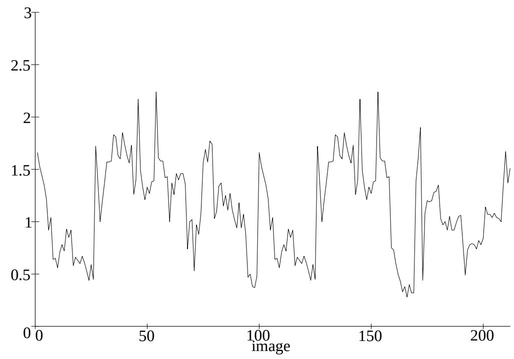

Figure 7-15: Distance in pixels between edge-based and enhanced reality positions for vertex 1.

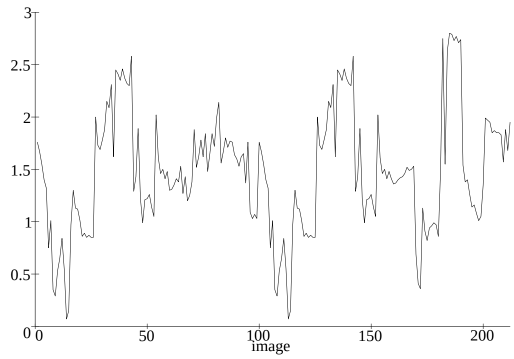

Figure 7-16: Distance in pixels between edge-based and enhanced reality positions for vertex 2.

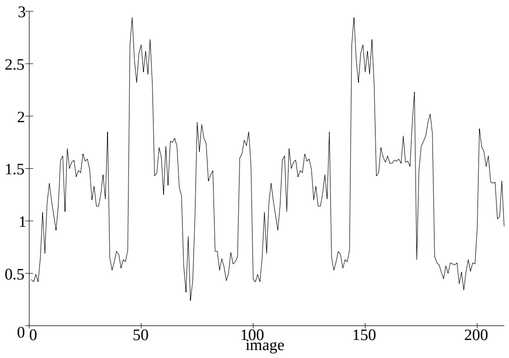

Figure 7-17: Distance in pixels between edge-based and enhanced reality positions for vertex 3.

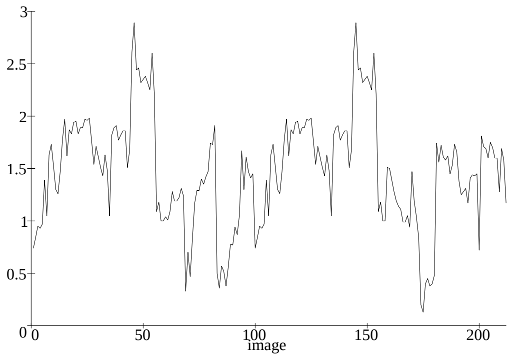

Figure 7-18: Distance in pixels between edge-based and enhanced reality positions for vertex 4.

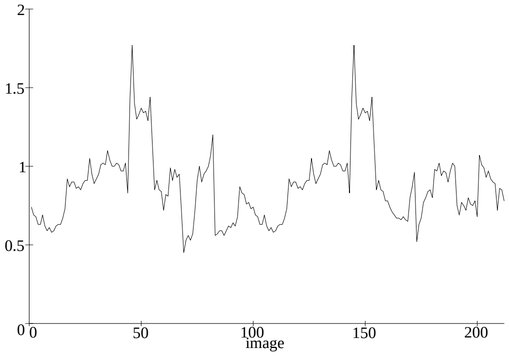

Figure 7-19: Average distance between edge-based and enhanced reality polygons.

this, the first measure probably over estimates the difference between the *edge* image and the *enhanced* image. A signed measure of the distance from point to line[1] was also computed to check for correlation in the errors. The average for the first measure was 0.09 pixels and for the second measure was 0.04 pixels making it unlikely that the errors in edge-based positions are correlated with those in the enhanced reality positions. The average difference between the two perimeters is likely the better measure of accuracy and using this measure our method is accurate to within a pixel.

7.2 Skull

After calibrating the fiducials as described in Chapter 6, enhanced reality visualizations were performed from several different view points. Figure 7-20 shows the initial view of a plastic skull. Figure 7-21 shows the results of the registration using the laser scanner. The white dots are CT data points for the skull superimposed upon an image of the skull. Figure 7-22 shows the results of our method using the initial view point. Note that as expected the error in the Figures 7-21 and 7-22 are comparable. Figures 7-23 through 7-32 show the results of our method using ten novel view points. The exact source of the misalignment present in some of the figures is not known. Two likely sources are the initial calibration and the fact that the fiducials are nearly coplanar. The errors are largest for view points significantly different from that used during the initial calibration which suggests that at least some of the misalignment is caused by errors introduced during the initial calibration. Also, as noted in Chapter 6 small perturbations in the laser scanner alignment cause relatively large variations in the calibrated positions of the fiducials. A more robust initial calibration and a method of handling coplanar fiducials, which together should eliminate these errors, are currently being investigated.

[1] The sign indicates which side of the line the point is on.

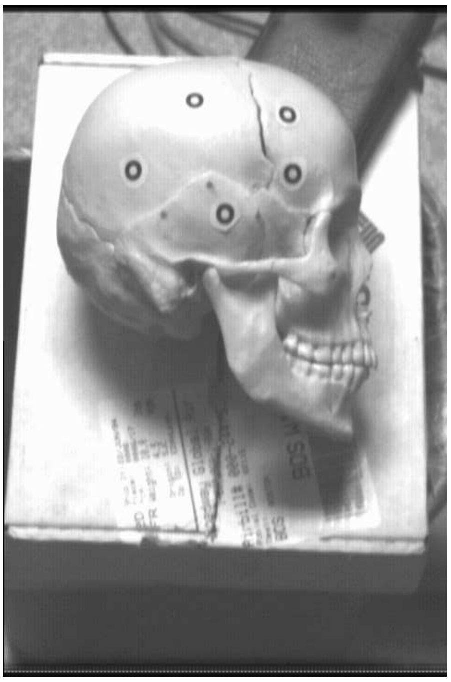

Figure 7-20: Plastic skull.

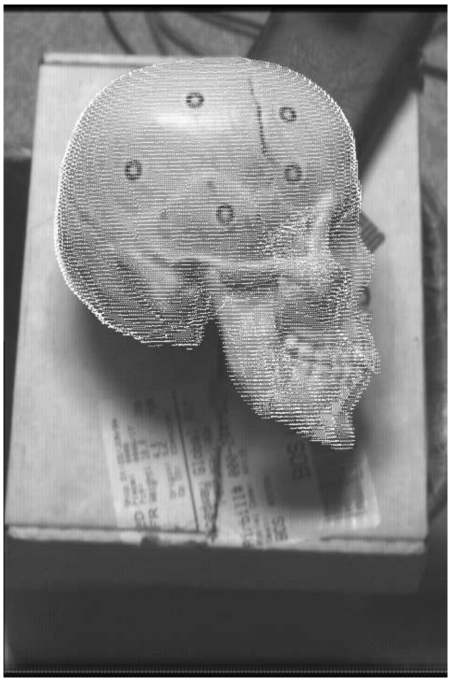

Figure 7-21: Initial registration using the laser scanner.

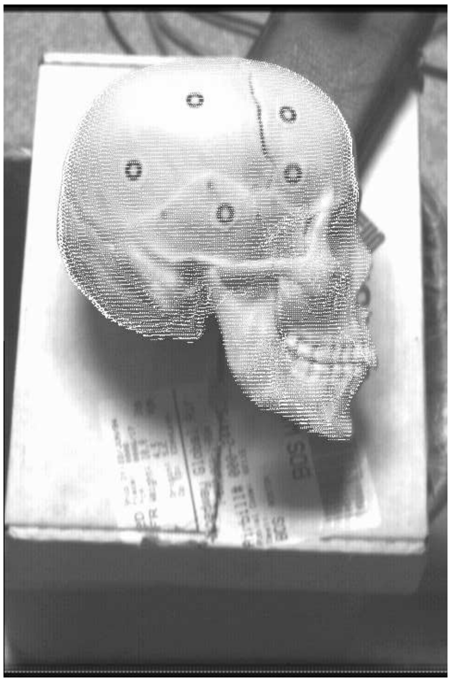

Figure 7-22: Skull initial view.

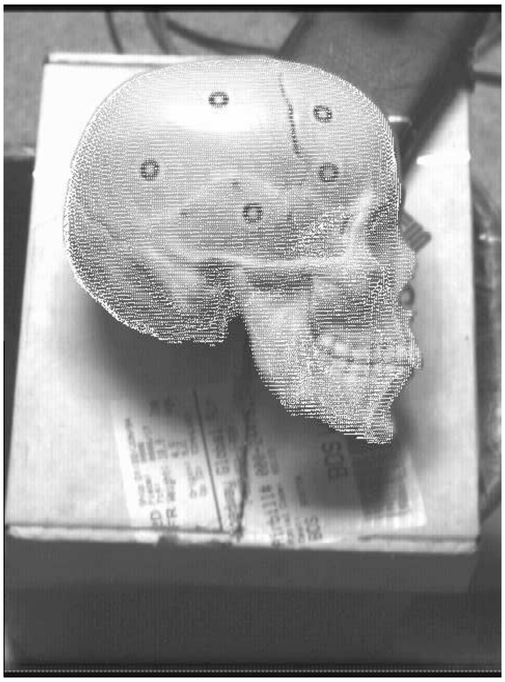

Figure 7-23: Skull view 1.

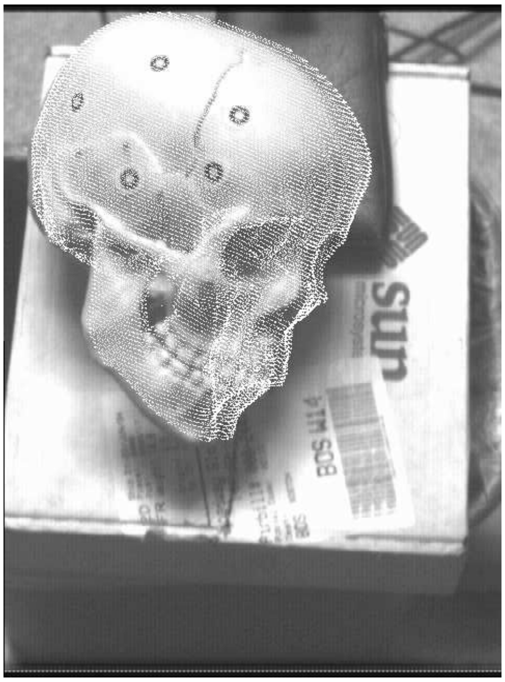

Figure 7-24: Skull view 2.

Figure 7-25: Skull view 3.

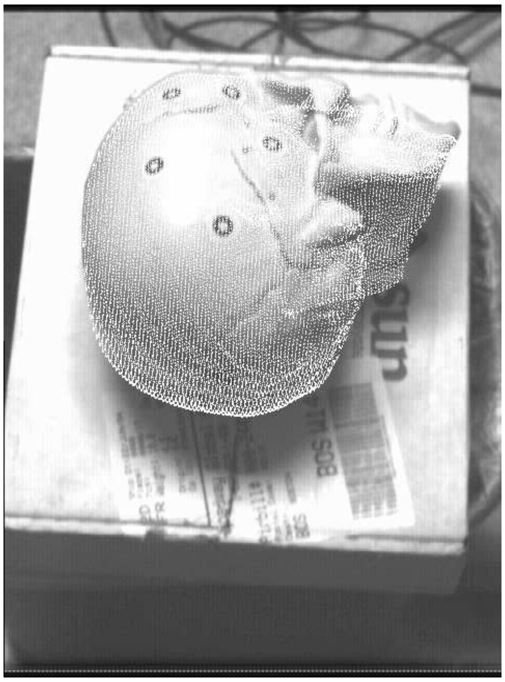

Figure 7-26: Skull view 4.

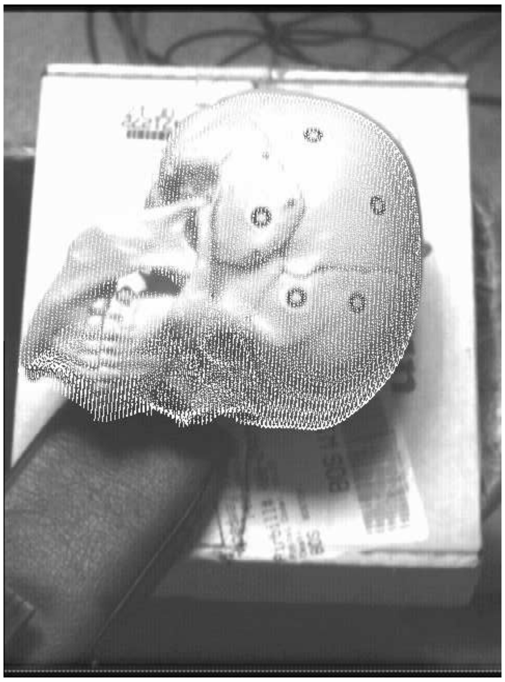

Figure 7-27: Skull view 5.

Figure 7-28: Skull view 6.

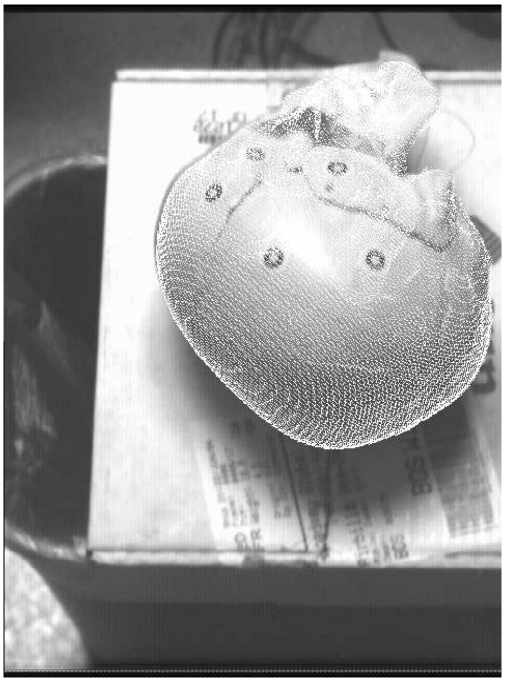

Figure 7-29: Skull view 7.

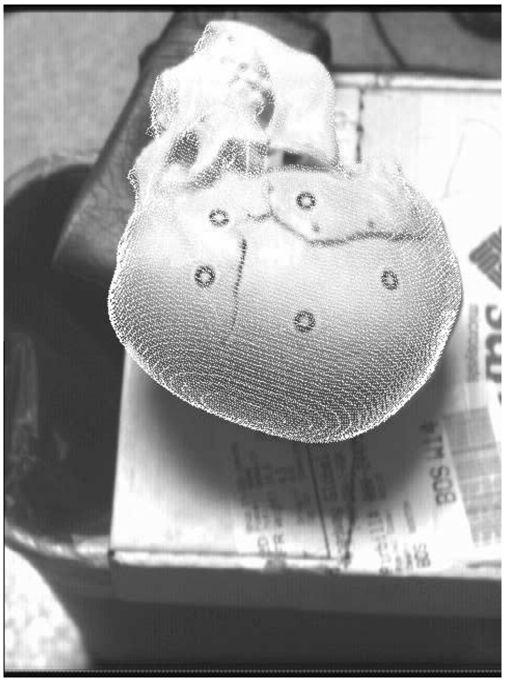

Figure 7-30: Skull view 8.

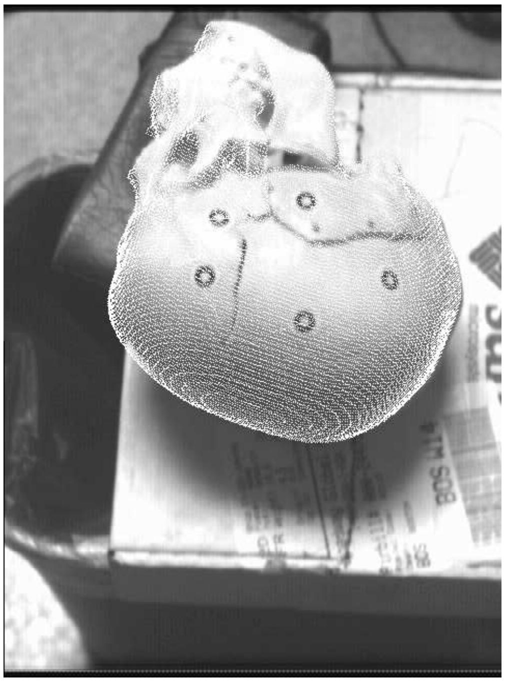

Figure 7-31: Skull view 9.

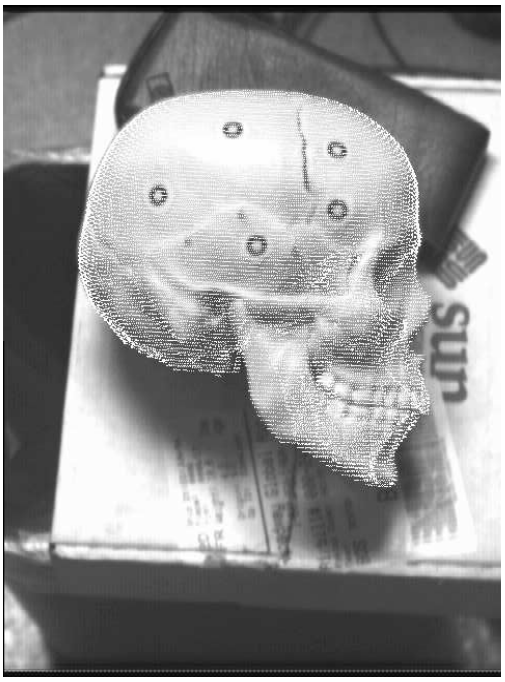

Figure 7-32: Skull view 10.

Chapter 8

Conclusions

8.1 Future Work

There are many improvements and extensions which can be made to the basic method presented in this report. Several of them have been alluded to in previous chapters.

8.1.1 Auto calibration

The current implementation requires knowledge of the ratio of pixel spacing in the x and y directions, $s_{\mathrm{x/y}}$ in order to recover s. While $s_{\mathrm{x/y}}$ is extremely stable and determining its value is straight forward, it would be nice to eliminate this requirement. Given enough noncoplanar fiducials it should be possible to solve for $s_{\mathrm{x/y}}$. Solving for $s_{\mathrm{x/y}}$ might be fairly time consuming but it should only need to be performed once. Similarly, it would be nice to be able to handle lens distortion (although we have not found it necessary). It is a simple matter to add a lens distortion model as a preprocessor. The fiducial locations are simply corrected by the amount specified by the model. In some cases considering distortion would surely improve the results. Unfortunately this requires finding the proper distortion model. There are several well established methods for determining lens distortion. Ideally, the model would be determined automatically. Lens distortion changes with several of the other camera parameters such as aperture, focus and zoom. Even so, given several data sets consisting of image points, model points, perspective transformation, aperture, focus and zoom settings we should be able to construct a model which takes into account the effect of changes to aperture, focus and zoom. It should be necessary to construct a new model or modify an old model only occasionally. Our method as it is currently implemented models a linear approximation to lens distortion implicitly. Automatically determining a value for $s_{\mathrm{x/y}}$ and a model for lens distortion are two examples of auto calibration. Auto calibration would bridge the gap between what is implicitly modelable and other required or desired parameters. The parameters which are implicitly modeled are those which can change from one image to the next. The parameters which are candidates for auto calibration are either fixed or vary on much longer time scales. This makes it possible to run an auto calibration routine in the background updating parameters as necessary but certainly not every frame. Auto calibration would enable a completely uncalibrated camera with a poor quality lens to be used to produce very accurate enhanced reality visualizations.

8.1.2 General Features and Self Extending Models

Where auto calibration enables camera parameters to be recovered, self extending models allow recovery of model parameters. Given a partial model and several solutions from different view points it should be possible to recover the three dimensional location of points not in the model but which are visible. This effectively extends the model. The ability to use general features in place of fiducials would be a significant improvement for many potential applications and makes self extending models truly useful. The circular fiducial currently used facilitates recovery of depth information. The same kind of size information can potentially be recovered from naturally occuring spatial features such as patches of texture, etc. Another possible option is to use stereo to recover depth information. Since relative depth is all that is required the stereo cameras need not be calibrated. Using a stereo setup would eliminate the need to know $s_{x/y}$ and would facilitate a stereo display.

8.1.3 Miscellaneous

Several other more modest improvements or extensions also exist. As noted in Chapters 4 and 7, noncoplanar points significantly improve the results of our method. With a slight modification to our method it should be possible to use planar data exclusively. The modification involves solving a quartic equation. The ability to function when all of the fiducials are coplanar would certainly be an asset. It is not clear what accuracy can be achieved by this approach. As noted in Chapter 6 there is room for improvement in the initial calibration of fiducials. At a minimum a better understanding of the source and nature of errors is needed. A more robust method of calibrating fiducials would also be very useful. Finally, the current implementation can be optimized significantly for speed.

8.2 Applications

Neurosurgery is just one application for enhanced reality visualization. There are numerous other potential applications both inside and outside the medical field. These applications range from manufacturing and repair to navigation and rescue. Enhanced reality visualization can be most readily applied in domains where models either already exist or are easily obtainable. In the medical field models are easily obtained using internal anatomy scanners such as MR and CT. Models already exist for many repair and manufacturing environments. Before enhanced reality visualization will be accepted for routine use an accurate and robust method of performing the visualization must be developed. Our method makes significant progress towards this goal. Better methods of generating models will make it practical to apply enhanced reality visualization to many more situations. Because it is anchored in the real world it has the potential to affect our everyday lives. For example, enhanced reality visualization can be used to transcend the limitations of computer monitors. By moving the display off the desk and into a visor or pair of goggles the display becomes much more useful. Multiple virtual screens can be defined. These virtual screens are anchored in the real world so the user, rather than fumbling with a mouse to expose the desired window, can simply look around and examine the contents of the various virtual screens. In effect the size of the display becomes unlimited. Since the virtual screens are anchored in the real world they are spatially organized which is a powerful organizational metaphor. For example, you can define a virtual screen containing a phone list and attach it to your bulletin board. Whenever you need to check a phone number on the list all that need be done is look at the bulletin board. The phone list will always be where you posted it.

8.3 Discussion

A new method for performing enhanced reality visualization has been developed. The method achieves good results using just a few fiducials placed near the volume of interest. Noncoplanar fiducials yield better results. Our method allows for motion and automatically corrects for changes to the internal camera parameters (focal length, focus, aperture, etc.) making it particularly well suited to enhanced reality visualization in dynamic environments. In a surgical application, we place a few fiducials placed near the surgical site immediately prior to surgery. An initial calibration is performed using a laser scanner. After the calibration is performed, our method is fully automatic, runs in nearly real-time and is accurate to a fraction of a pixel and requires only a single image.

Appendix A

Effects of Radial Distortion

As discussed in previous chapters, our method only models a linear approximation to radial distortion. In this appendix we will consider the consequences of this approximations. Radial distortion is typically modeled as follows [Slama, 1980]:

$$
\begin{aligned}
x'_{\mathrm{u}} &= x'_{\mathrm{d}} + \delta x && \text{(A.1)}\\
y'_{\mathrm{u}} &= y'_{\mathrm{d}} + \delta y && \text{(A.2)}\\
\delta x &= (x'_{\mathrm{d}} - x_0)\left(K_1 r'^2_{\mathrm{d}} + K_2 r'^4_{\mathrm{d}}\right) && \text{(A.3)}\\
\delta y &= (y'_{\mathrm{d}} - y_0)\left(K_1 r'^2_{\mathrm{d}} + K_2 r'^4_{\mathrm{d}}\right) && \text{(A.4)}
\end{aligned}
$$

Where x'_{u} and y'_{u} are the undistorted pixel coordinates, x'_{d} and y'_{d} are the distorted pixel coordinates with $r'^2_{\mathrm{d}} = (x'_{\mathrm{d}} - x_0)^2 + (y'_{\mathrm{d}} - y_0)^2$, and K_1 and K_2 are the radial distortion coefficients. If K_1 and K_2 are known, it is quite simple to use this radial distortion model to *correct* the data before passing it to our method. However, what if K_1 and K_2 are not known? The linear approximation contained in our method works well if radial distortion is not *too great*. It also works well if the fiducials are near the location of the enhancement.

As a preliminary step, we measured the radial distortion for our camera setup. Radial distortion was measured using the plumb-line method [Brown, 1971, Stein, 1993] for a 16mm and 25mm lens. The results are summarized in Table A.1. Plots of the distortion are also shown in Figures A-1 and A-2.

In order to gain some insight into the effect of radial distortion on our method we will attempt to quantify the difference between the radial distortion model

Lens	x_0	y_0	K_1	K_2
16mm	336	246	2.0e-8	2.0e-13
25mm	332	248	-4.0e-8	1.0e-13

Table A.1: Radial distortion parameters for a 16mm and 25mm lens.

described above and the linear approximation included in our method. To quantify the difference, a set of 5 three dimensional *control points* K were selected and a perspective transformation $\mathcal{P}$ (composed of a rigid transformation and a camera calibration matrix as described in Chapter 4) was defined. The control points were then projected onto the image plane using (4.2) producing a set of undistorted image points I_{u}. The undistorted image points were then distorted by $\delta d = [\delta x \; \delta y]$ based on the radial distortion measured above producing a set of distorted image points I_{d}. Now we have a set of control (model) points and a set of distorted image points. This is exactly the information that our method uses to compute a perspective transformation. We compute a new perspective transformation $\mathcal{P}'$ using (4.8) through (4.10). The effect of the linear approximation on an arbitrary three dimensional point M can be expressed as follows:

$$\bigtriangledown \;=\; \|(M\mathcal{P} + \delta d) - M\mathcal{P}'\|_2 \qquad \text{(A.5)}$$

Figures A-3 and A-4 show two plots of $\bigtriangledown$ for a plane parallel to the image plane and passing through the control points. The five spikes mark the image locations of the five control points. Notice that globally $\bigtriangledown$ is no worse than the raw radial distortion. Further, $\bigtriangledown$ is small in the vicinity of the control points even when the control points are located at the edge of the image. These two characteristics were true for all of the simulations that we ran. This confirms the claim that our method works well if the radial distortion is not *too great* or the enhancement is close to the fiducials.

Finally, we performed an experiment using a lens with significant distortion. Figure A-5 shows an image taken using a 4.8mm lens. The test object is the same one which appeared in earlier figures. The distortion is readily apparent. Figures A-6 through A-8 show the results of our method without correcting for distortion.

In general, a more accurate model produces more accurate results. Correcting for radial distortion undoubtedly would improve the results of our method. However, as shown in this appendix for enhanced reality visualization using reasonable cameras in realistic viewing situations the improvement is almost insignificant.

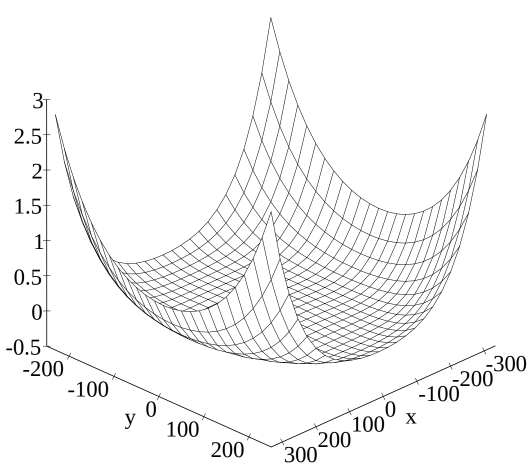

Figure A-1: Radial distortion in pixels for a 16mm lens.

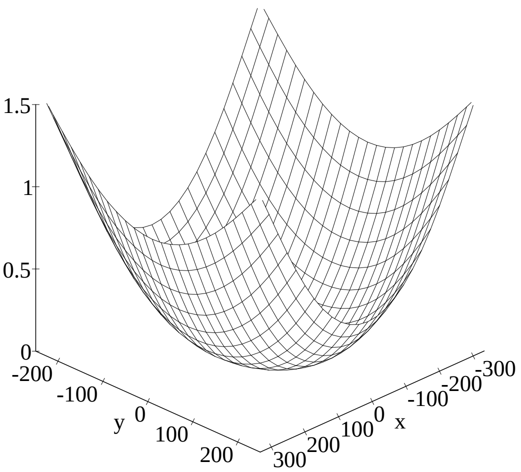

Figure A-2: Radial distortion in pixels for a 25mm lens.

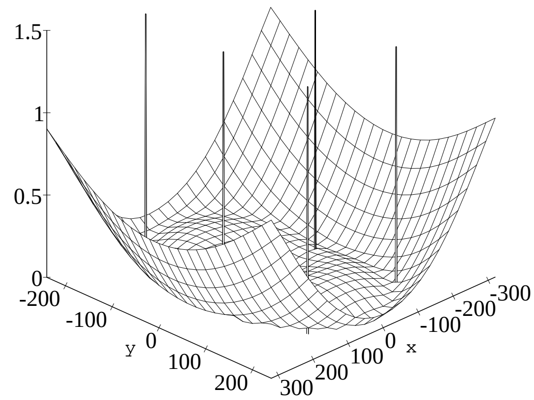

Figure A-3: Effect of radial distortion on our method with the fiducials near the center of the image.

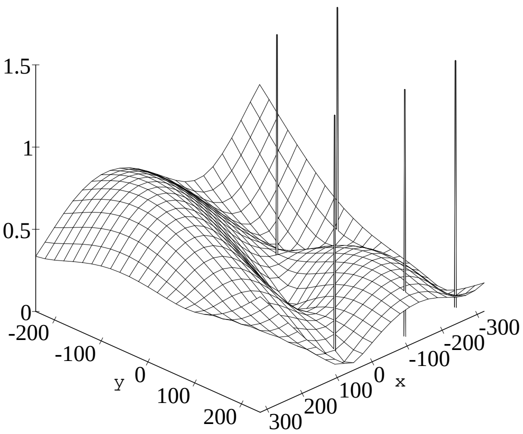

Figure A-4: Effect of radial distortion on our method with the fiducials near the edge of the image.

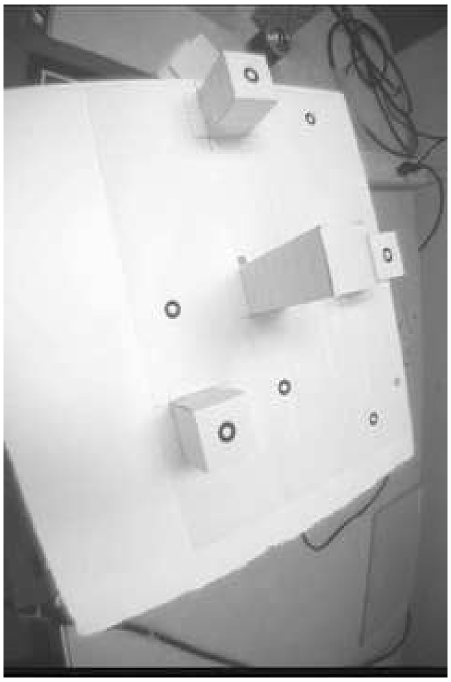

Figure A-5: Image with significant distortion.

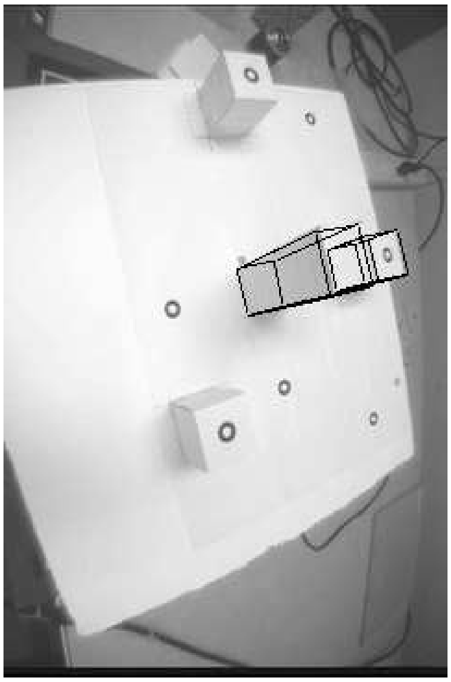

Figure A-6: Distorted image view 1.

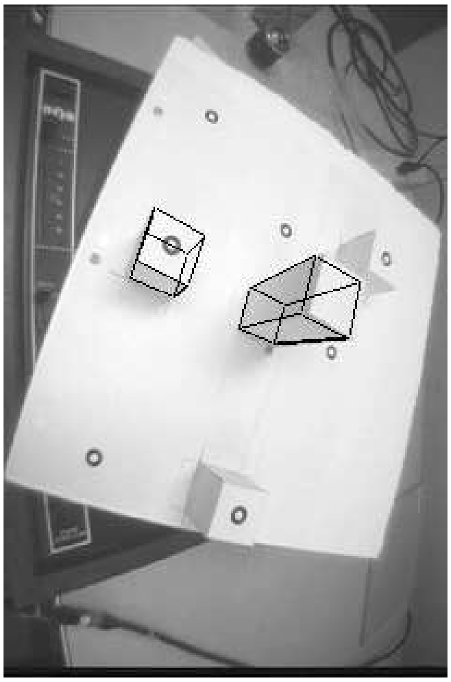

Figure A-7: Distorted image view 2.

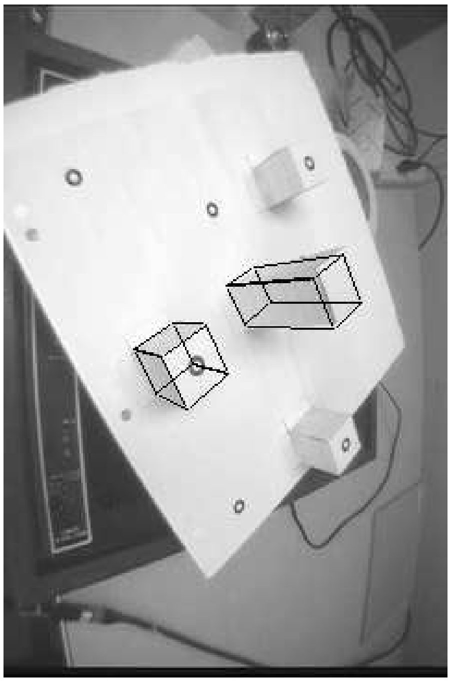

Figure A-8: Distorted image view 3.

Bibliography

[Adams *et al.*, 1990] Ludwig Adams, Joachim M. Gilsbach, Dietrich Meyer-Ebrecht Werner Krybus, Ralph Mosges, and Georg Schlondorff. CAS - a navigation support for surgery. In K. H. Hohne et al., editor, *3D Imaging in Medicine*, volume 60 of *NATO ASI, F*, pages 411–423. Heidelberg, 1990.

[Azuma and Biship, 1994] Ronald Azuma and Gary Biship. Improving static and dynamic registration in an optical see-through HMD. In *Computer Graphics*, pages 197–204. ACM SIGGRAPH, July 1994. Orlando, Fl.

[Bajura *et al.*, 1992] Michael Bajura, Henry Fuchs, and Ryutarou Ohbuchi. Merging virtual objects with the real world: Seeing ultrasound imagery within the patient. In *Computer Graphics*, pages 203–210. ACM SIGGRAPH, July 1992.

[Bemmel *et al.*, 1985] Jan H. Van Bemmel, Francois Gremy, and Jana Zvarova, editors. *Medical Decision Making: Diagnostic Strategies and Expert Systems*, New York, October 1985. IFIP-IMIA, North-Holland. Prague, Czechoslovakia.

[Black *et al.*, 1993] P. Black, R. Kikinis, W. Wells, D. Altobelli, W. Lorensen, H. Cline, and F. Jolesz. A new virtual reality technique for tumor localization. In *Congress of Neurological Surgeons*, 1993.

[Bose and Amir, 1990] C.B. Bose and I. Amir. Design of fiducials for accurate registration using machine vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(12):1196–1200, December 1990.

[Brown, 1965] Duane C. Brown. Decentering distortion of lenses. *Photogrammetric Engineering*, 32(3):444–462, 1965.

[Brown, 1971] Duane C. Brown. Close-range camera calibration. *Photogrammetric Engineering*, 37(8):855–866, 1971.

[Caudell and Mizell, 1992] Thomas P. Caudell and David W. Mizell. Augmented reality: An application of heads-up display technology to manual manufacturing processes. In *Proceedings of Hawaii International Conference on System Sciences, Vol II*, pages 659–669. IEEE Computer Society, January 1992. Kaual HI, RR P3.20.S9.

[Chang, 1993] Ifay F. Chang. Computerized patient record and clinical information system. Technical Report RC 19164, IBM, September 1993.

[Chaudhuri and Samanta, 1991] B.B. Chaudhuri and G.P. Samanta. Elliptic fit of objects in two and three dimensions by moment of inertia optimization. *Pattern Recognition Letters*, 12(1):1–7, January 1991.

[Chiorboli and Vecchi, 1993] G. Chiorboli and G.P. Vecchi. Comments on "design of fiducials for accurate registration using machine vision". *IEEE Transactions On Pattern Analysis And Machine Intelligence*, 15(12):1330–1332, December 1993.

[Church, 1945] E. Church. Revised geometry of the aerial photograph. Bulletin of Aerial Photogrammetry 15, Syracuse University, 1945.

[Dinstein *et al.*, 1984] Its'hak Dinstein, Fritz merkle, Tinwai D. Lam, and Kwan Y. Wong. Imaging system response linearization and shading correction. In *Conference on Robotics*, pages 204–209. IEEE, March 1984. Atlanta, GA.

[Duda and Hart, 1973] Richard Duda and Peter Hart. *Pattern Classification and Scene Analysis*. John Wiley, New York, 1973.

[Efrat and Gotsman, 1993] Alon Efrat and Craig Gotsman. Subpixel image registration using circular fiducials. Technical Report TECHNION CIS 9308, Technion, Israel Institute of Technology, Center for Intelligent Systems, February 1993.

[Faig, 1975] W. Faig. Calibration of close-range photogrammetry systems: Mathematical formulation. *Photogrammetric Engineering and Remote Sensing*, 41:1479–1486, 1975. Barker.

[Faugeras and Toscani, 1987] O. Faugeras and G. Toscani. Camera calibration for three dimensional computer vision. In *Proceedings of the International Workshop on Industrial Applications of Machine Vision and Machine Intelligence*. Seiken Symposium, February 1987. Tokyo, Japan; Barker TA1632.I595.

[Feiner *et al.*, 1993] Steven Feiner, Blair MacIntyre, and Doree Seligmann. Knowledge-based augmented reality. *Communications of the ACM*, 36(7):53–62, July 1993.

[Fischler and Bolles, 1981] M.A. Fischler and R.C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–385, June 1981.

[Ganapathy, 1984] Sundaram Ganapathy. Decomposition of transformation matrices for robot vision. In *Conference on Robotics*, pages 204–209. IEEE, March 1984. Atlanta, GA.

[Goshtasby, 1987] A. Goshtasby. Correction of image deformation from lens distortion. Technical report, University of Kentucky, 1987.

[Gottschalk and Hughes, 1993] Stefan Gottschalk and John F. Hughes. Autocalibration for virtual environments tracking hardware. In *Computer Graphics*, pages 65–71. ACM SIGGRAPH, August 1993.

[Grimson *et al.*, 1994] W.E.L. Grimson, T. Lozano-Pérez, G.J. Ettinger W.M. Wells III, S.J. White, and R. Kikinis. An automatic registration method for frameless stereotaxy, image guided surgery, and enhanced reality visualization. In *Computer Vision and Pattern Recognition*, pages 430–436. IEEE, June 1994. Seattle, WA.

[Grosky and Tamburino, 1987] W. Grosky and L. Tamburino. A unified approach to the linear camera calibration problem. Technical report, University of Michigan, 1987.

[Healey and Kondepudy, 1994] Glenn E. Healey and Raghava Kondepudy. Radiometric CCD camera calibration and noise estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(3):267–276, March 1994.

[Horn, 1986] Berthold Klaus Paul Horn. *Robot Vision*. MIT Press, Cambridge, MA, 1986.

[Hussain and Kabuka, 1990] Basit Hussain and Mansur R. Kabuka. Real-time system for accurate three-dimensional position determination and verification. *IEEE Transactions on Robotics and Automation*, 6(1):31–43, February 1990.

[Huttenlocher, 1988] Danial Peter Huttenlocher. *Three-Dimensional Recognition of Solid Objects From a Two-Dimensional Image*. PhD thesis, MIT, April 1988.

[Kamgar-Parsi and Kamgar-Parsi, 1989] Behrooz Kamgar-Parsi and Behzad Kamgar-Parsi. Evaluation of quantization error in computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(9):929–940, September 1989.

[Landau, 1987] U.M. Landau. Estimation of a circular arc center and its radius. *Computer Vision, Graphics, and Image Processing*, 38(3):317–326, June 1987.

[Lavallee and Cinquin, 1990] Stephane Lavallee and Philippe Cinquin. Computer assisted medical interventions. In K. H. Hohne et al., editor, *3D Imaging in Medicine*, volume 60 of *NATO ASI, F*, pages 301–312. Heidelberg, 1990.

[Lemoine *et al.*, 1991] D. Lemoine, C. Barillot, B. Gibaud, and E. Pasqualini. An anatomical-based 3D registration system of multimodality and atlas data in neurosurgery. In *Information Processing in Medical Imaging*, pages 154–164, 1991.

[Lenz and Tsai, 1988] Reimar K. Lenz and Roger Y. Tsai. Techniques for calibration of the scale factor and image center for high accuracy 3-D machine vision metrology. *IEEE Transactions On Pattern Analysis And Machine Intelligence*, 10(5):713–720, September 1988.

[Maybank and Faugeras, 1992] Stephen J. Maybank and Olivier D. Faugeras. A theory of self-calibration of a moving camera. *International Journal of Computer Vision*, 8(2):123–151, August 1992.

[Miller, 1990] Randolph A. Miller, editor. *Symposium on Computer Applications in Medical Care.* IEEE Computer Society, November 1990. Washington, DC.

[Pelizzari *et al.*, 1991] C. A. Pelizzari, K. K. Tan, D. N. Levin, G. T. Y. Chen, and J. Balter. Interactive 3d patient-image registration. In *Information Processing in Medical Imaging*, pages 132–141, July 1991. Wye, UK.

[Penna, 1991] M.A. Penna. Camera calibration: A quick and easy way to determine the scale factor. *IEEE Transactions On Pattern Analysis And Machine Intelligence*, 13(12):1240–1245, December 1991.

[Pieper *et al.*, 1992] Steven Pieper, Joseph Rosen, and David Zeltzer. Interactive graphics for plastic surgery: A task-level analysis and implementation. In *Symposium on Interactive 3D Graphics*, pages 127–134. ACM SIGGRAPH, March 1992. Cambridge, MA.

[Reggia and Tuhrim, 1985] James A. Reggia and Stanley Tuhrim, editors. *Computer-Assisted Medical Decision Making*, volume I and II. Springer-Verlag, New York, 1985.

[Safaee-Rad *et al.*, 1992] R. Safaee-Rad, K.C. Smith, B. Benhabib, and I. Tchoukanov. Application of moment and fourier descriptors to the accurate estimation of elliptical-shape parameters. *Pattern Recognition Letters*, 13(7):497–508, July 1992.

[Schweikard *et al.*, 1994] Achim Schweikard, Rhea Tombropoulos, Lydia Kavraki, John R. Adler, and Jean-Claude Latombe. Treatment planning for a radiosurgical system with general kinematics. In *Robotics and Automation*, pages 1720–1727. IEEE, May 1994. San Diego, CA.

[Slama, 1980] C.C. Slama, editor. *Manual of Photogrammetry*. American Society of Photogrammetry, fourth edition, 1980.

[Smith *et al.*, 1991] K.R. Smith, K. Joarder, R.D. Bucholz, and K.R. Smith. Multimodality image analysis and display methods for improved tumor localization in stereotactic neurosurgery. In *Engineering in Medicine and Biology*, page 210. IEEE, 1991. Volume 3.

[Sobel, 1974] I. Sobel. On calibrating computer controlled cameras for perceiving 3-d scenes. *Artificial Intelligence*, 5:185–198, 1974.

[Stein, 1993] Gideon P. Stein. Internal camera calibration using rotation and geometric shapes. Master's thesis, MIT, February 1993. MIT/AI/TR 1426.

[Thomas and Chan, 1989] Samual M. Thomas and Y.T. Chan. A simple approach for the estimation of circular arc center and its radius. *Computer Vision, Graphics, and Image Processing*, 45(3):362–370, March 1989.

[Tsai, 1987] Roger Y. Tsai. A versatile camera calibration technique for high-accuracy three dimensional machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE Journal of Robotics and Automation*, RA-3(4):323–344, August 1987.

[Verbeeck *et al.*, 1993] R. Verbeeck, D. Vandermeulen, J. Michiels, P. Suetens, G. Marchal, J. Gybels, and B. Nuttin. Computer assisted stereotactic neurosurger. *Image and Vision Computing*, 11(8):468–485, October 1993.

[Wang *et al.*, 1990] Jih-fang Wang, Vernon Chi, and Henry Fuchs. A real-time optical 3d tracker for head-mounted display systems. In *Symposium on Interactive 3D Graphics*, pages 205–215. ACM SIGGRAPH, March 1990. Snowbird, Utah.

[Ward *et al.*, 1992] Mark Ward, Ronald Azumaand Robert Bennett, Stefan Gottschalk, and Henry Fuchs. A demonstratedd optical tracker with scalable work area for head mounted display systems. In *Computer Graphics*, pages 43–52. ACM SIGGRAPH, March 1992. Cambridge, MA.

[Watkins, 1991] D. Watkins. *Fundamentals of Matrix Computations*. John Wiley and Sons, Inc., New York, 1991.

[Wells *et al.*, 1993] W. Wells, R. Kikinis, D. Altobelli, G. Ettinger W. Lorensen, H. Cline, P. L. Gleason, and F. Jolesz. Video registration using fiducials for surgical enhanced reality. In *Engineering in Medicine and Biology*. IEEE, 1993.

[Willson and Shafer, 1993] Reg G. Willson and Steven A. Shafer. What is the center of the image? Technical Report CMU-CS-93-122, Carnegie-Mellon University, Computer Science Department, April 1993.