Fast and Accurate Shape-Based Registration

David A. Simon

CMU-RI-TR-96-45

December 12, 1996

Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in the field of Robotics

Thesis Committee:

Takeo Kanade, Chair
Martial Hebert
Michael Erdmann
Eric Grimson, MIT

Keywords: shape-based registration, computer-assisted surgery, registration accuracy validation, geometric constraint analysis, geometric constraint synthesis

Dedicated to Lou and Sybil for starting me on this journey,
and to Nanny Frieda for reminding me to enjoy the ride.

Abstract

In many areas of science, engineering, medicine, and especially in the field of robotics, there is a need to establish a spatial mapping between two or more 3-dimensional (3-D) shape representations of an object. The *registration problem* is concerned with finding a spatial transformation which best aligns two object representations. Once this mapping is established, a variety of tasks can be performed using the aligned object representations including model-based localization, 3-D object recognition, real-time pose tracking and multi-modality sensor fusion.

The research goals of this dissertation are the design, implementation and validation of *fast* and *accurate* methods for performing 3-D shape-based registration. Fast registration is achieved via speed enhancements to an existing registration method (the iterative closest point algorithm). The implemented enhancements increase registration speeds by a factor of nearly two orders of magnitude, and the resulting speeds are much higher than those previously reported in the literature. The ability to perform high-speed registration enables completely new registration applications. For example, this dissertation describes a system for tracking the position and orientation of arbitrarily-shaped objects at speeds of roughly 10 Hz using a high-speed range finder.

The majority of this dissertation focuses on the problem of achieving and ensuring high accuracy shape-based registration. This goal is addressed via two processes: intelligent data selection and online accuracy estimation. Intelligent data selection is based on the observation that all data are not equally beneficial for performing accurate registration. Judicious selection and careful collection of a limited amount of data can result in better registration accuracy than random use of larger amounts of data. In this dissertation, two techniques are proposed for performing intelligent data selection: an optimization process (Constraint Synthesis), and a numerical analysis method which provides a criterion measure for the optimization process (Constraint Analysis). Registration experiments demonstrate that data configurations which are automatically synthesized using these methods can significantly increase registration accuracy.

A fundamental problem in registration is knowing whether an estimated transformation satisfies accuracy requirements *at the time of registration*. A conventional root-mean-squared error, coupled with the criterion measure from constraint analysis provides the ability to estimate true registration accuracy online. This estimate can be used to guide the online collection of registration data by determining when additional data are required to satisfy accuracy requirements.

While there are many areas to which registration can be applied, the chosen domain for this dissertation is medicine, and most of the results are presented in the context of computer-assisted surgery. Despite the specific nature of the chosen application area, the underlying ideas and methods have application to a broad range of registration problems.

.

Acknowledgments

There are many individuals who have provided guidance, friendship and support throughout my years as a graduate student and to whom I owe thanks. First among them are my thesis committee members. My advisor, Takeo Kanade, has been a mentor and friend who has taught me many things about life. In particular, Takeo has taught me what it means to be an independent researcher, and perhaps most importantly, the significance of clear and precise communication of one's research results. He has also provided the wonderful environment in which I have been able to conduct my work. Martial Hebert has, in many ways, been a second advisor during my Ph.D. studies. Some of the more mathematical ideas within this dissertation would never have evolved without the many enjoyable discussions with Martial. Mike Erdmann has been an excellent sounding board for my ideas and was very helpful at getting me to think about my work in a broader context. Eric Grimson was the member of my committee with the most experience in the domain of computer-assisted surgery. For that reason, Eric's encouragement and feedback were particularly important since he understood the potential utility of this work in a domain in which we both share a strong interest.

Much of the fun in conducting research is interacting with interesting people along the way. Some people provide inspiration by example, some spark new ideas through informal discussions, and some provide support and encouragement. Bob O'Toole, Fritz Morgan and Mike Blackwell have been good friends and co-workers in the medical robotics lab who have each inspired me in their own way. Tony DiGioia has been an overflowing source of medical-related wisdom, and is the best boss a person could hope for while trying to simultaneously complete a Ph.D. and work full time. Kevin Dowling snaked his way through an early draft of this document, and has been a friend throughout the years. Andrew Johnson has provided insightful feedback on my work, and has been a source of interesting office conversation, both technical and otherwise. Shumeet Baluja and I worked together to apply his PBIL method to the Constraint Synthesis problem described in this dissertation. Sarah Gibson has been a friend for many years, and in part, I have her to thank for directing me to the medical domain. David LaRose read drafts of several sections of the dissertation and had some excellent feedback. More than one of the illustrations in this dissertation originated from the pen of Branko Jaramaz. I have great memories of early research experiences with Lee Weiss and Shree Nayar, both of whom have provided much counsel over the years. Andy Gruss provided assistance with his high-speed VLSI range sensor, a fundamental component of the tracking system described in the dissertation. MaryJo Dowling and Jim Kocher have provided artistic, photographic, and video assistance time and again. Cheryl Aston did a souper job of proof-reeding this document. Rich Wallace provided CT scanning services and Chuck Mandella provided anatomic specimens. There are many other friends and associates who have had an influence on my work and life over the years including:

Mark Wheeler, Eric Kischell, Joyoni Dey, Julio Rosenblatt, Jim Moody, Joni Ropelewski, Bruce Colgan and many others whom I’m surely forgetting.

My family, immediate and distant, has always been a source of support, encouragement and love. Thanks Mom, Dad, Alice, Danny, Kira, Edith, Julie and Sandy. I love you all.

Finally, I would like to thank the two people in my life who bring me more joy than a person deserves, and who have both put up with more long, daddy-less nights than I care to recall. Jesse, you make me smile and are a constant reminder to me of what is truly important in life. Andi, your support and love are incredible sources of strength for me. Thank you for sharing them. It never ceases to amaze me that even as the years go by, we are still truly blessed.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

In many areas of science, engineering, medicine, and especially in the field of robotics, there is a need to establish a spatial mapping between two or more 3-dimensional (3-D) shape representations of an object. The *registration problem* is concerned with finding a spatial transformation which best aligns two object representations. Once this mapping is established, a variety of tasks can be performed using the spatially aligned object representations. Example tasks involving registration include:

- Model-based localization: given a geometric model of an arbitrarily-shaped object and the specification of a particular point (or set of points) within this model, it is possible to accurately locate the desired point(s) on the corresponding *physical* object. To perform this task, the geometric object model must be spatially registered to the physical object.
- 3-D object recognition: in the field of computer vision, the problem of recognizing and locating an object from 3-D image data has been well researched. A central component in many approaches to this problem is performing 3-D registration.
- Real-time pose tracking: tracking the pose (position and orientation) of arbitrarily-shaped objects in real-time requires the ability to perform registration at very high speeds.
- Multi-modality sensor fusion: in certain areas, complementary sensor data describing an object is available from multiple sensing modalities. Via registration, this complementary data can be fused for subsequent processing or visualization. Display of spatially aligned multi-modality data provides a human viewer with more information than independent display of the individual data sets.

The research goals of this dissertation are the design, implementation and validation of *fast* and *accurate* solution methods for 3-D shape-based registration. The goal of fast registration is addressed via a high-speed registration solver which is demonstrated by tracking the pose of arbitrarily-shaped objects at speeds of roughly 10 Hz. The goal of accurate registration is addressed via methods for the intelligent selection and analysis of registration data, and the online estimation of registration accuracy.

1.1 Motivation

While there are many areas to which registration can be applied, the chosen domain for this dissertation is medicine, and most of the results are presented in the context of computer-assisted surgery. Despite the specific nature of the chosen application area, the underlying ideas and methods presented in this dissertation have application to a broad range of registration problems.

The area of medical robotics and computer-assisted surgery (MRCAS) has emerged within the past five years as a novel field of investigation as evidenced by the many conferences, books and journals in the area [Ayache, 1995] [Ayache and Duncan, 1995] [Bucholz, 1995] [DiGioia et al., 1994] [DiGioia and Taylor, 1995] [Dohi, 1995] [Maciunas, 1993] [Satava et al., 1995] [Taylor et al., 1995]. Research in this area combines sensing, mechanical actuation, computer graphics, image processing, simulation, and high performance computing to address problems in the planning and execution of surgical tasks.

Figure 1-1 illustrates a typical application in computer-assisted orthopaedic surgery. The top half of the figure contains tasks which are performed offline (i.e., pre-operatively), while the bottom half depicts tasks performed online (i.e., intra-operatively). For this application, selected portions of the patient's anatomy are imaged using a sensor such as a computed tomographic (CT) scanner prior to surgery. The resulting images are cross sections of the anatomy which can be stacked to create a 3-D data volume in which image intensity is proportional to density of the corresponding anatomy. These images are used to generate a plan of the procedure which will be subsequently executed in the operating room. The pre-operative plan is constructed by a physician using computer-aided graphical planning tools. Example tasks in computer-assisted orthopaedic surgery include: placement of devices such as acetabular and femoral implants in total hip replacement surgery [DiGioia et al., 1995] [Taylor et al., 1994], placement of implants in total-knee replacement surgery [Lea et al., 1995b], or placement of screws into the vertebral pedicles in spine fusion surgery [Lavallee et al., 1994] [Nolte et al., 1994]. For such tasks, the pre-operative plan specifies the placement of the relevant hardware (e.g., prostheses, screws) relative to the anatomy in the medical images. Thus, the desired location of the hardware is known relative to the coordinate system of the pre-operative medical images.

In the operating room, precise execution of the tasks specified by the pre-operative plan must be achieved. There are many task execution methods in computer-assisted surgery which differ in the amount of direct control which a surgeon has over the underlying surgical tools. The most autonomous mode of computer-assisted surgical execution arises when a robot provides the actuation forces to perform tasks such as drilling, milling or cutting of the patient's anatomy. A less autonomous alternative is passive navigational guidance in which feedback is provided to the surgeon in the form of computer graphics or medical images.

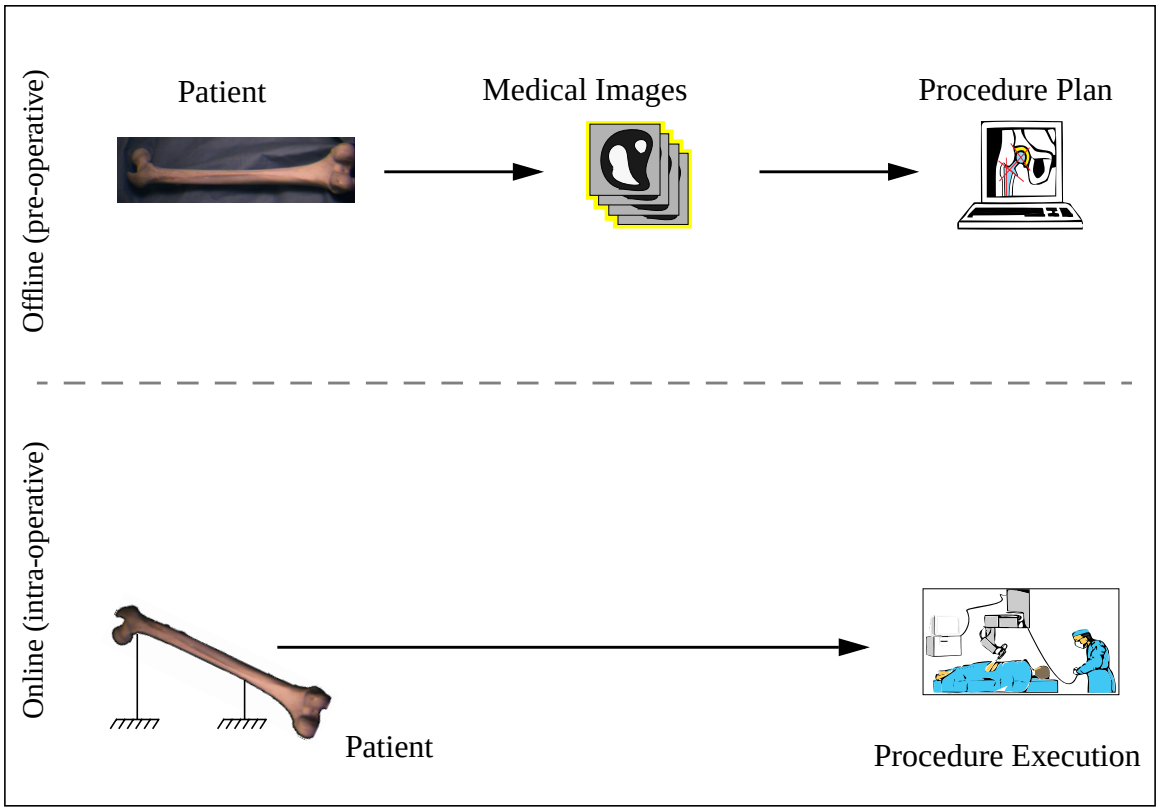


Figure 1-1: Offline planning and online execution in computer-assisted surgery. The procedure plan specifies or implies a set of actions which must be precisely executed during surgery.

This feedback is derived from the pre-operative plan and the locations of conventional surgical tools which are tracked in real-time. Regardless of the mode of surgical task execution, a fundamental problem must be solved before the pre-operative plan can be executed. Recall that the pre-operative plan is constructed in a coordinate system attached to the pre-operative medical images. Surgical execution, on the other hand, is performed in a coordinate system relative to a portion of the patient's anatomy which is assumed to be rigidly fixtured in space. Therefore, before the plan can be executed it is necessary to establish a transformation which maps points in the pre-operative plan into corresponding points on the patient. Establishing this transformation requires solution of the registration problem.

Fiducial-based registration is a method which has been applied to several application areas including neurosurgery [Maciunas, 1993] and orthopaedic surgery [Taylor et al., 1994]. In fiducial-based registration, markers are rigidly affixed to the relevant anatomy before acquisition of the pre-operative medical images, as shown in Figure 1-2. The location of these markers can then be precisely extracted from the medical images. In the operating room, the locations of these markers can be determined using a sensing device such as an optical digitizing probe. Since the correspondences between the pre- and intra-operative marker locations are known, it is possible to find a transformation which minimizes an error measure

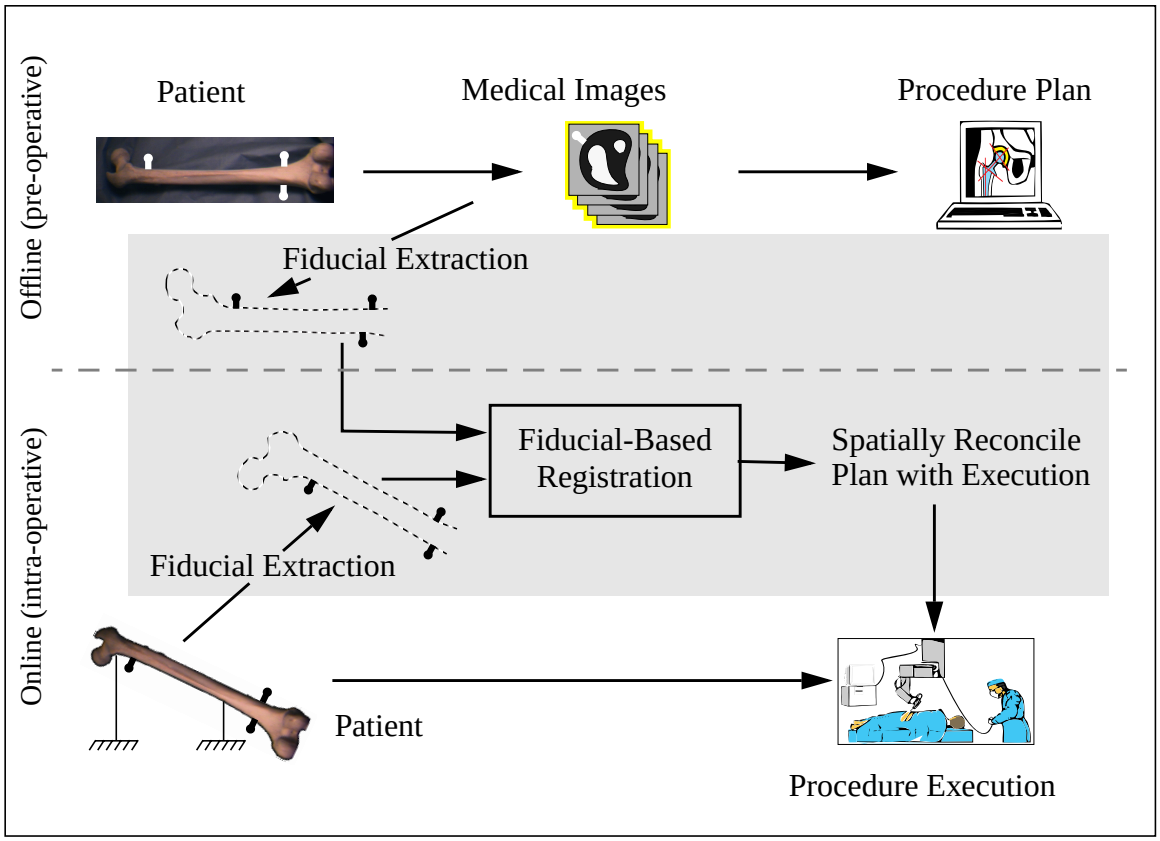


Figure 1-2: Fiducial-based registration of a pre-operative plan to intra-operative execution. Fiducial markers attached to the registration object can be extracted in the pre- and intra-operative coordinate systems, and used to estimate the required registration transformation.

between the corresponding markers. Once this transformation is known, it can be used to accurately execute the pre-operative plan during surgery.

Unfortunately, the use of fiducial markers has several negative consequences. First, insertion of the markers can require a non-trivial surgical procedure prior to the primary surgery. Second, the markers can often be bulky and invasive, as in the case of stereotactic head frames used in neurosurgery. Third, since the markers are often constructed of metal, they may introduce noise or artifacts which degrade the quality of the pre-operative medical images. Finally, in most cases the markers must be physically exposed during surgery and are sometimes distant from the primary surgical site. This may cause additional trauma to the patient and add to the time required to perform the surgery.

An alternative to fiducial-based registration is to use features which are intrinsic to the underlying anatomy to perform registration. For example, if anatomical landmarks can be identified both pre- and intra-operatively, these landmarks can serve the same purpose as the above fiducial markers. In general, it is difficult to accurately and reliably identify corre-

sponding discrete point landmarks in this manner. However, in certain cases it is possible to extract information such as the shape of an anatomical object's bounding surface. As shown in Figure 1-3, if an object's boundary can be accurately extracted both pre- and intra-operatively, this information can be used to solve the registration problem. This is the approach to registration which is used in this dissertation. Techniques for solving the shape-based registration problem are described in detail in Chapter 2.

A critical step in Figure 1-3 which has received little attention in the literature is the intra-operative shape extraction process. In this step, geometric data from the relevant anatomy are collected during surgery. This is a difficult sensing task for several reasons. First, there is often limited visibility of the relevant anatomy due to obstruction from surgical tools and fixators, blood and other fluids, and occluding portions of the anatomy. Second, the data accuracy required for shape-based registration may be quite high, placing constraints on the type of sensing which can be used. In addition, any sensor used during surgery must have a proven safety record, must not harm the patient or clinicians, and must be highly reliable. Third, the data acquisition process must be fast since longer surgical times result in

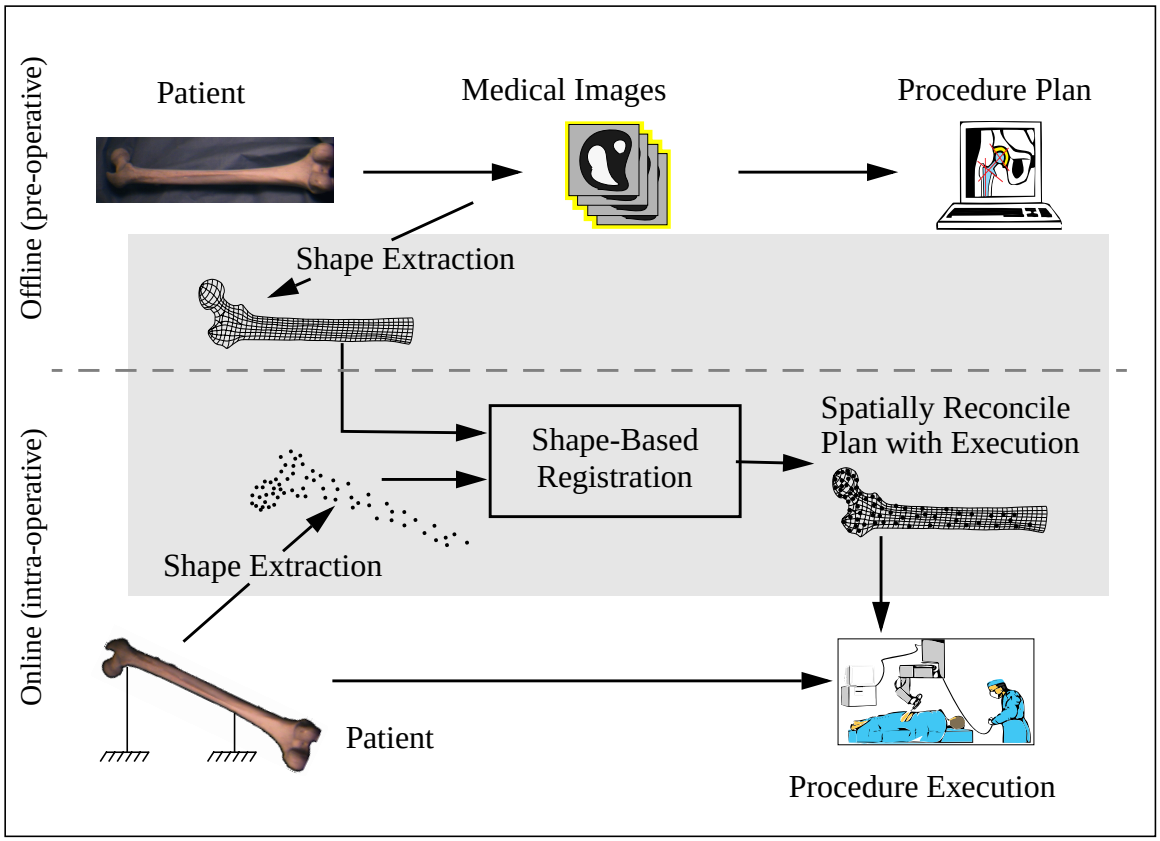


Figure 1-3: Shape-based registration of a pre-operative plan to intra-operative execution. Representations of the shape of the registration object can be extracted in the pre- and intra-operative coordinate systems, and used to estimate the required registration transformation.

increased risk of patient infection, increased blood loss, and increased monetary costs. Finally, the type of anatomical tissue to be localized places constraints on the choice of sensing modality since the associated anatomical boundaries must be visible to the sensor.

In orthopaedic surgery, the anatomy to be localized is usually bone, and the sensors available for intra-operative shape extraction include: discrete point digitizers which measure the location of a probe tip placed in contact with an object; conventional X-ray imaging which has the added cost of radiation exposure to the patient; and ultrasonic imaging which may not be sufficiently accurate for many registration tasks. Each additional datum acquired with one of these sensors may increase the overall time spent in surgery, and in the case of X-ray imaging will increase the radiation exposure as well. Therefore, it is often desirable to minimize the amount of intra-operative data required to attain sufficiently accurate registration results.

1.2 Scope of the Dissertation

The majority of this dissertation focuses on the problem of attaining and ensuring high accuracy shape-based registration results. A framework for achieving this goal is presented in Figure 1-4. The example application used in this framework is computer-assisted surgery, but as suggested earlier, the proposed approach has broad applicability to other areas. The approach consists of two phases, an offline (pre-operative) phase and an online (intra-operative) phase. The critical offline component is *Intelligent Data Selection* (IDS), which is based on the observation that all data are not equally beneficial for performing accurate registration. Judicious selection and careful collection of a limited amount of data can result in better registration accuracy than random collection of larger amounts of data. IDS consists of two components, an optimization process referred to as *Constraint Synthesis*, and a numerical analysis method which provides a criterion measure for the optimization process, referred to as *Constraint Analysis*. The criterion measure resulting from constraint analysis is a measure of expected registration accuracy as a function of a hypothesized set of registration data. The input to IDS is the surface model which will subsequently be used to perform shape-based registration. The output of IDS is a configuration of data points of fixed size called a data collection plan (DCP) which is near-optimal with respect to expected registration accuracy. The offline process can be performed as soon as the surface model becomes available, and typically requires several hours of computation time to generate the DCP.

The online process is centered around the shape-based registration solver. In the medical application, the solver requires three inputs, a pose estimate to initialize the registration process, a surface *Model* constructed from pre-operative medical images, and a set of *Data* collected during surgery.[1] In computer-assisted surgery, an initial pose estimate for shape-based registration can often be acquired using anatomical landmark-based corresponding point registration. Computationally, this method is similar to fiducial-based registration methods;

however, no fiducial markers are required. When an initial pose estimate is available, a more descriptive name for the registration process is *shape-based pose refinement*.

The Data collection process includes the physical act of collecting the particular Data points specified by the DCP from the registration object using an appropriate sensor. In the experiments of this dissertation, a human Data collector positions the sensor and oversees the Data collection process. There is, however, a potential inconsistency in this framework. The DCP specifies a configuration of points to be collected on the registration object. However, the location of the object itself is not precisely known, and therefore the points in the DCP can not be collected exactly. Fortunately, initial and incremental pose estimates from the registration process can be fed back to the collection process to help guide Data collection. In practice, a computer-graphic rendering of the surface Model can be displayed to the Data collector with the locations of the desired points highlighted. Using the pose estimates fed back from registration, it is also possible to superimpose an estimate of the current location of the Data collection sensor on the graphic rendering. This information can be used by the collector to acquire the points specified by the DCP within reasonable tolerances.

In registration applications requiring high-reliability, it is desirable to have an online measure of confidence in the resulting registration accuracy. The same constraint analysis criterion used in IDS, together with a conventional root-mean-square (RMS) registration error, can provide an online measure of confidence in registration accuracy. Using this measure it is possible to estimate whether particular registration accuracy requirements have been satisfied *as registration is being performed*. Online estimates of registration accuracy are critical in applications requiring high-reliability.

In the registration examples of Figures 1-2 and 1-3, patient anatomy at the primary surgical site is assumed to be rigidly fixtured during surgery. Fixturing prevents motion of the anatomy relative to the tools which are used to execute the surgical procedure. If motion occurs, the transformation established via registration becomes invalid and the pre-operative plan will require periodic registration for precise execution. One contribution of this dissertation is the demonstration that shape-based registration can be performed at speeds much higher than those previously reported in the literature. This capability is demonstrated via a system for tracking the pose (position and orientation) of human faces at speeds of roughly 10 Hz. The system operates by registering a surface model of a human face to discrete point surface measurements of the same face acquired using a real-time range sensor. This type of system has potential application for tracking non-fixtured patient anatomy in computer-assisted sur-

1. Throughout the dissertation, the terms “Model” and “Data” (note capitalization) refer to the object descriptions used to perform registration. This terminology was chosen to reduce ambiguity resulting from the generic usage of these words. Unless otherwise noted in the text, Model refers to a surface model and Data refers to a set of discrete surface points used in either the registration or intelligent data selection processes.

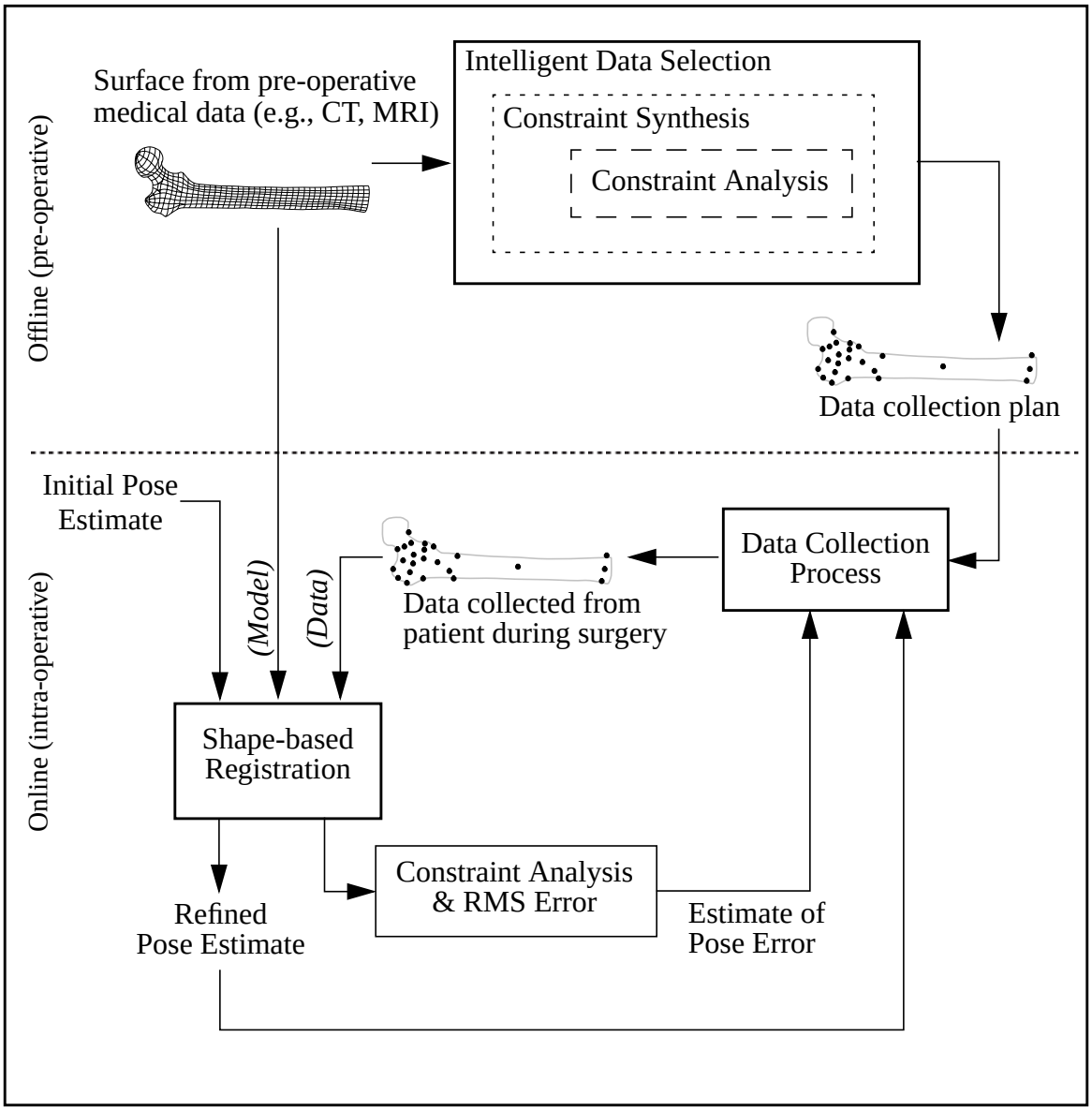


Figure 1-4: A framework for accurate shape-based registration in a computer-assisted surgical application. The offline component generates a data collection plan, a set of data points which are near-optimal with respect to expected registration accuracy. The online component uses the data collection plan to guide the collection of data from the registration object. This data is then used by the registration solver to estimate the registration pose transformation. Online estimates of the registration transformation and transformation error are fed back to the data collection process to help guide the data collection process. These online estimates also provide a mechanism for ensuring that registration accuracy requirements are satisfied.

gery. Eliminating fixturing devices in certain surgical procedures is desirable due to the invasive nature of the devices. While the tracking system demonstrated in this dissertation is not yet reliable enough for surgical applications, it may be suitable for applications in areas such as human-computer interaction and manufacturing which are not life-critical.

1.3 Dissertation Overview

The remainder of the dissertation is organized as follows.

Chapter 2 provides background on the shape-based registration problem and presents the registration solution methods used in this dissertation. Methods are discussed for handling outliers in the Data, and for dealing with convergence to non-global minima. A very high-speed registration solver which improves registration computation times by up to two orders of magnitude is presented. An application for tracking the pose of human faces at rates as fast as 10 Hz is described, with additional details provided in Appendix A. The important and difficult problem of quantitative evaluation of registration accuracy is discussed, and several measures for describing accuracy are defined. In particular, the importance of evaluating registration accuracy in a task-specific manner is explained.

Chapter 3 presents the theoretical framework and numerical formulation of the constraint analysis method. The inputs to constraint analysis are a surface Model and a configuration of Data points which lie on the surface. The output is a scalar measure which is a good predictor of registration accuracy. It is shown that constraint analysis can be computed at very high speeds for a typical Model and Data configuration (approximately 1 ms). Two fundamental problems related to constraint analysis are described: dependence upon object scale and dependence upon choice of coordinate-system. Solutions are proposed for handling these problems, and rigorous justifications are presented for the proposed solutions. The utility of constraint analysis in the context of Intelligent Data Selection is demonstrated through a variety of experimental results and illustrative examples. It is shown that the measure computed by constraint analysis provides an upper bound on registration accuracy, and that this measure can be computed quickly without the need to perform registration.

Chapter 4 describes the constraint synthesis problem which is shown to be a complex, multi-dimensional optimization problem. Several non-linear search algorithms are proposed for solving constraint synthesis. The goal of these algorithms is to maximize the criterion measure provided by constraint analysis as a function of the particular Data points which are included in a selected Data configuration. The output of the algorithms is a Data configuration which is near-optimal with respect to expected registration accuracy. As mentioned above, there is a fundamental problem associated with the effect of object pose uncertainty during the Data collection process. The implications of this problem are discussed, and methods for handling object pose uncertainty are suggested. Experimental results are pre-

sented which compare the relative performance of several constraint synthesis solution methods. Additional results demonstrate the effectiveness of the proposed solution method for generating Data configurations which exhibit superior registration accuracy. The effects of object pose uncertainty and sensor noise on registration accuracy are also quantified in these experiments.

Chapter 5 demonstrates how the proposed registration framework presented in Figure 1-4 can be applied to problems in the real world. The problem of satisfying registration accuracy requirements using Data sets of minimal size is discussed, and methods for solving this problem are proposed. It is shown that generation of minimally-sized Data sets requires the ability to estimate registration accuracy online, during the registration process. A method is proposed for performing online accuracy estimation, and experimental results which demonstrate the feasibility of the method are presented. Additional experimental results presented in this chapter provide the most convincing demonstration of the potential benefits of the proposed constraint analysis and synthesis methods. The results suggest that these methods can be used in practice to significantly improve shape-based registration accuracy.

Chapter 6 contains the conclusion. The contributions of this dissertation are summarized, and directions for future work are proposed.

Several appendices follow the conclusion. Appendix A describes the high-speed pose tracking application which demonstrates the speed with which shape-based registration can be performed. Appendix B contains a derivation related to constraint analysis which was too detailed to include in Chapter 2. Appendix C presents experimental results which illustrate the effect of scale upon constraint analysis.

Chapter 2

Shape-Based Registration

The goals of this chapter are to provide background on the shape-based registration problem, familiarize the reader with the registration solution methods used in this dissertation, and discuss measures for evaluating registration accuracy. Figure 2-1 highlights the shape-based registration component addressed in this chapter within the object localization framework presented in Chapter 1. In Section 2.1, the registration problem is described and an example problem from a medical application is presented. In Section 2.2, a review of the related literature is presented with emphasis on registration solution methods and medical applications. Section 2.3 describes the iterative closest point (ICP) algorithm which is the basis for most of the experimental registration work reported in this dissertation. Methods are discussed for handling outliers in the Data, and for dealing with convergence of the registration algorithm to non-global minima. Section 2.4 addresses the problem of performing registration at high speeds, and describes speed enhancements to the ICP algorithm which improve execution time by two orders of magnitude. The capabilities of the speed enhanced iterative closest point (EICP) algorithm are demonstrated in a high-speed pose tracking system which is described in detail in Appendix A. Section 2.5 deals with the important and difficult problem of quantitative evaluation of registration accuracy. Several measures for describing accuracy are defined, and the importance of evaluating registration accuracy in a task-specific manner is explained. Section 2.6 concludes the chapter with a brief discussion of the implications and contributions of the work presented in this chapter.

2.1 Problem Description

Registration is the problem of determining the relative pose (position and orientation) between two descriptions of the same object. These descriptions may be either geometric or photometric, and 2- or 3-dimensional. The goal of registration is to find a spatial transformation which brings the two object descriptions into alignment as measured by a suitable cost metric. The applied transformation may be either rigid or deformable, and may or may not include a scale term. The work presented in this dissertation concentrates on the problem of *rigid* registration *without scale* of *3-dimensional geometric* object descriptions.

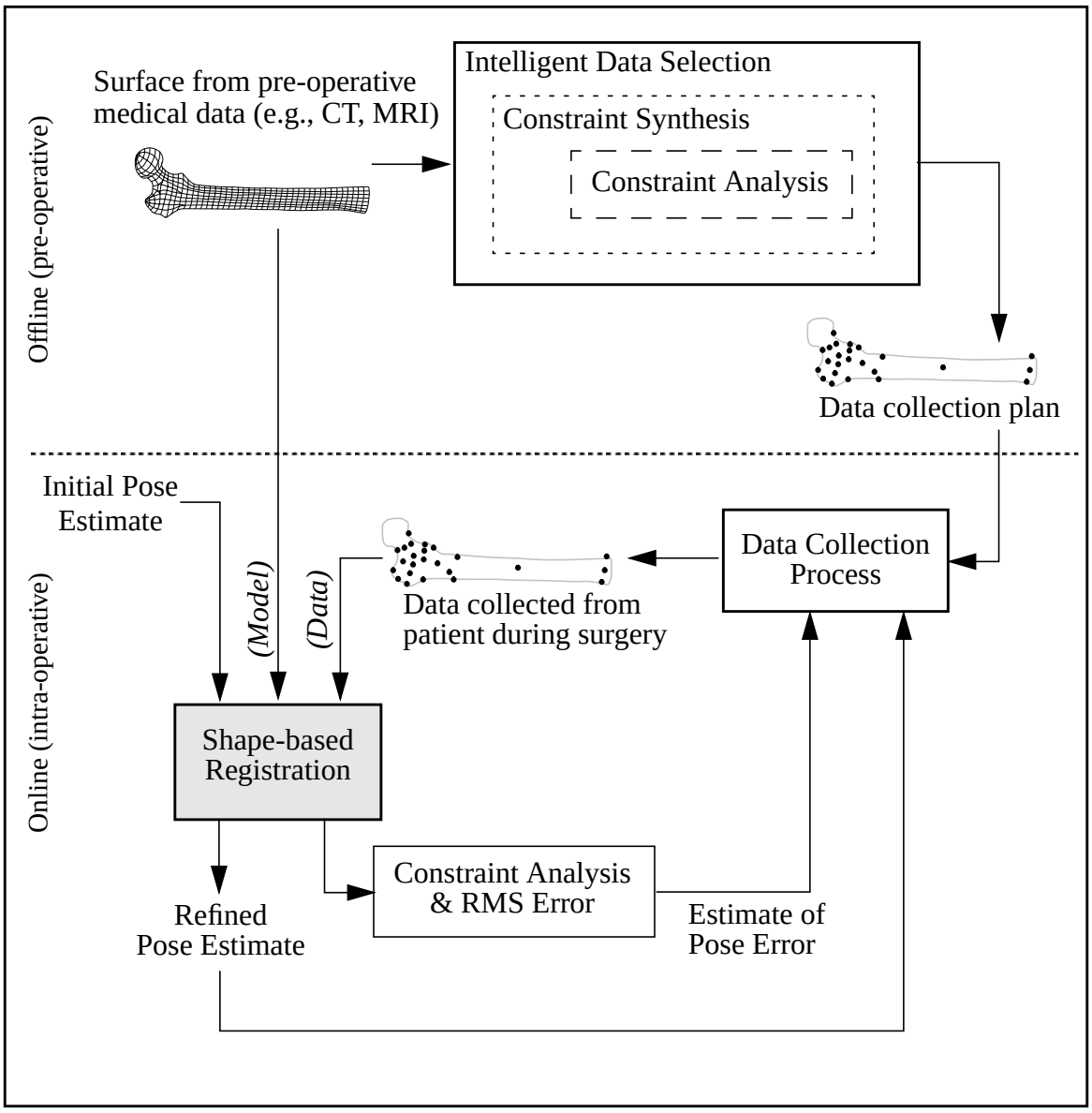


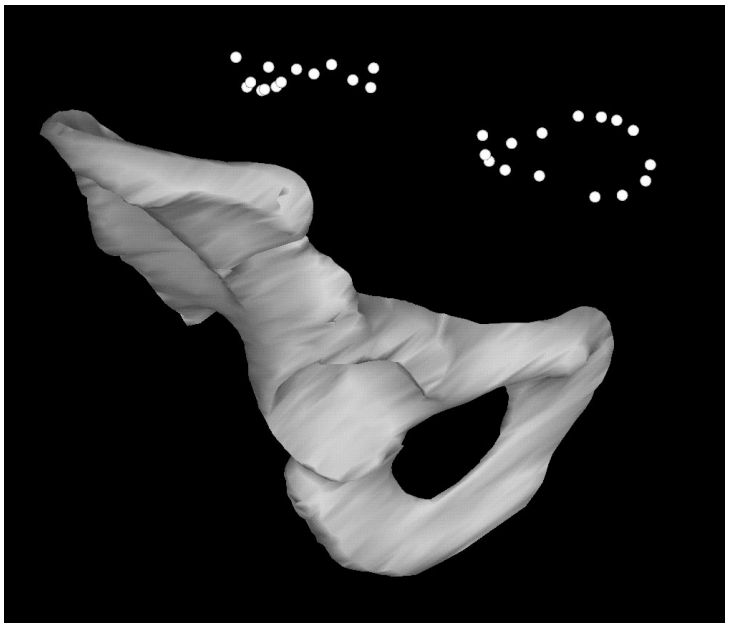
Figure 2-1: The shape-based registration component addressed in Chapter 2 (highlighted) within the object localization framework.

A geometric object description encodes information about an object's shape. Volumetric shape descriptions explicitly represent the regions of space which are occupied by an object. Boundary shape descriptions explicitly represent the transition between an object and its surround (i.e., the object's surface). The work in this dissertation uses boundary representations to describe objects. Primitives which can be used to describe object surfaces include: sets of points, sets of lines, sets of curves, sets of polygons (e.g., triangle meshes), implicit surfaces, or parametric surfaces. The experimental work presented in this dissertation uses two surface representations: triangle meshes and sets of discrete surface points; however,

most of the results can be extended to other surface representations. Throughout the dissertation, the terms “Model” and “Data” refer to the two object descriptions used to perform registration. Unless otherwise noted in the text, Model refers to a triangle mesh surface, and Data refers to a set of discrete surface points.

Figure 2-2 shows surface descriptions of a human pelvis phantom (physical object model) before and after registration. The shaded surface, corresponding to the Model, was constructed from computed tomographic (CT) images of the pelvis phantom, and the spheres,

Before:



After:

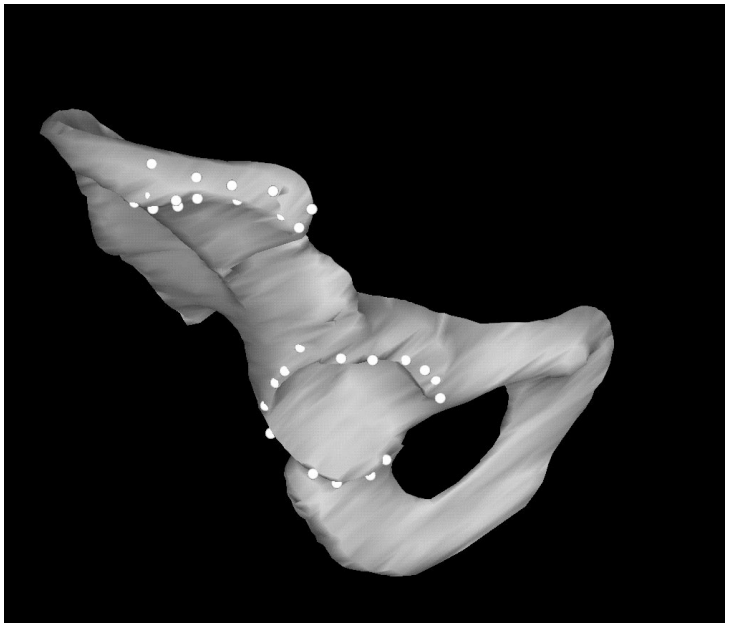


Figure 2-2: Pelvis Data and Model before and after registration.

corresponding to the Data, were collected from the same phantom using a digitizing probe. As suggested by the figure, the goal of registration is to find a spatial transformation which can be applied to the Data in order to bring it into alignment with the Model. Once this *registration transformation* has been found, it provides a mapping between any point in the Model's coordinate system and the corresponding point in the Data's coordinate system, and vice versa.

2.2 Background

This dissertation concentrates on the problem of registering two or more 3-D geometric object descriptions in which the applied transformation is rigid. Much research has been done on related 2-D and photometric registration problems. An excellent survey of 2-D photometric image registration with application to areas such as aerial photo interpretation, object recognition, and medical image processing can be found in [Brown, 1992]. Recently, 3-D photometric image registration has received considerable attention with the availability of high-quality, volumetric medical data [van den Elsen et al., 1993] [Wells et al., 1995] [Woods et al., 1993]. Registration of 2-D geometric descriptions to 3-D geometric models has been the focus of work in the computer vision literature [Fua and Leclerc, 1994] [Haralick and Joo, 1988] [Lowe, 1991] [Wheeler, 1996]. This mode of registration has been proposed in medical applications in which video images [Uenohara and Kanade, 1995] or X-ray images [Feldmar et al., 1995] [Lavallee and Szeliski, 1995] are used to derive the 2-D geometric descriptions.

A fundamental 3-D to 3-D geometric registration problem is the estimation of the relative pose between two sets of points in which point correspondences are known. There are two general categories of solution methods for this problem: quaternion-based and singular value decomposition-based (SVD-based). Both types of solution methods are formulated as least-squares minimization problems. In quaternion-based methods [Faugeras and Hebert, 1986], [Horn, 1987], rotations are represented as quaternions in order to simplify the problem of enforcing the orthonormality constraint which arises when using matrices to represent rotations. SVD-based solution methods have been demonstrated [Arun et al., 1987] [Haralick et al., 1989]. A medical application of corresponding point registration can be found in the Robodoc system which is described in [Taylor et al., 1994].

A minimum of three points is required to solve the 3-D corresponding point problem. When additional points are available, the problem is overconstrained and it becomes possible to remove outlier correspondences resulting from noisy data. Outlier elimination requires the specification of which points to eliminate. Most methods for identifying outliers use a residual error threshold which partitions the point-set into inlier and outlier correspondences. Robust statistical methods for determining appropriate residual thresholds are described in [Haralick et al., 1989], [Zhang, 1994] and [Zhuang and Huang, 1994].

A general class of 3-D to 3-D registration problems includes the registration of a Model composed of point-sets, line-sets, curve-sets or surfaces, to Data composed of one of these representations. For example, the work described in this dissertation involves the registration of point-sets to surfaces. More formally, the goal of these problems is to solve the equation:

$$\min_{t} \; d[M, t(D)] \tag{2.1}$$

where M is a description of the Model, D is a description of the Data, t is a rigid transformation, and d is a measure of similarity between the object descriptions. A solution method for this class of problems was proposed in [Besl and McKay, 1992] which describes the iterative closest point (ICP) algorithm. Since the ICP algorithm is the basis for much of the registration performed in this dissertation, a detailed description of the algorithm is presented in Section 2.3. An independently developed and very similar algorithm to ICP which was applied to registration of range images of outdoor terrain is described in [Zhang, 1994]. Prior to the above work, similar ideas were proposed but not implemented for the 3-D case in [Cox et al., 1990]. Two strengths of the ICP algorithm are that it can be applied to a variety of object representations, and that explicit correspondences are not required.

A number of other methods have been used to solve variants of the 3-D to 3-D registration problem. These methods differ in the selected cost function and in the search mechanism used to find optimal pose transformations. An excellent summary of three methods which register surfaces to point-sets is presented in [Cuchet et al., 1995] in the context of neurosurgical applications. The method proposed in [Grimson et al., 1995] uses a quasi-Newton optimization method in a coarse-to-fine registration process which tends to avoid local minima in the pose space. The methods described in [Champleboux et al., 1992] and [Lavallee et al., 1991] use a cost function which is very similar to the one used in ICP; however, unlike ICP, optimization is performed using the Levenberg-Marquardt algorithm which requires the explicit computation of gradients. In [Henri et al., 1995] a quasi-stochastic search strategy is coupled with a robust estimator in an attempt to minimize the effect of outliers. A proposed enhancement to the ICP algorithm incorporates surface normals into the cost function in an attempt to avoid local minima in the search space [Betting et al., 1995]. A surface matching technique which has been applied to multi-modal data registration is described in [Jiang et al., 1992]. A potential problem with this method is the difficulty of extracting corresponding surface data from a single anatomical surface using two different sensing modalities. In [Wheeler, 1996], a robust method is proposed for matching 3-D range images to 3-D surface models in a manner which reduces the effect of noise and outliers. This method uses M-estimation and dynamic recomputation of correspondences to achieve robustness.

In any complex multi-dimensional search problem it is possible to get trapped in local minima of the search space. In registration, there are two possible types of local minima: those close to the global minimum, and those distant from it. For the work described in this dissertation, it is assumed that the latter condition will not arise since sufficiently good initial pose estimates are available via anatomical landmark correspondences. When distant local minima are problematic, techniques which attempt to guide the search towards the global minimum can be helpful at avoiding them [Besl and McKay, 1992] [Grimson et al., 1995]. A particularly promising approach to this problem based on geometric hashing is described in [Johnson and Hebert, 1996]. In order to deal with the former type of local minima, those near the global minimum, several researchers have suggested repeating the registration process several times, each time randomly perturbing the solution about the current optimum [Cuchet et al., 1995] [Grimson et al., 1995]. A similar approach has been implemented in this work and is described in Section 2.3.

A key assumption made by ICP and other registration algorithms is that one geometric object description is a subset of the other (i.e., for each point or region described by the Data, there must be a corresponding point or region in the Model). For reasons outlined in Section 2.3, erroneous pose estimates will result when this condition is violated. Data which violate this assumption are outliers to the registration process. Outliers can result from several factors including: data collection in non-overlapping regions, noise in the data collection process, and artifacts introduced during the generation of the geometric descriptions (e.g., surface model creation artifacts). Whatever the source, it is desirable to identify and eliminate outliers during the registration process. For the ICP algorithm, robust 3-D to 3-D corresponding point methods can be used to eliminate outliers [Haralick et al., 1989] [Zhang, 1994] [Zhuang and Huang, 1994]. Some researchers have suggested the use of a least median of squares estimator [Masuda and Yokoya, 1994]; however, the computation time required to implement this technique is high. A robust approach which uses M-estimation based on the Lorentzian function to reduce the effect of outliers is described in [Wheeler, 1996]. In this approach, correspondences with small errors are weighted more heavily than those with large errors. The approach described in [Johnson and Hebert, 1996] appears to work well when the density of points in the Model and Data are roughly equal, and connectivity information is available for both. Many research groups suggest using simple thresholding to eliminate outliers in which the threshold is based upon desired accuracy or estimated noise in the data [Cuchet et al., 1995] [Grimson et al., 1995] [Henri et al., 1995] [Jiang et al., 1992] [Lavallee and Szeliski, 1995] [Zhang, 1994]. Such techniques appear to work well in practice as demonstrated later in this dissertation.

Most registration algorithms require the computation of a Euclidean distance between geometric entities of the Data and Model (e.g., distance between a point and surface). This computation is usually the most time consuming portion of the registration process. Therefore, many groups have attempted to accelerate distance computation using a variety of methods.

Chamfer matching [Barrow et al., 1977] with a pre-computed distance map has been used to accelerate registration [Cuchet et al., 1995] [Jiang et al., 1992]. A related method is described in [Lavallee and Szeliski, 1995] in which oct-tree splines are used for the same purpose. K-d trees [Friedman et al., 1977] have been used in [Zhang, 1994] to reduce the time required to calculate distances. In [Besl and McKay, 1992] an extrapolation process is proposed which uses a variation of line search optimization to speed convergence of the ICP algorithm. One of the contributions of this dissertation is the demonstration that the ICP algorithm can be run at speeds much faster than previously believed. Speed enhancements to the ICP algorithm are described in detail in Section 2.4.

There are many examples of registration applied to problems in medicine. Medical applications of registration can be divided into at least two categories: registration for data fusion and subsequent visualization, and registration for spatial localization. Applications of registration for data fusion include: multi-modality data visualization [Jiang et al., 1992] [Woods et al., 1993] in which fused data from multiple sensing modalities provide more information when viewed together than when viewed separately; and detection of anatomical changes between images of the same anatomy taken at different times [Ettinger et al., 1994]. An excellent survey of techniques and applications in medical *image-based* registration can be found in [van den Elsen et al., 1993]. The second registration category is registration for spatial localization, and is the type of registration investigated in this dissertation. In these applications, registration can be used to estimate the spatial location of a portion of a patient's anatomy with respect to a representation of the same anatomy (e.g., constructed from pre-operative medical images). An excellent survey of this type of registration can be found in [Lavallee, 1995], while a more general review appears in [Maurer and Fitzpatrick, 1993]. In computer-assisted surgery, registration has been used to perform tasks such as: guiding robotic tool movements [Kwoh et al., 1988] [Taylor et al., 1994], guiding or constraining a surgeon's tool movements [DiGioia et al., 1995] [Galloway and Maciunas, 1990] [Lavallee et al., 1994] [Nolte et al., 1994] [Radermacher et al., 1994], superimposing graphical overlays of internal anatomy upon a surgeon's view of the patient [Grimson et al., 1994], or guiding the position of radiosurgical equipment [Schweikard et al., 1994].

Recently, two research groups have described systems which employ surface-based registration techniques for an orthopaedic application [Lavallee et al., 1994] [Nolte et al., 1994]. Both groups describe systems for planning and executing the insertion of screws into the pedicle component of human vertebrae. Both employ registration techniques similar to that described in this dissertation. In particular, they register 3-D surfaces that were derived from pre-surgical CT images, to discrete point data from a coordinate measuring device. In addition, Nolte et al. perform a validation of the errors resulting from registration by comparing their surface-based results to a high accuracy approximation of ground-truth. The analyses

and techniques described in this dissertation should be useful for improving the registration accuracies in applications such as these.

One distinction of medical registration is the type of data which is used. Medical sensing modalities include computed tomographic (CT) imaging, magnetic resonance imaging (MRI), X-ray imaging, ultrasonic imaging, positron emission tomographic (PET) imaging, and single photon emission computed tomographic (SPECT) imaging. Most of these sensing modalities provide 3-D volumetric data. In addition, conventional sensing modalities such as video imaging, optical range imaging and 3-D point digitizers have also been used in medical localization problems. Examples of registration for localization in medical applications include: registration of CT or MRI images to computational stereo images [Betting et al., 1995] [Henri et al., 1995], registration of CT images to X-ray images [Feldmar et al., 1995] [Hamadeh et al., 1995] [Lavallee and Szeliski, 1995], registration of CT or MRI images to optical range data [Grimson et al., 1995], registration of CT images to digitizer data [Lavallee et al., 1995] [Nolte et al., 1994], corresponding-point registration of landmarks extracted from CT to digitizer-based landmark measurements [Lea et al., 1995b] [Taylor et al., 1994], registration of CT or MRI images to ultrasound data [Troccaz et al., 1994], and registration of SPECT images to MRI images via an intermediate step of registering both to optical range sensor data [Peria et al., 1994].

An interesting approach to registration is presented in [Radermacher et al., 1994] in which physical templates constructed using rapid manufacturing techniques based on pre-operative medical images are used to localize patient anatomy. In this approach, when a template is physically mated to the relevant anatomy, the pose of the anatomy relative to the template is known. Thus, the template provides a known reference frame in which to execute a medical procedure. In [Lea et al., 1995a], an approach for diagramming the structure of spatial relationships in complex registration systems is presented. This approach provides a common language for researchers to communicate the structure of systems involving registration.

2.3 A Registration Solution Method

The corresponding point registration problem is illustrated in Figure 2-3. Given two independently measured point-sets and associated correspondences, the problem is to find a transformation which minimizes a cost function. For the work described in this dissertation, the cost function to be minimized is the following least-squared distance metric:

$$\min_{\boldsymbol{R},\,\boldsymbol{T}} \sum_i \left\| M_i - (\boldsymbol{R} D_i + \boldsymbol{T}) \right\|^2 \tag{2.2}$$

where D_i represents points in the Data, M_i represents points in the Model, and the goal is to find a rotation, $\boldsymbol{R}$, and translation, $\boldsymbol{T}$, which minimize the least-squared distance between the

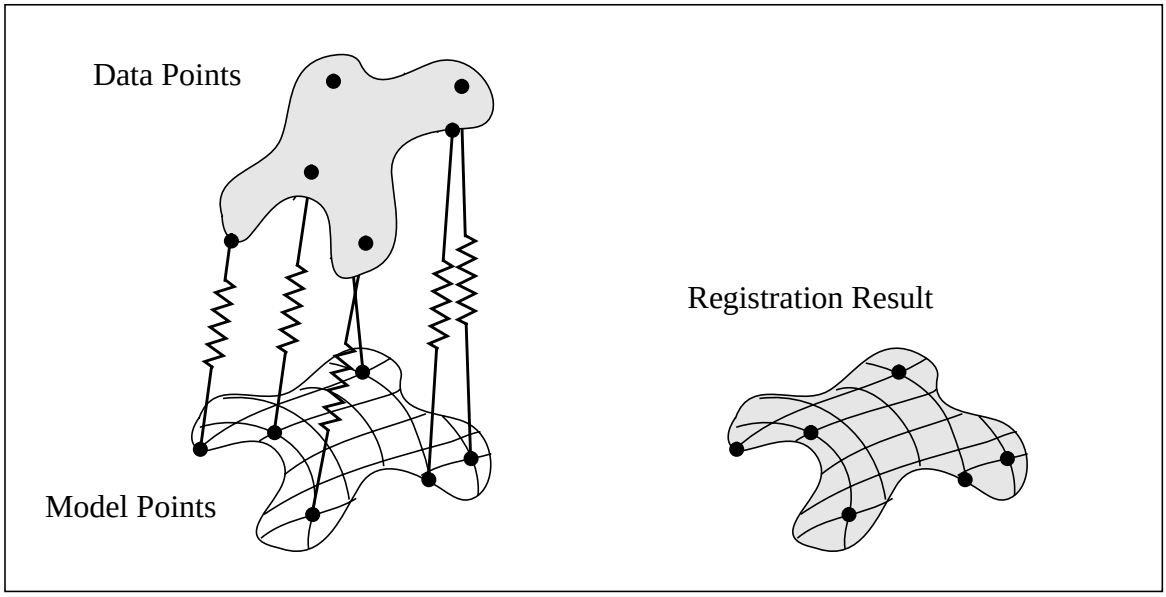


Figure 2-3: The corresponding point registration problem.

points. In a surgical application, the M_i might correspond to the locations of fiducial markers extracted from pre-operative CT data, while the D_i might correspond to locations of the same markers measured during surgery with a digitizing probe. A physical analogy for this problem is two sets of points in which corresponding pairs are connected by springs. The goal of registration is to minimize the total potential energy in the springs. This problem can be solved directly (i.e., non-iteratively) using either quaternion-based methods [Faugeras and Hebert, 1986] [Horn, 1987] or singular value decomposition (SVD) based methods [Arun et al., 1987] [Haralick et al., 1989]. Using these methods, it is also possible to weight each of the terms in the summation of Equation 2.2 by a constant coefficient. This is useful if the uncertainties associated with the data are known and are not constant. Extending the above spring analogy, the addition of weights to corresponding point registration is equivalent to connecting the points with springs of different spring constants.

2.3.1 The ICP Algorithm

For the more general class of 3-D to 3-D registration problems, the (D_i, M_i) correspondences are unknown a priori, and the object descriptions may not be composed of point-sets. The iterative closest point (ICP) algorithm proposed in [Besl and McKay, 1992] is one approach for solving this class of problems. An outline of the algorithm is presented in Figure 2-3.

1. Initialize the cumulative transformation parameters $\boldsymbol{R}$ and $\boldsymbol{T}$ to the identity transformation. Reset the iteration counter, k, to zero.
2. For each discrete point D_i in the Data set, compute the *closest* point (in terms of Euclidean distance) M_i which lies on the surface of the Model.

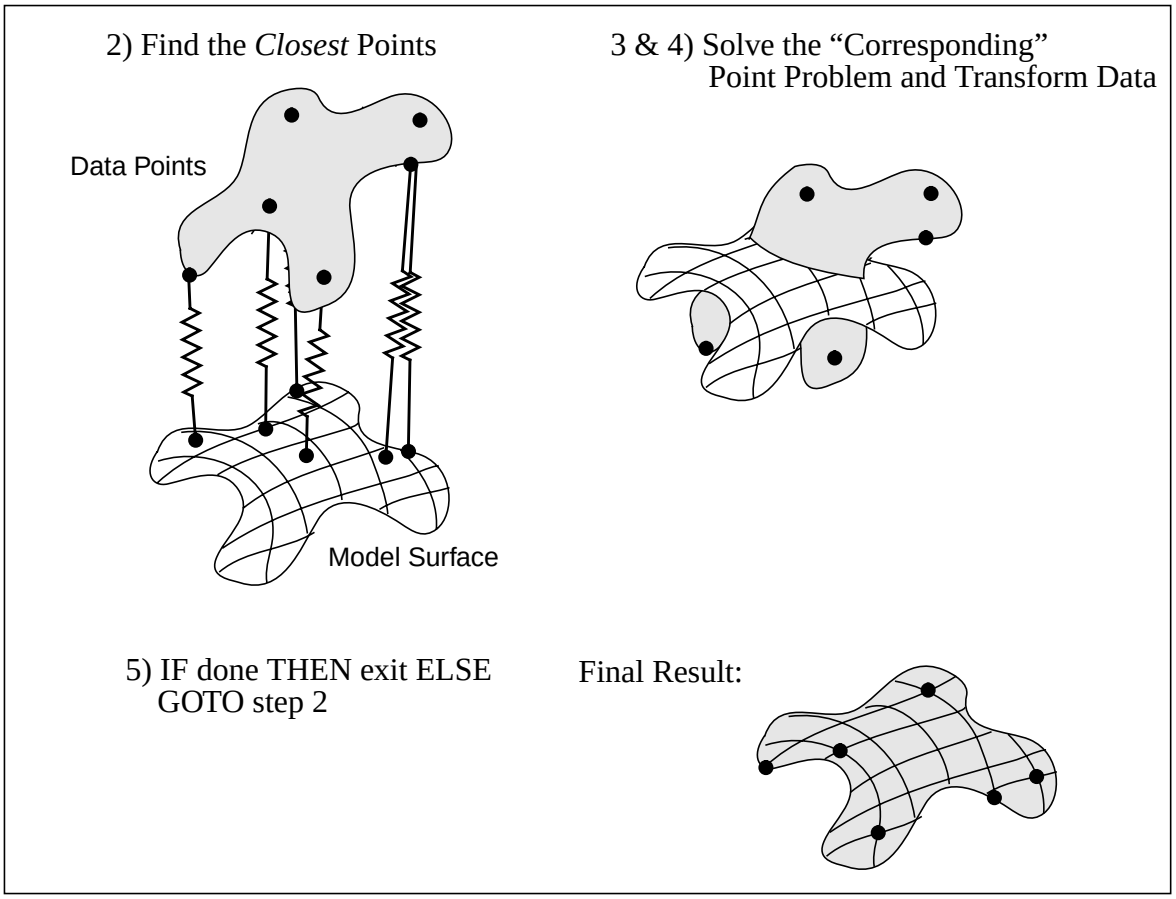


Figure 2-4: Outline of the iterative closest point (ICP) algorithm. Numbers refer to the steps in the description of the algorithm above.

3. Using the correspondences from step 2, find a rotation, $\boldsymbol{R}_k$, and a translation, $\boldsymbol{T}_k$, which minimize Equation 2.2 via the corresponding point registration method described in [Horn, 1987].
4. Apply the incremental transformation from step 3 to all Data points, D_i. Update the cumulative transformation parameters $\boldsymbol{R}$ and $\boldsymbol{T}$ based upon the incremental transformations, $\boldsymbol{R}_k$ and $\boldsymbol{T}_k$.
5. If a stopping criterion is satisfied, terminate, else goto step 2.

There are several stopping criteria which can be used with ICP. The particular stopping criteria used in this dissertation include:

i. Stop if the incremental rotation and translation relative magnitudes are both less than thresholds:

$$\frac{|\boldsymbol{R}_k|}{|\boldsymbol{R}|} < \varepsilon_{Rr} \text{ and } \frac{|\boldsymbol{T}_k|}{|\boldsymbol{T}|} < \varepsilon_{Tr} \tag{2.3}$$

(Refer to Section 2.5.1 for a description of how the rotation magnitudes are computed.)

ii. Stop if the incremental rotation and translation absolute magnitudes are both less than thresholds:

$$|\boldsymbol{R}_k| < \varepsilon_{Ra} \text{ and } |\boldsymbol{T}_k| < \varepsilon_{Ta} \tag{2.4}$$

iii. Stop if the change in residual error from Equation 2.2 is less than a threshold:

$$\sum_i \|M_i - (\boldsymbol{R}D_i + \boldsymbol{T})\|^2 < \varepsilon_{LSE} \tag{2.5}$$

iv. Stop if the total number of iterations exceeds a threshold: $k > k_{max}$. This condition is usually logically or'ed with one of the above.

Unless otherwise noted, the stopping criterion used in this dissertation is the relative transformation magnitude criterion (labeled i. above). Suitable values for ε_{Rr} and ε_{Tr} have been determined empirically to be roughly 10^{-5}. Reducing these values further has minimal effect upon registration accuracy.

Many registration methods use a cost function similar to the one in Equation 2.2. In [Hauser and Taylor, 1990] it is suggested that if the Model and Data are both surface descriptions, then solving Equation 2.2 is approximately equivalent to minimizing the volume between the surfaces. While the precise formulation of cost functions used in other methods may differ [Grimson et al., 1995] [Henri et al., 1995], almost all methods include a term expressing the distance between the Model and Data similar to the one in Equation 2.2. Variations in cost function formulation include the incorporation of weighting terms:

$$\sum_i w_i \|M_i - (\boldsymbol{R}D_i + \boldsymbol{T})\|^2 \tag{2.6}$$

where w_i are constants which weight the importance of individual correspondences based upon prior knowledge such as estimates of noise. Inclusion of surface normals into the cost function has also been proposed [Betting et al., 1995]. Since many different approaches to registration use similar cost functions, the techniques described in this dissertation are not specific to the ICP algorithm.

2.3.2 Local Minima Suppression

The ICP algorithm works quite well, especially when an approximate pose estimate is available for initialization. In general, there is no guarantee that the ICP algorithm will converge

to the global minimum; however, convergence tends to be very good in practice. Techniques exist for guiding the algorithm to the region of the global minimum when non-global convergence is a problem [Besl and McKay, 1992] [Grimson et al., 1995]. A promising approach to this problem based on geometric hashing is described in [Johnson and Hebert, 1996]. In the medical applications described in this dissertation, high quality initial pose estimates are derived by solving the corresponding point problem using manual specification of anatomical landmarks to determine correspondence. Therefore, the work described in this dissertation concentrates on detection, rather than avoidance, of local minima *which are far from the global minimum*.

Several researchers have demonstrated the existence of local minima very close to the global minimum [Cuchet et al., 1995] [Grimson et al., 1995]. Getting trapped in one of these local minima can lead to sub-optimal registration accuracy. An approach similar to that presented in [Grimson et al., 1995] has been implemented in this work and is outlined below.

1. The ICP algorithm is run, and the resulting pose transformation, ($\boldsymbol{R}$, $\boldsymbol{T}$), is recorded. If the least-squared error of Equation 2.2 resulting from this trial of ICP is smaller than errors from all previous trials of ICP, set the best transformation, ($\boldsymbol{R}^*$, $\boldsymbol{T}^*$), to the current transformation, ($\boldsymbol{R}$, $\boldsymbol{T}$).
2. A new initial pose is generated by randomly perturbing about the current best transformation, ($\boldsymbol{R}^*$, $\boldsymbol{T}^*$). The method for generating these random perturbations is the same as that described below in Section 3.4.2. The random perturbations are uniformly distributed, with maximum translation magnitude and rotation angle (from angle-axis representation) specified by the parameters $\theta_{p\text{-}max}$ and $\tau_{p\text{-}max}$, respectively. See Section 3.4.2 for additional details.
3. Steps 1 and 2 are iterated until I_p trials of the ICP algorithm have been performed with no additional improvement in the resulting least-squared error.

A variation on this method is to repeat the entire process several times, each time reducing the values of $\theta_{p\text{-}max}$ and $\tau_{p\text{-}max}$. The motivation is similar to the idea behind simulated annealing [Kirkpatrick et al., 1983]. Initially, large perturbations will help to avoid minima which are far from the global minima. As the process converges, progressively smaller perturbations will help to avoid local minima which are closer to the correct result.

The above procedures work well in certain situations; however, they are somewhat heuristic and are not guaranteed to avoid local minima. Local minima can often be detected at the termination of a registration trial by examining the values of the individual residuals, $\left\| M_i - (\boldsymbol{R}D_i + \boldsymbol{T}) \right\|$. If a significant number of residuals are above a threshold, there is a higher likelihood of convergence to a local minima. Histograms are useful for visualizing these error residuals. In critical applications, manual methods should be used to test for local minima in registration pose. Manual verification is possible in surgical applications using

visual checks of identified anatomical landmarks. For example, by pointing to an anatomical landmark on an actual patient with a digitizing probe, it is possible to highlight the location of this landmark on a visual display of the pre-operative image data using the estimated registration transformation. A visual check can then be performed to determine whether the selected landmark has been correctly identified in the pre-operative image. By performing this type of visual test using several landmarks, local minima in the registration pose space can be identified.

2.3.3 Outlier Elimination

When outliers due to noise or violation of the Data-Model subset constraint are present, additional processing is necessary to achieve accurate registration results. The experiments reported in this dissertation use the following threshold-based outlier detection method.

1. Run the ICP algorithm normally until convergence.
2. Calculate all corresponding point pair residuals, $\|M_i - (\boldsymbol{R}D_i + \boldsymbol{T})\|$. If any of these residuals is larger than a user specified threshold, ε_o, remove a fraction, ρ_o, of these outlier Data points.
3. Iterate steps 1 and 2 until there are no longer any residuals which are greater than the threshold.

If necessary, local minima suppression via perturbation and outlier elimination can be run together by replacing step 1 of the outlier elimination algorithm with:

1. Run the local minima suppression version of the ICP algorithm until convergence.

With this modification, outliers are only removed after the perturbation method has converged. The effect of applying local minima suppression, outlier elimination, and the hybrid method including both approaches is reported in experimental results throughout the dissertation.

The ICP algorithm has provided a basic registration capability which, together with the speed enhancements described in the next section, have been used throughout this dissertation. The ICP algorithm was chosen due to its simplicity and independence of data representation. A summary of the ICP parameters described in this section is presented in Table 2-1.

2.3.4 A Registration Example

To demonstrate the registration methods described in this section, a registration trial was performed using discrete point Data and a Model similar to those of Figure 2-2. The Data were collected from a cadaver pelvis with an optical digitizing probe. The Model was con-

Table 2-1: Complete list of registration parameters.

	Parameter	Description
Termination	Termination criterion	The ICP stopping condition which is used: i) change in relative transformation magnitude ii) change in absolute transformation magnitude iii) change in absolute least-squared error
	Termination thresholds: ε_{Rr}, ε_{Tr} ε_{Ra}, ε_{Ta} ε_{LSE}	Depending upon which termination criterion is selected, it is necessary to specify one of: 1) relative rotation ε_{Rr} and translation ε_{Tr} thresholds 2) absolute rotation ε_{Ra} and translation ε_{Ta} thresholds 3) absolute least-squared error threshold, ε_{LSE}
	k_{max}	Maximum number of ICP iterations before termination.
Minima Suppression	Minima suppression	Specifies whether perturbation-based local minima suppression is enabled.
	$\theta_{p\text{-}max}$	Maximum angle of the uniformly distributed random rotational perturbation.
	$\tau_{p\text{-}max}$	Maximum magnitude of the uniformly distributed random translational perturbation.
	I_p	Number of iterations required without additional improvement before termination of minima suppression.
Outlier Elimination	Outlier elimination	Specifies whether threshold-based outlier elimination is enabled.
	ε_o	Threshold on residual error which must be exceeded in order for a Data point to be eliminated.
	ρ_o	Ratio of Data points with residuals exceeding the threshold, ε_o, which are eliminated after a run of ICP.

structed from CT images of the same cadaver pelvis. An initial estimate of the registration transformation was derived using an anatomical landmark-based registration method. In this method, a surgeon identified three anatomical landmark points from the CT images of the pelvis. The surgeon then collected Data at the same three landmark points on the actual pelvis using an optical digitizing probe. An initial transformation estimate was then computed using the corresponding point registration method outlined in Figure 2-3 [Horn, 1987]. The resulting transformation estimate became the starting pose for the shape-based registration process.

The registration parameters used in this example are presented in Table 2-2. Local minima suppression and outlier elimination were both enabled. Sixty Data points (not including the landmark points) were collected and registered to the Model. Figure 2-5 shows the evolution of translation, rotation and registration error as a function of ICP iteration for a single trial of the ICP algorithm. Translation and rotation parameters are plotted relative to the starting pose established by anatomical landmark correspondence, and rotations are represented about a coordinate system centered in the acetabulum (the hemispherical cup of the pelvis). From the translation and rotation graphs, it can be seen that the initial pose estimate in this case was within about 40 mm and 5 degrees of the final pose. These are typical initial pose errors for registration of the pelvis using the selected anatomical landmarks. The bottom graph in Figure 2-5 shows the evolution of the root-mean-squared (RMS) error,

$$\sqrt{\sum_i \left\| M_i - (\boldsymbol{R}(k) D_i + \boldsymbol{T}(k)) \right\|^2}, \tag{2.7}$$

and the maximum residual error (MRE),

$$\max_i \left\| M_i - (\boldsymbol{R}(k) D_i + \boldsymbol{T}(k)) \right\|, \tag{2.8}$$

as a function of ICP iteration, *k*. Note that the RMS error converges monotonically towards zero. For this particular trial of the ICP algorithm, it can be seen that there are outliers in the Data since the MRE value upon convergence is roughly 15 mm, a value much larger than the expected Data or Model noise magnitude.

Table 2-2: Registration example parameters.

Parameter	Value
Termination criterion	1) change in relative transformation magnitude
Termination thresholds: ε_{Rr}, ε_{Tr}	$\varepsilon_{Rr} = 10^{-4}$, $\varepsilon_{Tr} = 10^{-4}$
k_{max}	500
Data set size, N	60
$\tau_{p\text{-}max}$	10.0 mm
$\theta_{p\text{-}max}$	7.0 deg
I_p	10 iterations
ε_o	1.5 mm
ρ_o	0.1

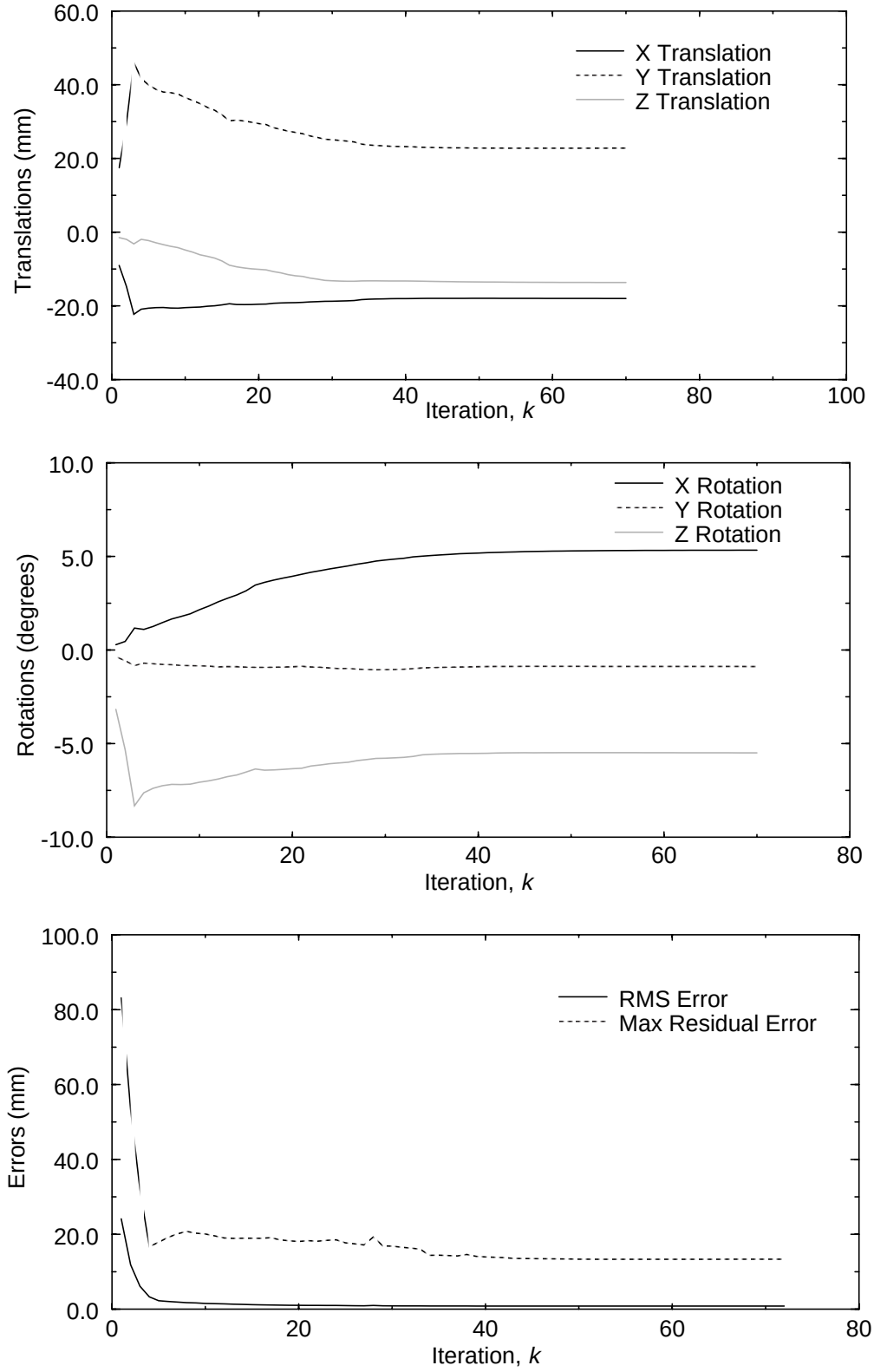


Figure 2-5: Relative translation, relative rotation, root-mean-squared error, and maximum residual error versus ICP iteration number. The transformation parameters are relative to the initial pose.

The local minima suppression and outlier elimination methods described in Sections 2.3.2 and 2.3.3 require multiple trials of the ICP algorithm. The effect of applying local minima suppression is illustrated in the top graph of Figure 2-6, which plots RMS and MRE error values at the termination of each ICP trial. The benefit of local minima suppression is demonstrated by the fact that the error values corresponding to the final ICP trial are slightly smaller than the error values corresponding to the initial ICP trial. The graph also demonstrates that the local minima suppression algorithm terminates when there is no additional improvement in the RMS error after a fixed number of ICP trials, ten in this example.

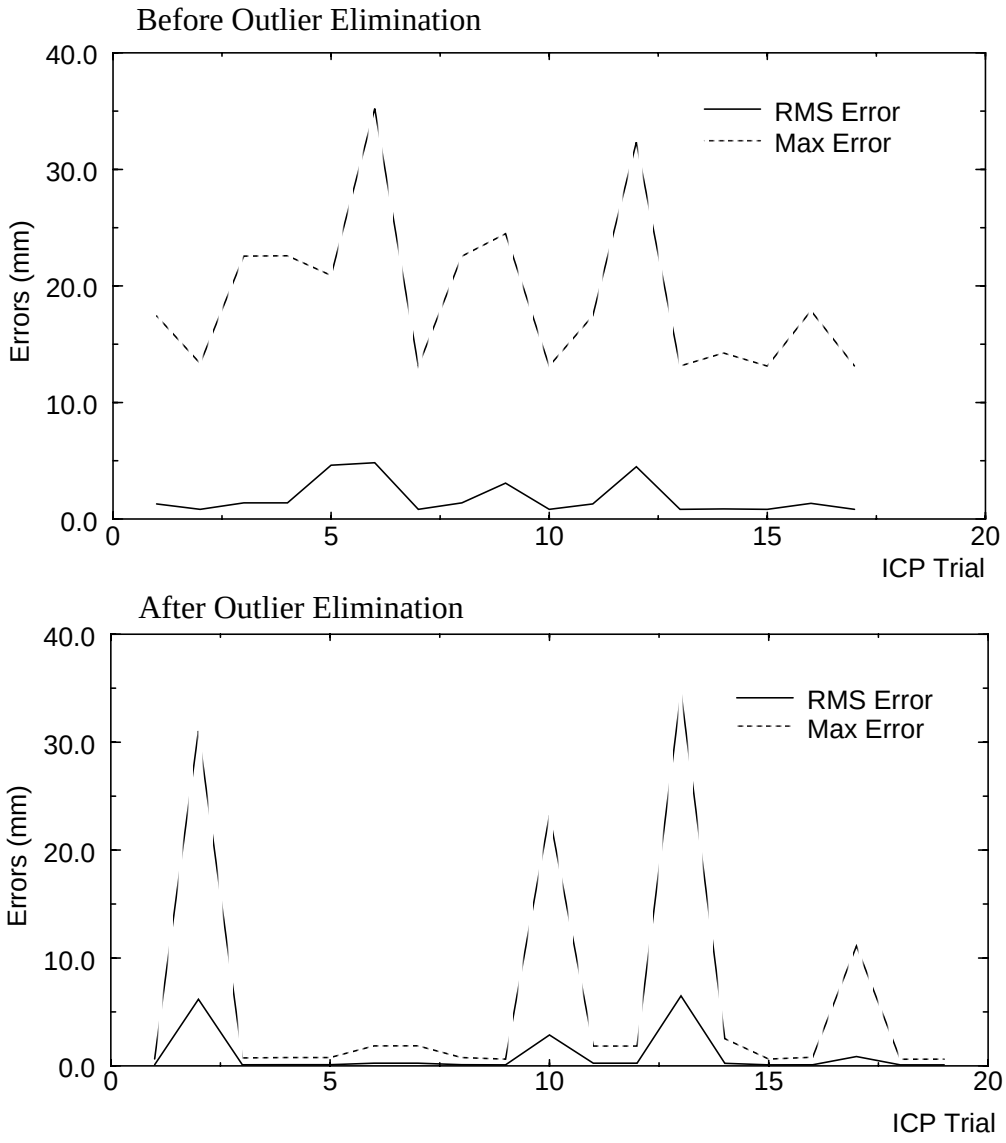


Figure 2-6: Root-mean-squared error and maximum residual error as a function of ICP trial number. The top graph corresponds to the trials before outlier elimination. The bottom graph corresponds to the trials after outlier elimination.

The bottom graph of Figure 2-6 plots RMS and MRE errors versus ICP trial after a single iteration of the outlier elimination method. In this example, five Data points have been removed between the ICP trials of the top and bottom graphs of Figure 2-6 via the method described in Section 2.3.3. The effect of removing the outlier Data is evident by comparing error values from the two graphs. RMS and MRE values for the final ICP trial are substantially smaller after outlier elimination.

The above registration example demonstrates the basic mechanisms of the ICP algorithm, the local minima suppression method, and the outlier elimination method, each of which have been used throughout this dissertation. The next section describes speed enhancements to the basic ICP algorithm which can improve ICP execution time by as much as two orders of magnitude. Section 2.5 provides a comprehensive discussion of registration accuracy, including experimental results which demonstrate the accuracy which can be achieved using the above methods.

2.4 A Speed Enhanced Registration Solution Method

Due to its simplicity, the ICP algorithm is well suited to high-speed implementation. In particular, unlike some other registration methods [Hauser and Taylor, 1990], time consuming gradient calculations are not required. For this reason, it has been possible to use the ICP algorithm as the primary component of a system for tracking the pose of arbitrarily-shaped 3-D objects at rates up to 10 Hz. To achieve these results, it was necessary to add several speed enhancements to the basic ICP algorithm. Each of these enhancements: k-d trees, closest point caching, fast surface point computation, and decoupled acceleration are described in the following sections. A description of the high speed tracking system is presented in Section 2.4.6, while details of the system are presented in Appendix A.

2.4.1 K-d trees

The most computationally expensive step in the ICP algorithm is finding the closest points (step 2 in the ICP algorithm description of Section 2.3). In general if there are N_D points in the Data and N_M geometric entities (i.e., points, lines, triangles) in the Model, then the complexity of a single closest point query is $O(N_D N_M)$. However, as suggested in [Besl and McKay, 1992] and demonstrated in [Zhang, 1994], the average complexity of a closest point query can be reduced to $O(N_D \log N_M)$ by the use of a k-dimensional binary tree, or simply k-d tree [Bentley, 1975]. The use of k-d trees for closest point computation converts the closest point computation to the search of a binary tree. At each node of the tree, a test is performed to decide which side of a hyperplane the closest point will lie on. Using this method, large regions of the search space can be pruned at each level in the search. A closest point algorithm based on the k-d tree [Friedman et al., 1977] was implemented in this work, and is the most significant factor for improving ICP execution speed as demonstrated below.

2.4.2 Closest Point Caching

A second speed improvement was realized by caching closest points. Referring to the Model as *M* and the Data as *D*, points in *M* and *D* which are proximal at iteration *k*, are highly likely to be proximal at iteration *k*+1. Thus, rather than finding the single closest point in *M* for a given Data point $D_i[k]$, the *N* closest points in *M* can be found and cached together with the point $D_i[k]$. (For reasons described below, when the cache size is *N*, in practice *N+1* closest points must be found.) There is little overhead involved in finding *N+1* closest points when *N* is a small number like 5. On the next iteration of the ICP algorithm, since the point $D_i[k+1]$ is likely to be close to the point $D_i[k]$, it is also likely that the closest point in *M* to $D_i[k+1]$ will be one of the points cached on the previous iteration. It is possible to determine conclusively whether the closest point is contained in the cached set by performing the following test. First, define $CP_n(D_i[k])$ as the *n*th closest point in *M* to D_i at iteration *k*, where *n*=0 corresponds to the closest point. Second, define $dist_{n,\,i,\,k} = \left\| D_i[k] - CP_n(D_i[k]) \right\|$, the distance between a point, D_i, and it's *n*th closest point at iteration *k*. In order to test if the closest point at iteration *k+1* is in the cache built at iteration *k*, the following condition must be satisfied:

$$\left\| D_i[k] - D_i[k+1] \right\| < \frac{(dist_{(N+1),\,i,\,k} - dist_{0,\,i,\,k})}{2} \tag{2.9}$$

This test compares the magnitude of the previous incremental transformation to the distance between the closest cached point and the *N+1*st closest point, where *N* is the total number of points in the cache. A variation on this test can determine whether the closest point at iteration *k*+1 is the *same* as the closest point at iteration *k*. This condition is satisfied if:

$$\left\| D_i[k] - D_i[k+1] \right\| < \frac{(dist_{1,\,i,\,k} - dist_{0,\,i,\,k})}{2} \tag{2.10}$$

The overall result of caching is that closest points can often be found without requiring a full search of the k-d tree. Rather, only the points in the cached set must be tested.

A similar caching technique can be applied to *spatially* (rather than *temporally*) adjacent points. If two data points $D_i[k]$ and $D_{i+1}[k]$ are spatially proximal, then it is likely that their corresponding closest points $M_i[k]$ and $M_{i+1}[k]$ will also be spatially proximal. An analogous caching technique can be applied to this situation; however, caching has not yet been implemented for spatially adjacent points. This approach is worth pursuing in the future if additional speed improvements are required in the registration process.

2.4.3 Fast *Surface* Point Computation

In the work described in this dissertation, since the Model is a triangle mesh surface, computation of the closest point requires an additional step. The output of the k-d tree based closest point algorithm is the *vertex*, V_i, which is closest to the Data point, D_i, as shown in Figure 2-7. Given V_i, the closest Model point M_i will lie within, or on the border of, one of the triangles to which the vertex belongs.[1] In order to find M_i, it is necessary to project D_i into the planes defined by each of these triangles. The resulting projected points will either lie inside or outside of a given triangle. For each triangle, if the projected point lies inside the triangle, define C_j as this point, where j is the triangle index. For projected points which lie outside of the triangle, C_j is defined as the closest point on the border of triangle j to the projected point. Finally, M_i is found as the point which is closest to D_i among all C_j. In order to perform these computations quickly, once D_i is projected into each of the planes, all computations are performed in 2-D rather than 3-D. Thus, during initialization, each triangle is stored in both its 2-D and 3-D representations.

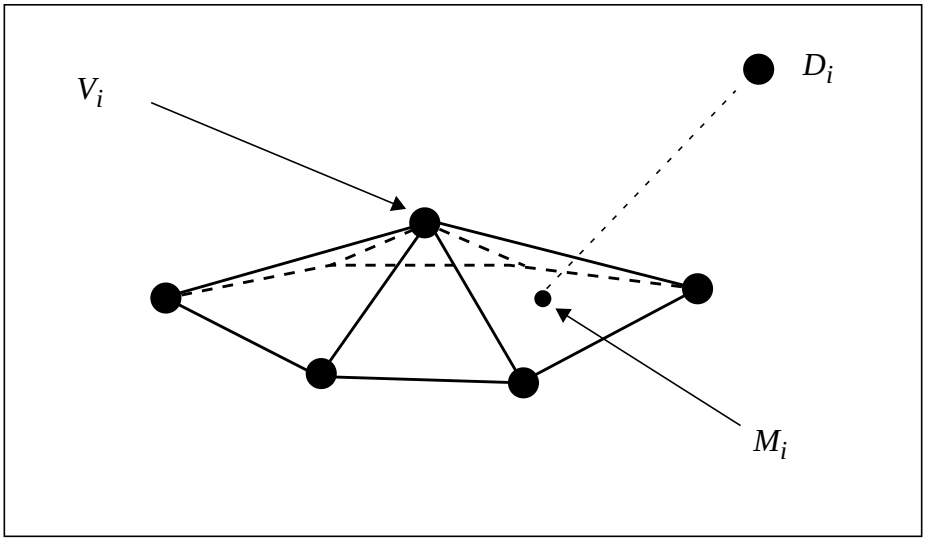


Figure 2-7: Computing the closest surface point M_i, requires projecting the test point, D_i, into the planes of each of the triangles which contain the closest vertex, V_i.

1. This is not strictly true, as there are pathological cases for which M_i will lie in a totally different triangle. For example, this may occur if part of an object is very thin, resulting in 2 or more facets which are proximal but which lie on opposite sides of the object. In practice, such pathological cases have not been a problem and have therefore been ignored.

2.4.4 Acceleration

A final speed improvement was realized using a modified version of the *accelerated* ICP algorithm described in [Besl and McKay, 1992]. The accelerated ICP algorithm adds the following step to the basic ICP algorithm after step 3:

3b. If the incremental transformations ($\boldsymbol{R_{k-1}}$, $\boldsymbol{T_{k-1}}$), ($\boldsymbol{R_{k-2}}$, $\boldsymbol{T_{k-2}}$) and ($\boldsymbol{R_{k-3}}$, $\boldsymbol{T_{k-3}}$) corresponding to iterations *k*-1, *k*-2, and *k*-3 are *well aligned*, extrapolate the current incremental transformation.

The “well aligned” condition above tests that the pose is moving in a roughly constant direction from one iteration to the next. Extrapolation is performed by scaling the current incremental transformation. The scale factor is a function of the least-squared error and the magnitude of the incremental transformations at the previous three iterations. See [Besl and McKay, 1992] for details.

Besl and McKay calculate a single acceleration scale factor for both translation and rotation. In the current work, better results were achieved by decoupling the acceleration of translation and rotation. There are two reasons for doing this. First, in Besl and McKay’s approach, the well aligned condition is tested once for both rotation and translation. Thus, for example, if rotation was well aligned but translation was not, no acceleration would be performed. However, an acceleration on rotation alone seems desirable in this situation. A second reason for decoupling is related to the scale factor used in extrapolation. Besl and McKay used the same scale factor to extrapolate both rotation and translation components. This scale factor is designed to extrapolate the solution as far as possible in a single step without overshoot. In the coupled version, the size of the scale factor is governed by the component (translation or rotation) which would cause the solution to overshoot first. The other component can usually be accelerated further. By decoupling, translation and rotation are independently accelerated as far as possible without overshoot.

2.4.5 Enhancement Results

Four speed enhancements have been described: closest point computation via k-d trees, closest point caching, fast surface point computation, and decoupled acceleration. The results of applying each of these enhancements to a single registration problem are summarized in Table 2-3. In this problem, the Data set contained 2432 points and the Model was a triangle mesh containing 4860 triangle facets. The initial pose error was a rotation of roughly 10 degrees about each axis, and a translation of roughly 10% of the object size along each axis. The number of points in the closest point cache was 5.

Table 2-3: Effect of the ICP speed enhancements upon processing time and iteration count.

Type	*Time*	*%Time*	*Iterations*	*Rot-Acc*	*Trans-Acc*
none	908.8	100.0	122	0	0
a	261.2	28.7	35	11	11
kd	62.2	6.8	122	0	0
kd/a	18.0	2.0	35	11	11
kd/d	13.1	1.4	25	13	7
kd/d/c	11.9	1.3	25	13	7
kd/d/c/2d	8.3	0.9	25	13	7

In the table, *Type* indicates the enhancements used:

- none: no speed enhancements
- a: coupled acceleration
- kd: k-d tree based closest point computation
- d: decoupled acceleration
- c: closest point caching
- 2d: fast surface point computation

Time is the total ICP execution time in seconds. *%Time* is the percentage of time relative to the slowest time. *Iterations* is the number of ICP iterations. *Rot-Acc* and *Trans-Acc* are the number of iterations for which rotation and translation were accelerated, respectively.

The speed improvements shown in Table 2-3 demonstrate the relative utility of each of the described enhancements for a given registration problem. In general, the relative utility is a function of the underlying data, the initial pose, and the ICP termination conditions. Acceleration and k-d tree search are always the two most important enhancements. The relative utility of k-d tree search increases with the number of points in the data set. Caching is useful when the termination thresholds (e.g., ε_{Rr}, ε_{Tr}) are small, since the number of cache hits will be large during the “fine-tuning” iterations of the ICP algorithm.

Additional speed improvements to the ICP algorithm would be possible via a multi-processor implementation. The closest point computation is easily parallelized by dividing the Data points across multiple processors, and the expected speed improvement is proportional to the number of processors. Depending upon future interest in high-speed applications of

this algorithm, a multi-processor implementation of ICP may be built to attain higher speeds than those demonstrated here.

2.4.6 A Demonstration of the Speed Enhanced ICP Algorithm: High-Speed Tracking

In order to demonstrate the capabilities of the speed enhanced ICP algorithm, a system was built for tracking the 3-D pose of arbitrarily-shaped rigid objects at speeds of roughly 10 Hz. Static accuracies on the order of 1 mm in translation and 1 degree in rotation have been achieved. The system has tracked human faces using Data from a high-speed VLSI range sensor developed at Carnegie Mellon University (CMU) [Gruss et al., 1992]. The implemented tracking system is independent of the sensor used or the object to be tracked. While other researchers have addressed the problem of pose tracking using range data [Grimson et al., 1994] [Yamamoto, 1993], to this author's knowledge none have demonstrated sub-second performance without the use of fiducial markers. Details of the tracking system, including accuracy results, are presented in detail in Appendix A.

High-speed 3-D pose tracking has potential uses in a variety of applications. In manufacturing, 3-D tracking could allow a mechanism (e.g., a robot) to perform a task (e.g., grasping) on arbitrarily-shaped moving parts. In human computer interaction, high-speed pose estimation could be used to track body movements for subsequent interpretation as input to a computer. In medical registration, patient tracking could help eliminate invasive and bulky fixation devices. The system described in Appendix A demonstrates the feasibility of high-speed 3-D pose tracking system with potential application to these problems.

2.5 Measures of Registration Accuracy

To quantify the accuracy which results from a registration trial, it is necessary to have appropriate measures of registration accuracy. When knowledge of the true registration transformation is available, a class of measures referred to as *ground-truth-based* accuracy measures can be used. Such measures are useful when experimentally validating a registration approach in the laboratory. Typically, ground-truth information is not available during the actual registration process. For quantifying registration accuracy in the field, a second class of measures which does not require ground-truth can be used. These are referred to as *non-ground-truth-based* accuracy measures.

A second distinction of accuracy measures is whether or not they are *task-specific* or *generic*. Task-specific measures have a direct physical relation to the task which is being performed. For example, in the computer-assisted surgical task outlined in Figure 1-3, it is possible to express registration accuracy in terms of the implant placement error induced by registration inaccuracies. Task-specific measures are often more useful for understanding the implications of registration inaccuracies than generic measures of registration error.

This section defines several accuracy measures which are used throughout the dissertation. The results of a registration experiment are presented to demonstrate the value of task-specific accuracy measures.

2.5.1 Accuracy Measure Definitions

When the true registration transformation is not known, the least-squared distance metric of Equation 2.2 forms the basis for several measures of registration accuracy. After convergence of a registration trial, it is possible to calculate the individual residual distances, $\|M_i - (\boldsymbol{R}D_i + \boldsymbol{T})\|$, between each Data point and its corresponding closest Model point. From these residuals, the following four measures can be defined:

$$\text{LSE} = \frac{1}{N} \cdot \sum_{i}^{N} \|M_i - (\boldsymbol{R}D_i + \boldsymbol{T})\|^2 \tag{2.11}$$

$$\text{RMS} = \sqrt{\frac{1}{N} \cdot \sum_{i}^{N} \|M_i - (\boldsymbol{R}D_i + \boldsymbol{T})\|^2} \tag{2.12}$$

$$\text{ARE} = \frac{1}{N} \cdot \sum_{i}^{N} \|M_i - (\boldsymbol{R}D_i + \boldsymbol{T})\| \tag{2.13}$$

and

$$\text{MRE} = \max_{i}(\|M_i - (\boldsymbol{R}D_i + \boldsymbol{T})\|) \tag{2.14}$$

where LSE is the normalized least-squared error, RMS is the normalized root-mean-squared error, and ARE and MRE are the average and maximum residual errors, respectively. These four measures have the advantage that they can be evaluated without ground-truth pose transformations, and are therefore useful as online checks of a registration result. The primary disadvantage of these measures is that they do not guarantee good registration accuracy. They are necessary, but not sufficient conditions for accurate registration. In addition, the MRE is sensitive to noise in the Data or Model. More reliable estimates of registration accuracy are possible with the use of ground-truth pose transformations; however, such transformations are typically available only during algorithm development and testing.

With reference to Figure 2-8, the transformation ${}^{M}T_{E}$ is an *estimate* resulting from the registration process of the transformation between the Model and Data coordinate systems. The transformation ${}^{M}T_{G}$ is the *ground-truth* or true transformation between the Model and Data coordinate systems. In this dissertation, ground-truth is determined either by creating the

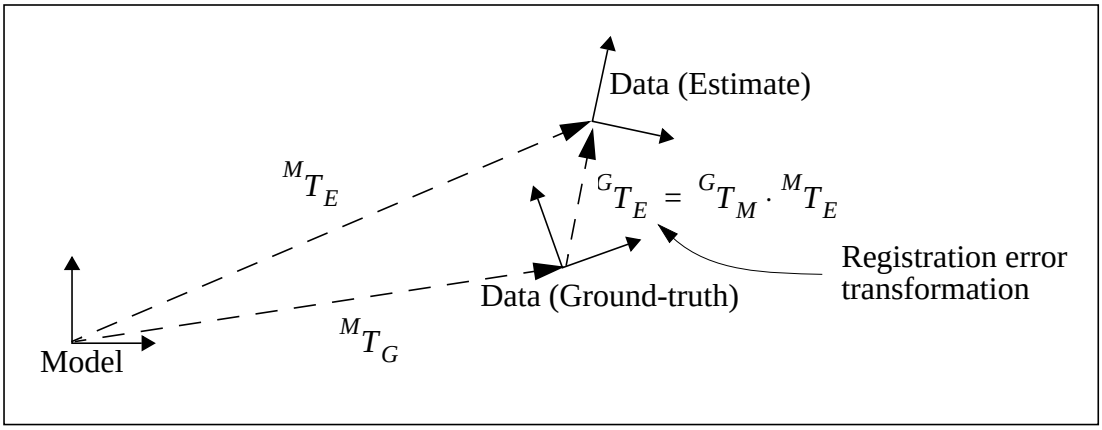


Figure 2-8: Derivation of the registration error transformation.

Data from the Model, or via a highly accurate calibration. In the former case, the Data are synthetically created by selecting discrete points on the Model and transforming these points by a known amount. In the latter case, fiducial-based registration is performed in parallel with surface-based registration to provide a highly-accurate registration transformation which is used as ground-truth. The *registration error transformation*, ${}^{G}T_{E}$, represents the difference between the estimated and ground-truth transformations. All ground-truth based accuracy measures are derived from this transformation.

The transformation ${}^{G}T_{E}$ is a function of 6 parameters, $(t_x, t_y, t_z, \omega_x, \omega_y, \omega_z)$, where $(\omega_x, \omega_y, \omega_z)$ are rotations about the *x*, *y*, and *z* axes, respectively, and (t_x, t_y, t_z) are translations along the newly rotated *x*, *y*, and *z* axes. These six parameters may be useful for quantifying registration accuracy in certain situations. Alternately, norms of the rotation and translation error components can be computed as:

$$\|E_t\| = \sqrt{t_x^2 + t_y^2 + t_z^2} \tag{2.15}$$

and:

$$\|E_\omega\| = \sqrt{\omega_x^2 + \omega_y^2 + \omega_z^2} \tag{2.16}$$

where $\|E_t\|$ and $\|E_\omega\|$ are the norms of the translation and rotation components of ${}^{G}T_{E}$, respectively. Equation 2.16 is valid as long as the individual rotation errors are small so that the approximation $\sin(\omega) \approx \omega$ holds. For large rotation errors, a more accurate measure can be derived by converting rotation error into an angle-axis representation as illustrated in Figure 2-9. This representation can describe any 3-D rotation as a single rotation about a given axis [Craig, 1986] [Spoor and Veldpaus, 1980]. The rotation angle, θ, from the angle-axis representation provides an alternative to the rotation error norm of Equation 2.16 for

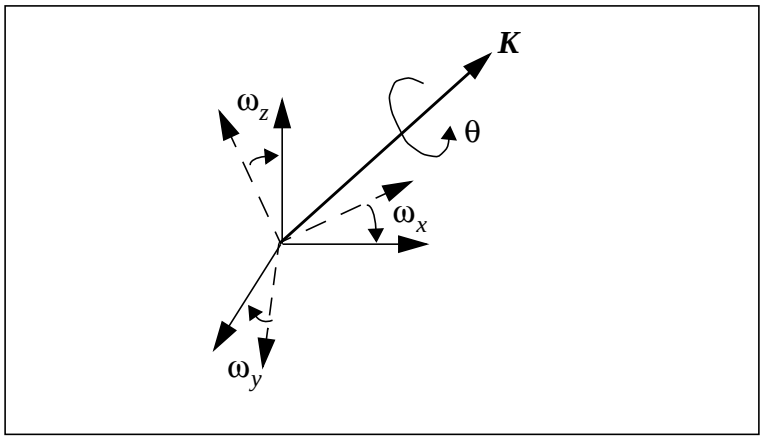


Figure 2-9: A rotation given by $(\omega_x, \omega_y, \omega_z)$ can be represented as a single rotation about an axis, $\boldsymbol{K}$, through an angle, θ.

large rotation errors; however, it can be difficult to compute for small values of θ. In practice, the angle-axis measure is used when θ is greater than 5 degrees.

Correspondence error measures (CEMs) provide an alternative to the error norm accuracy measures discussed above. A CEM measures the displacement between corresponding points induced by applying the registration error transformation to a copy of the Model, as illustrated in Figure 2-10. There are several steps required to compute a CEM. First, a dense sampling of fixed points on the surface of the Model is selected (e.g., all of the vertices of a triangle mesh surface model). Second, a copy of this set of points is created, and the copy is transformed with respect to the original set by the registration error transformation, ${}^{G}T_{E}$.

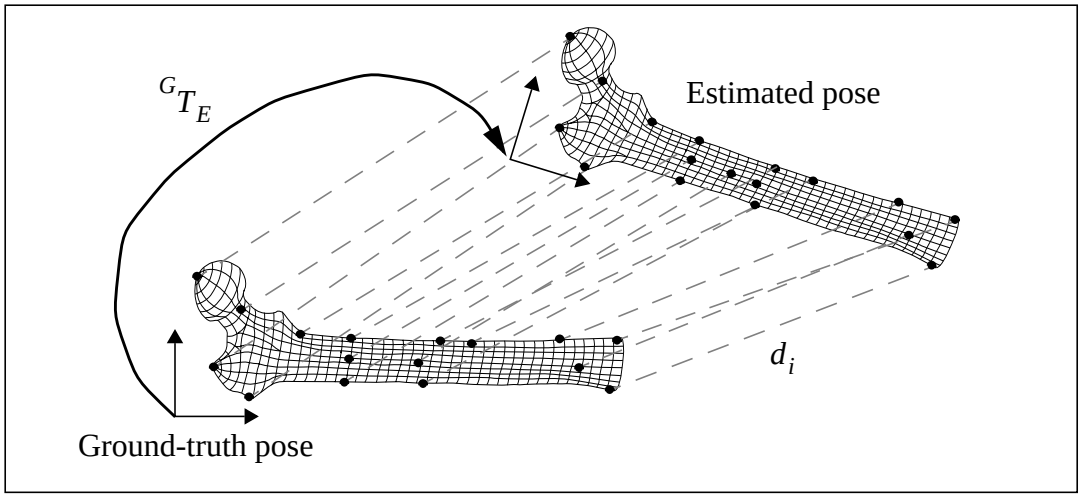


Figure 2-10: Correspondence error measures are derived by calculating the displacements between corresponding points on the Model and a copy of the Model which has been transformed by the registration error transformation.

Between each original point and its corresponding displaced copy, a Euclidean distance, d_i, is computed. Given these d_i, the following measure can be computed:

$$\text{ACE} = \frac{1}{N} \cdot \sum_{i}^{N} d_i \tag{2.17}$$

where ACE is the average correspondence error, and N is the number of selected points. Similarly, the maximum correspondence error (MCE) can be computed as:

$$\text{MCE} = \max_{i}(d_i) \tag{2.18}$$

The MCE is an intuitively satisfying measure of registration error. It provides an upper bound on the displacement induced by the registration error transformation for any point within a rigid body. Therefore, this measure has a physically meaningful relation to the task being performed. For example, consider a problem from computer-assisted surgery in which the task is to guide a drill to a precisely specified point within a bone. Assuming that registration is the only source of error in this task, the MCE would provide an upper bound on the maximum distance between the desired and actual final drill-tip position. The MCE is a useful measure for validation of a registration method when ground-truth pose information is available.

As seen in Figure 2-10, the MCE is computed using only points on the *surface* of a rigid body. It is now shown that for an arbitrary transformation, the maximally displaced point always lies on the surface, as opposed to the interior, of a rigid body.

Theorem 2-1: *The maximal point displacement resulting from the application of an arbitrary transformation to a rigid body will occur at a point on the surface of the rigid body, and not in the interior.*

PROOF. It is well known from Chasles' theorem that every spatial displacement is the composition of a rotation about some axis and a translation along the same axis [McCarthy, 1990]. This axis will be referred to as the helical axis. Applying an arbitrary transformation to a rigid body, each point within the body will be displaced by a component parallel to the helical axis (from translation along the axis), and a component orthogonal to the helical axis (from rotation about the axis). Displacement parallel to the helical axis is the same for every point within a rigid body since it results from a pure translation. Displacement of a point orthogonal to the helical axis is proportional to the distance between the point and the axis. Therefore, the point which is farthest from the helical axis will be maximally displaced by the rotation. This same point will also be maximally displaced by the entire applied transformation since rotation and translation components are orthogonal, and

translation is constant for all points. Given a rigid body and an arbitrary axis, the furthest point within the rigid body from the axis must lie on the body's surface and not in the interior. Therefore, the maximal point displacement resulting from the application of an arbitrary transformation to a rigid body must occur at a point on the surface of the rigid body and not in the interior. ❒

Correspondence error measures provide an unambiguous gauge of registration accuracy. In contrast, there is a fundamental ambiguity related to the choice of coordinate systems in the translation error norm measure of Equation 2.15. The nature of this ambiguity is illustrated in Figure 2-11. Recall that the registration error transformation, ${}^{G}T_{E}$, measures the difference between a ground-truth and an estimated pose transformation. However, there is an arbitrary decision which must be made regarding the placement of the coordinate system about which ${}^{G}T_{E}$ is represented. In Figure 2-8 for example, the registration error is represented with respect to the ground-truth coordinate system, *G*. The effect of this choice is demonstrated in Figure 2-11 by the difference in translation error between the cases on the left and right. The apparent distortion of error is an artifact of the choice of coordinate system. The actual displacement between both sets of rigid bodies in Figure 2-11 is identical. Mathematically, the relation between the two transformations can be expressed as:

$$ {}^{G}T_{E}{}^{*} \;=\; {}^{G^{*}}T_{G}\,{}^{G}T_{E}\,{}^{G}T_{G^{*}} \tag{2.19} $$

where ${}^{G}T_{G^{*}}$ is the transformation between the coordinate system on the left, and the one on the right. Due to the effect illustrated in Figure 2-11, careful selection of a coordinate system is required when using measures such as $\|E_{t}\|$ and $\|E_{\omega}\|$.

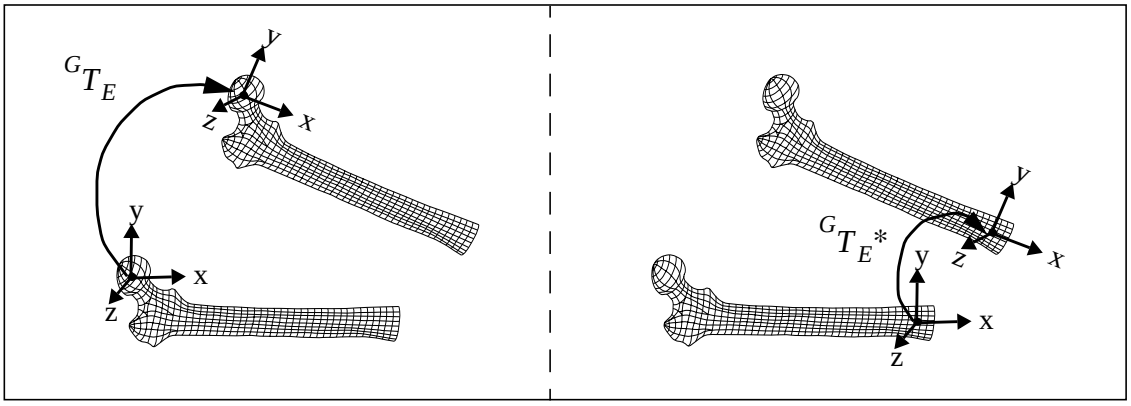


Figure 2-11: Registration error transformations as a function of coordinate system placement. Left: the coordinate system is centered on the femoral head. Right: the coordinate system is centered at a point on the femoral shaft. Displacement between the two rigid bodies is identical; however, the transformation ${}^{G}T_{E}$ varies as a function of coordinate system placement.

Another example of the above ambiguity is given in the simple 2-D example of Figure 2-12. On the left, the two lines labeled 1 and 2 represent a rotational misalignment error such as that resulting from registration. With respect to a coordinate system positioned at point **a**, this error can be described as a pure rotation. In order to describe the same error with respect to a coordinate system positioned at point **b**, line 2 must first be rotated about point **b** so that it is parallel to line 1, and then shifted back to point **a**. The angles of rotation about points **a** or **b** are the same; however, the translations differ.

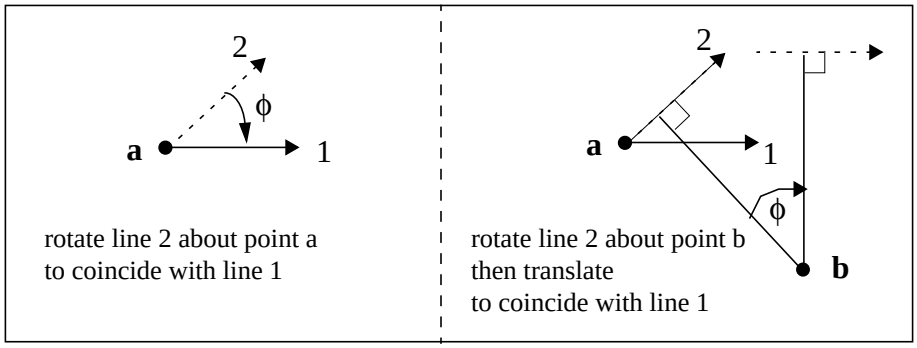


Figure 2-12: Two ways to represent the same spatial transformation. Left: the transformation between lines 1 and 2 can be represented as a pure rotation about point **a**. Right: the same transformation can be represented as a rotation about point **b**, followed by a translation back to point **a**.

2.5.2 Accuracy Measure Demonstration

In this subsection, some of the accuracy measures defined in the previous section are demonstrated using an example task from computer-assisted surgery. The task is to accurately "place" an orthopaedic implant within a human femur based upon geometric information from a pre-operative plan. For the experiment performed in this section, the following steps which parallel those in Figure 1-3 were performed: a pre-operative plan was constructed from CT images of the femur; registration was performed using a Model constructed from the CT images and Data collected from the bone using a digitizing probe; and the accuracy of registration was evaluated using fiducial-based registration to provide the ground-truth. The actual placement of the physical implant was never performed. The goal of the experiment was to evaluate the effect which the physically measured registration error would have on implant misalignment.

To perform the experiment, a human cadaveric femur was CT scanned in water at 1 mm intervals. A schematic of the pre-operative plan constructed from these images is shown in Figure 2-13. Each of the numbered points correspond to origins of coordinate systems which are used to represent registration error in the results below. The pre-operative plan

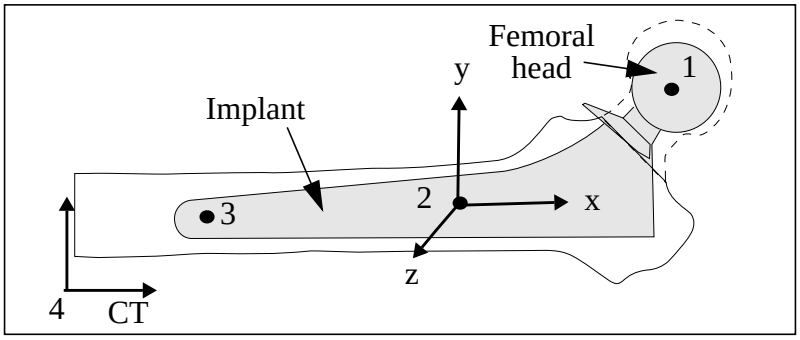


Figure 2-13: Schematic of a pre-operative plan illustrating the desired placement of an implant within a femur for total hip replacement surgery. The numbered points indicate origins of coordinate systems used to represent registration errors.

was constructed manually via a graphical user interface which allowed the user to overlay a model of the implant upon orthogonal views of the CT images.

After construction of the plan, a surface model was built from the CT images for use as the registration Model. One hundred Data measurements were collected within a region of the proximal femur using a digitizing probe which was accurate to 0.1 mm. Data collection was limited to areas of the bone which are clinically accessible during total hip replacement surgery. The EICP algorithm was used to estimate the registration transformation, and corresponding point registration was performed using 6 spherical fiducial markers to calculate the ground-truth transformation. Descriptions of these fiducial markers and additional details of the experiment can be found in Section 5.3 and [Simon et al., 1995].

Table 2-4 summarizes the results of the experiment using the error norms defined in Equations 2.15 and 2.16. Within each row of the table the error norms are calculated with respect to a coordinate system defined in Figure 2-13. The coordinate systems numbered 1-3 are parallel, with the x-axis in the direction of the implant shaft's central axis, the y-axis defined by the projection of the femoral head centroid onto the x-axis, and the z-axis defined as the cross product of the first two. The origin of each coordinate system was selected for its relevance to the implant placement task: 1) centroid of the implant femoral head; 2) centroid of the implant; and 3) distal tip of the implant. The fourth coordinate system is the one used by the CT scanner.

In the table, the magnitudes of the translation errors are a function of coordinate system, while the magnitudes of the rotation errors are independent of coordinate system. This is consistent with the explanation of Figure 2-12. The key point is that each of the results in Table 2-4 refers to the *same physical displacement*. These results demonstrate the potential ambiguity of accuracy measures which require selection of a coordinate system. When using such measures, it is crucial that the coordinate system have a physically meaningful

Table 2-4: Registration error norms for implant placement task as a function of coordinate system location.

Location of coordinate system origin	$\|E_t\|$ (mm)	$\|E_\omega\|$ (deg)
1 - Head Centroid	0.25	0.97
2 - Implant Centroid	0.31	0.97
3 - Implant Distal Tip	0.17	0.97
4 - CT	1.77	0.97

relation to the task being performed. For this example, depending upon the selected coordinate system, errors could either be reported as 0.17 mm in translation or 1.77 mm in translation, a factor of 10 difference. Fortunately, meaningful error values for this task are closer to the smaller value.

Table 2-5 presents the results of the same experiment using correspondence error measures rather than error norms. Using the approach described on page 36, the correspondence errors, d_i, are computed for each of the points shown in the table. The first three points are the points labeled 1-3 in Figure 2-13. The fourth point is the extremum point for which the resulting d_i value is equal to the maximum correspondence error (MCE). These results are independent of the choice of coordinate system. Each of the reported d_i values can be interpreted as the implant misalignment which would result at the specified point if the implant were actually placed using the estimated registration transformation. It is no coincidence that the d_i values are the same as the corresponding translation norm magnitudes from Table 2-4. This can be explained with respect to Figure 2-12. On the right side of the figure, the translation required to realign line 3 with line 1 is the same as the displacement which would be induced at point **b** by rotating line 2 together with point **b**, about point **a** to match line 1.

Table 2-5: Registration correspondence errors for implant placement task as a function of corresponding point location

Location of corresponding point (i)	d_i (mm)
1 - Head centroid	0.25
2 - Implant centroid	0.31
3 - Implant distal tip	0.17
4 - Extremum point	0.54

The fourth point in Table 2-5 is the extremum point, the point for which d_i is maximum over all points within the implant. As noted earlier, the MCE provides an upper bound on the implant misalignment introduced by registration. As suggested by Theorem 2-1, the extremum point can be calculated as the point on the implant's surface which is farthest from the helical axis representation of the registration error transformation. Figure 2-14 illustrates the relation of the points of Table 2-5 to a projection of the helical axis error transformation. From the figure, it can be seen that the extremum point is farthest from the helical axis, and each of the d_i values from the table are proportional to the distance between the helical axis and the associated point. For this reason, 3-D graphical renderings of the helical axis superimposed on the relevant rigid body are useful for visually interpreting errors resulting from registration.

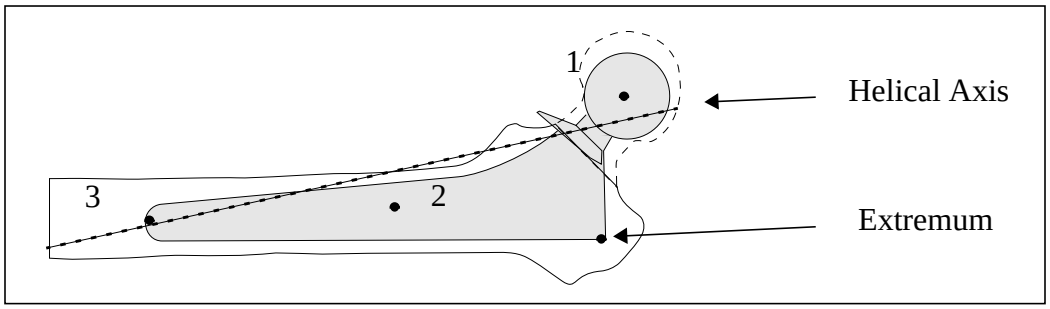


Figure 2-14: Relation of the helical axis error representation to the points in Table 2-5. The error transformation is represented as a rotation about the helical axis and a translation along it. The d_i values of Table 2-5 are proportional to the distance between the corresponding point and the helical axis.

2.6 Discussion

The registration methods described in this chapter have provided reliable tools for use in both research and application. The primary innovation in the described implementation is the ability to perform registration at high-speed, as demonstrated by the real-time pose tracking system. The methods described for handling outliers and avoiding local minima have been very effective in the experiments described throughout this dissertation. An implementation of the registration method described in this chapter is currently being prepared for use in a clinical trial in which computer-assisted surgical techniques are being applied to a problem in total hip replacement surgery [DiGioia et al., 1995].

The registration literature contains many reports of registration accuracy in which the particular metrics used are not carefully described. It is not uncommon for registration accuracy to be reported using descriptions such as: "Our method is shown to be accurate to 1 mm". As demonstrated in this chapter, there is potential ambiguity in such descriptions. When using accuracy measures which depend upon a choice of coordinate system, a careful description

of the selected coordinate system is required. When registration is used to perform a particular task, accuracy should be reported using measures which have a direct physical relation to the task. The maximum correspondence error (MCE) was shown to be a generic measure of registration accuracy which represents the largest point displacement within an object resulting from registration inaccuracies.

This chapter has emphasized the details of performing and evaluating shape-based registration without regard to the underlying Data. The next chapter presents a method for assessing the quality of registration Data for the purpose of achieving accurate registration results. It is shown that careful selection of registration Data can have a profound impact upon the resulting registration accuracy.

Chapter 3

Constraint Analysis

The goal of the Intelligent Data Selection (IDS) method outlined in this dissertation is to guide the collection of a fixed number of discrete point Data measurements in a manner which maximizes registration accuracy for a given arbitrarily-shaped object. As discussed later in this dissertation, by combining IDS with online accuracy estimation, it is possible to minimize the number of Data measurements required to satisfy a given accuracy requirement. As outlined in Figure 3-1, the input to the IDS process is a surface Model of the registration object, and the output is a plan for acquiring discrete-point measurement Data. This chapter presents the theoretical framework and numerical formulation of the first component of IDS, *constraint analysis.* As input, constraint analysis requires a surface Model of the registration object and a hypothesized set of discrete point Data measurements. The output of constraint analysis is a sensitivity measure which is shown to be a good indicator of expected registration accuracy. Using this sensitivity measure, the *constraint synthesis* algorithm described in Chapter 4 can generate sets of discrete-point Data measurements which are shown to be near-optimal in terms of the resulting registration accuracy. The remainder of this chapter is organized as follows. In Section 3.1, several motivating examples for the constraint analysis problem are provided. Section 3.2 includes a review of the relevant literature. Section 3.3 presents a derivation of constraint analysis and discussions of the effect of object scale and coordinate system placement upon the analysis. Section 3.4 contains experimental results which demonstrate that constraint analysis is a good predictor of registration accuracy. Section 3.5 concludes the chapter with a discussion of constraint analysis and its relation to the other components of this dissertation.

3.1 Problem Description

The goal of registration as formulated in the previous chapter is to find a rotation, $\boldsymbol{R}$, and translation, $\boldsymbol{T}$, which minimize the equation:

$$\sum_i \left\| M_i - (\boldsymbol{R} D_i + \boldsymbol{T}) \right\|^2 \tag{3.1}$$

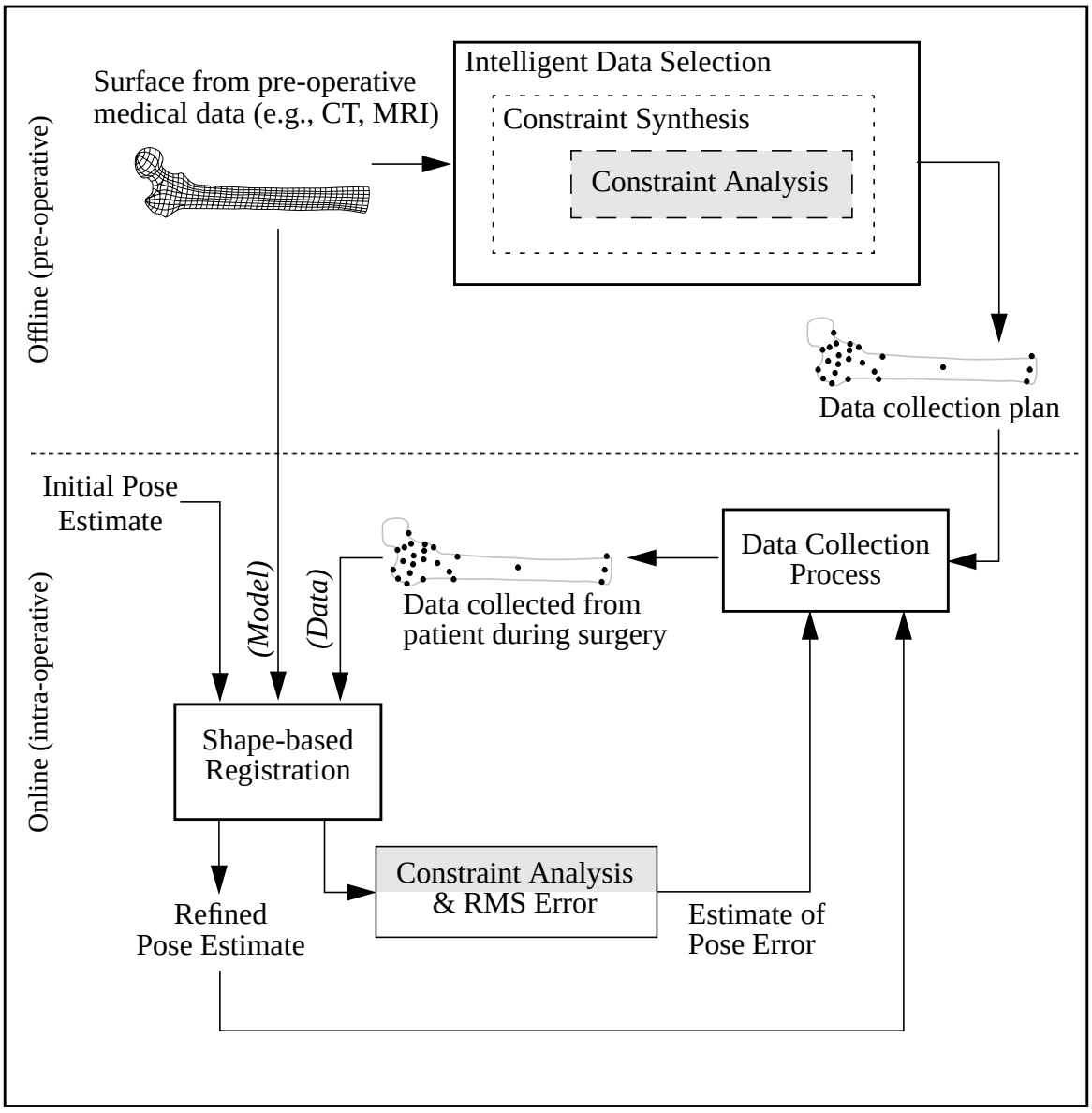


Figure 3-1: The role of constraint analysis (shaded) in Intelligent Data Selection and online error estimation.

where D_i are discrete point Data measurements, and M_i are the corresponding closest points on a surface Model. One aspect of registration which has received little attention in the literature is the selection of the measurements, D_i, which are used to minimize Equation 3.1. As demonstrated in the following examples, the particular choice of measurements can have a profound effect upon registration accuracy.

Consider the problem illustrated in Figure 3-2 in which the goal is to estimate the pose of a slotted cylinder by acquiring discrete point Data measurements on its surface. If measure-

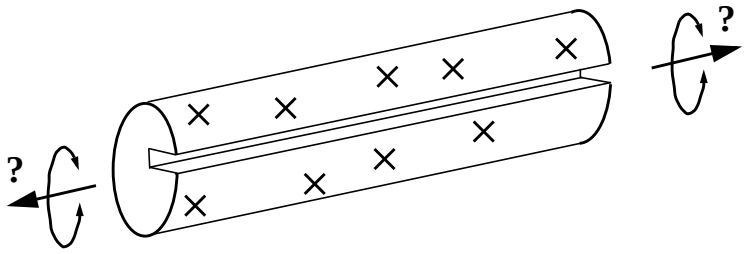


Figure 3-2: Ambiguities in estimating the pose of a slotted cylinder using Data measurements at locations indicated by the marks.

ments were only collected at locations indicated by the marks, it would be impossible to determine orientation about the central axis or translation along it. Additional Data collected in the disk-shaped end regions and within the slot would allow determination of translation and rotation, respectively. The problem with the Data measurement configuration shown in Figure 3-2 is that it does not completely constrain the pose of the cylinder. In other words, "freedoms" or singularities exist between the Model and Data causing multiple solutions in $\boldsymbol{R}$ and $\boldsymbol{T}$ to Equation 3.1. While these freedoms are visually apparent in this example, when dealing with more complex surfaces such as the Pelvis Model of Figure 2-2, identification of this condition becomes more difficult.

Another example which illustrates the effect of Data selection upon registration accuracy is described with reference to Figure 3-3. In this example, the problem is to estimate the pose of a cube (disregarding symmetries) using the three Data configurations shown in the figure. The configurations, C1, C2 and C3, contain 25, 4, and 4 points per face, respectively, for totals of 150, 24 and 24 points per cube.

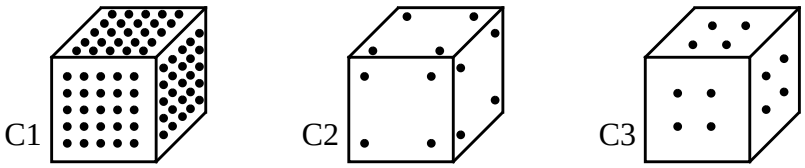


Figure 3-3: Three registration Data configurations on a cube: C1, C2 and C3.

To demonstrate the effect of Data configuration upon accuracy, registration was performed between a noiseless surface Model of the cube, and corrupted versions of the three Data configurations. The corrupted Data sets were generated by adding zero mean Gaussian noise to the nominal point locations of Figure 3-3 (see Section 3.4.2 for a detailed description of the experimental method). Registration was performed 100 times for each Data configuration from known, random initial poses. Figure 3-4 shows the resulting normalized least-squared errors of Equation 2.11, plotted relative to the error for configuration C1. As might

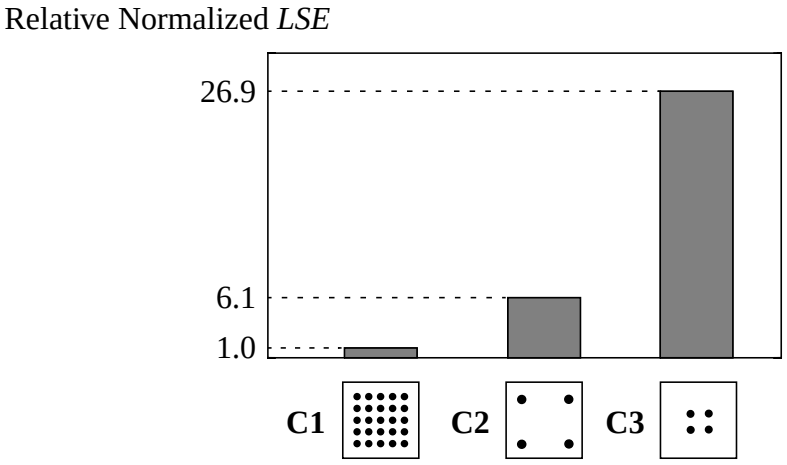


Figure 3-4: Least-squared error (*LSE*) resulting from registration of the cube with the Data configurations of Figure 3-3 normalized relative to the *LSE* of configuration C1.

be expected, configuration C1 results in the best registration accuracy. This agrees with the intuition that larger quantities of registration Data will result in better registration accuracy. For the two 24 point configurations, C2 clearly provides better accuracy than C3. An explanation of this result is given in the next section; however, note that in the absence of measurement noise, registration error would be zero for all three configurations. The difference between configurations C2 and C3 is related to the sensitivity of registration accuracy to measurement noise.

The goal of this chapter is to develop a theory of constraint analysis which provides a mathematical basis for understanding the examples presented above. Constraint analysis provides a very fast numerical method for quantitatively evaluating discrete point Data configurations in terms of expected relative registration accuracy. The importance of fast numerical evaluation is discussed in Chapter 4, in which constraint analysis is used as an evaluation metric for constraint synthesis.

3.2 Background

This section contains a review of the literature in areas related to constraint analysis. There are a number of engineering problems described in the literature which share an underlying mathematical structure with constraint analysis. In particular, there are similarities to work in analysis and design of industrial fixtures and robotic grasps, robotic manipulability and other measures of robot dexterity, and robot kinematic calibration. The most closely related work to constraint analysis was reported by a group at the Ohio State University on the problem of industrial part localization for surface measurement and inspection. This section

presents a brief overview of the literature in each of the above areas in order of increasing relevance.

3.2.1 Analysis and Design of Industrial Fixtures and Robotic Grasps

Design of industrial fixtures and robotic grasps has the goal of physically immobilizing an arbitrarily-shaped object using a limited number of contact points. In both of these problems, choice of contact point location can greatly affect the stability of the resulting hold. As demonstrated in Section 3.1, the choice of discrete point measurement locations in registration can greatly affect the resulting accuracy. At a fundamental level, the underlying mathematics of fixture and grasp analysis, especially in the frictionless case, is similar to the mathematics of constraint analysis presented below in Section 3.3. An overview of problems in robot grasping and dexterous manipulation can be found in [Mishra and Silver, 1989]. From Section IV of this paper in particular, the relation between frictionless grasp analysis and constraint analysis is apparent.

In [Asada and By, 1985], a method is presented for analyzing the local uniqueness of a pose between an object and a set of fixture elements. Given a CAD model of an object and a set of fixture contact positions, the authors determine whether motion of the object with respect to the fixture is possible within a small vicinity of pose space. A similar problem is examined in the context of robotic grasping in [Asada and Kitagawa, 1989]. A *form closure* grasp is one which constrains a rigid object using mechanical fingers such that object motion is geometrically constrained in all directions. This paper presents a linear programming technique for determining whether form closure is satisfied in a given situation. The analysis considers not only smooth regions of an object, but also the case of contact at surface discontinuities (e.g., convex or concave edges). The goals of both of the above papers are similar to the goal of constraint analysis. Fixture and grasp analysis, however, differ from constraint analysis in one important regard; it is impossible for a fixture or a finger to penetrate the surface of an object. Any computational algorithms for designing fixtures or grasps must ensure that this constraint is maintained. During registration, however, a point in the Data may be positioned on either side of the surface of the Model. Since constraint analysis does not incorporate the non-penetration constraint used in grasp or fixture analysis, the work presented in this dissertation does not map directly to these problems.

The work by Ohwovoriole and Roth [Ohwovoriole and Roth, 1981] in the area of screw theory is used extensively in fixturing and grasping research. Their paper outlines a theoretical framework for the analysis of differential object motion under frictionless point constraint. While the formulation of constraint analysis presented below does not make explicit use of screw theory, apparent similarities are noted.

3.2.2 Robot Dexterity Measures

Robot dexterity measures provide a quantitative evaluation of the ease with which the position of a robot can be changed (or equivalently the ease with which it can apply a force) when the robot is in a specified configuration. These measures are based upon a robot's Jacobian matrix. Dexterity measures are used in the manipulator design process to evaluate the implications of kinematic design decisions (e.g, manipulator size, link lengths) on the robot's ability to perform a given task. Dexterity measures are also used to select appropriate configurations when programming a robot to perform a task.

The manipulator Jacobian relates the rate of change of a manipulator's endpoint location to the rate of change of each of the joint angles. Robot dexterity measures are scalar sensitivity measures derived from the Jacobian which express a manipulator's ability to move its endpoint in an arbitrary direction from a given configuration. As demonstrated below, in constraint analysis there is a matrix which relates the rate of change of a first order approximation of the registration cost metric, Equation 3.1, to the rate of change of object pose. As in the case of robot dexterity, it is possible to derive scalar sensitivity measures from this matrix which express the ability to localize an object in all directions from a given set of measurements.

Perhaps the best known measure of robot dexterity is the manipulability measure described in [Yoshikawa, 1990]. Manipulability is basically the determinant of the manipulator Jacobian. It can be shown that manipulability is also a function of the eigenvalues of the Jacobian matrix multiplied by its transpose. Other robot dexterity measures can be derived from these eigenvalues as well. Excellent overviews of robot dexterity measures are presented in [Kim and Khosla, 1991] and [Nahvi and Hollerbach, 1996]. The measures used for constraint analysis in this dissertation are identical to those used as measures of robot dexterity by previous researchers. Additional discussion of these measures is presented in Section 3.3.2.

3.2.3 Pose Selection for Kinematic Calibration of Robots

Pose selection for kinematic calibration of robots is a very similar problem to the problem described in the previous section. The goal of kinematic calibration is to identify manipulator parameters (e.g., link lengths, relative joint orientations, etc.) based on measurements of the robot's end-effector pose at several commanded configurations [Hollerbach, 1993]. The goal of pose selection is to choose a small number of poses which maximize kinematic parameter identification accuracy, assuming that pose measurements are corrupted by noise. Once again, there is a fundamental similarity to the goal of constraint analysis.

Since the Jacobian matrix is fundamental to the kinematic calibration pose selection problem and the robot dexterity measure problem, the same criterion measures are used in these

problems. An overview of work in optimal pose selection for kinematic calibration is presented in [Nahvi and Hollerbach, 1996]. That paper also introduces a criterion measure called the “noise amplification index” which is used later in this dissertation. In [Borm and Menq, 1991] and [Zhuang et al., 1994], methods for selecting optimal robot poses for kinematic calibration are presented. The optimization methods presented in those papers are similar in spirit to the constraint synthesis work presented in Chapter 4.

3.2.4 Sensitivity Analysis for Registration

Among the literature reviewed for this dissertation, work done by a group at The Ohio State University [Sahoo and Menq, 1991] [Menq et al., 1992] [Sahoo and Menq, 1988] is most similar to the constraint analysis method described in this chapter. These papers discuss a method for evaluating the sensitivity of registration accuracy as a function of the underlying data in the context of industrial part localization for surface measurement and inspection. Their formulation is quite similar to the independently derived constraint analysis method presented in subsequent sections of this chapter. The work presented in this dissertation improves upon the earlier work in several ways. First, this dissertation presents a solution to an important problem which was not addressed in the earlier work: automatic selection of optimal measurement configurations based upon the sensitivity analysis. Additional improvements presented in this dissertation include: recognition and treatment of the scale dependence problem which affects how rotations and translations are weighted in the analysis; recognition and treatment of the coordinate system dependence problem; choice of a different criterion measure; improved geometric interpretation of the analysis; and substantially more experimental validation.

The application of the Ohio State work was the localization and inspection of industrial parts using CAD models. While this domain differs from the medical domain in terms of registration requirements, underlying data, etc., the fundamental problems are the same. Therefore, the results presented in this dissertation can be directly applied to problems in industrial parts localization.

3.3 Constraint Analysis Formulation

This section presents the theoretical and computational framework of constraint analysis. Section 3.3.1 describes the mathematical derivation with additional details presented in Appendix B. Section 3.3.2 compares several criterion measures and justifies the selection of a particular measure which is used throughout the remainder of the dissertation. Section 3.3.3 discusses a fundamental problem related to scale dependence in constraint analysis. It is shown that the scale of surfaces used in constraint analysis affects the relative weight which rotational versus translational transformations have upon the analysis. Section 3.3.4 discusses a dependence between constraint analysis and the position of the

coordinate system in which the analysis is performed. A method for dealing with this dependence is proposed.

3.3.1 Derivation

The derivation of constraint analysis begins by posing the following question as illustrated in Figure 3-5. Given a discrete point which lies on a surface, how does the distance between the point and the surface vary as the point is perturbed by a small amount about its resting position? This question can be answered by computing the gradient of this distance with respect to an arbitrary transformation of the surface point.

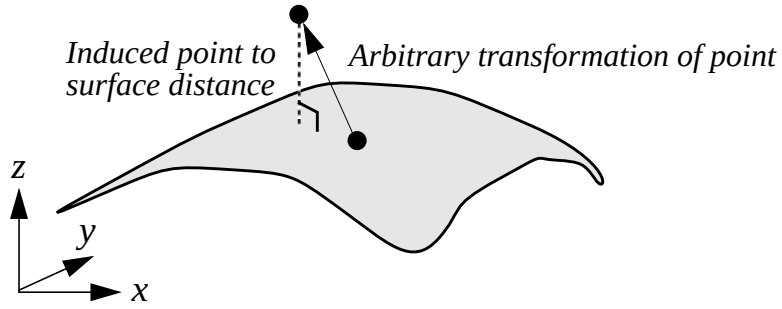


Figure 3-5: Distance induced by a small perturbation of a point lying on a surface.

The distance between a point $\boldsymbol{x}$, and a surface is defined as the length of the shortest line between the point and the surface. In general, there is no closed form analytical expression for this distance given an arbitrary surface; however, several researchers have proposed the following local approximation [Sampson, 1982] [Taubin, 1991]:

$$D(\boldsymbol{x}) = \frac{F(\boldsymbol{x})}{\|\nabla F(\boldsymbol{x})\|} \tag{3.2}$$

where $F(\boldsymbol{x}) = 0$ is the implicit equation of the surface, $\|\nabla F(\boldsymbol{x})\|$ is the magnitude of the gradient to the surface, $\boldsymbol{x}$ is a point which may or may not lie on the surface, and $D(\boldsymbol{x})$ is the approximate distance. It can be shown that $D(\boldsymbol{x})$ is a first order approximation of the true point-to-surface distance, and is the exact distance when the surface is a plane.

Assume that there exists a point $\boldsymbol{x}_s$, which lies on the surface such that $D(\boldsymbol{x}_s) = 0$. This point can be perturbed with respect to the surface by applying a differential transformation, *T*, to the point. *T* can be represented by a homogeneous transformation which is a function of 6 parameters $(t_x, t_y, t_z, \omega_x, \omega_y, \omega_z)$, where $(\omega_x, \omega_y, \omega_z)$ are rotations about the *x*, *y*, and *z*

axes, respectively, and (t_x, t_y, t_z) are translations along the newly rotated *x*, *y*, and *z* axes. Define:

$$\boldsymbol{t} = \left[t_x \; t_y \; t_z \; \omega_x \; \omega_y \; \omega_z \right]^{\mathbf{T}} \tag{3.3}$$

as the 6-vector of transformation parameters. The gradient of the point-to-surface distance, *D*, with respect to $\boldsymbol{t}$ specifies how *D* varies for an arbitrary transformation of the point. As shown in Appendix B, this gradient can be computed and the resulting 6-vector, $\boldsymbol{V}(\boldsymbol{x}_s)$, is defined as:

$$\boldsymbol{V}(\boldsymbol{x}_s) = \frac{\partial}{\partial \boldsymbol{t}} D(T(\boldsymbol{x}_s)) = \begin{bmatrix} \boldsymbol{n} \\ \boldsymbol{x}_s \times \boldsymbol{n} \end{bmatrix} \tag{3.4}$$

where $\boldsymbol{n}$ is the unit normal to the surface evaluated at the point $\boldsymbol{x}_s$, and $\boldsymbol{x}_s \times \boldsymbol{n}$ is the cross-product of the radius vector with the surface normal. $\boldsymbol{V}(\boldsymbol{x}_s)$ relates a small transformation specified by a vector $d\boldsymbol{t}$ to a corresponding change in distance between the point and the surface. In other words:

$$D(T(\boldsymbol{x}_s)) = \boldsymbol{V}^{\mathrm{T}}(\boldsymbol{x}_s) d\boldsymbol{t} \tag{3.5}$$

Drawing a parallel to the robot dexterity measures discussed in Section 3.2, the sensitivity vector $\boldsymbol{V}(\boldsymbol{x}_s)$ of Equation 3.4 is similar to the manipulator Jacobian. In the robot grasping and fixture analysis literature, $\boldsymbol{V}(\boldsymbol{x}_s)$ is a wrench system which represents a frictionless force applied to a body in the direction $\boldsymbol{n}$ at the point $\boldsymbol{x}_s$ [Mishra and Silver, 1989]. This expression is also used by the Ohio State University group in their sensitivity analysis formulation [Menq et al., 1992].

There exists an intuitive geometric interpretation of Equation 3.5. Grouping the first three elements of $d\boldsymbol{t}$ into a vector describing the translation components, and the last three elements into a vector describing the rotation components:

$$D_\tau(\boldsymbol{x}_s) = \boldsymbol{n}^{\mathrm{T}} d\tau \tag{3.6}$$

$$D_\theta(\boldsymbol{x}_s) = (\boldsymbol{x}_s \times \boldsymbol{n})^{\mathrm{T}} d\theta = (d\theta \times \boldsymbol{x}_s)^{\mathrm{T}} \boldsymbol{n} \tag{3.7}$$

where $d\tau = \left[dt_x \; dt_y \; dt_z \right]^{\mathrm{T}}$ and $d\theta = \left[d\omega_x \; d\omega_y \; d\omega_z \right]^{\mathrm{T}}$, while D_τ and D_θ are the components of the distance induced by translation and rotation, respectively.

Geometric interpretations of Equations 3.6 and 3.7 are presented in Figures 3-6 and 3-7, respectively. Figure 3-6 indicates that a small translation of the point in a direction parallel

to the surface normal at that point will result in the largest change in distance. Similarly, a small translation perpendicular to the surface normal will result in no change in distance since the point will be moving tangentially to the surface. The rotational case in Figure 3-7 can be interpreted by recognizing that a differential rotation through the origin specified by the vector $d\theta$, will induce a translation of the point $\boldsymbol{x}_s$ given by $d\theta \times \boldsymbol{x}_s$. The resulting translation can then be interpreted as in Figure 3-6. An experimental result which provides intuition into the meaning of the sensitivity vector, $\boldsymbol{V}(\boldsymbol{x}_s)$, is presented in Section 3.4.1.

Until now, the derivation has considered how the distance between a single point and a surface changes as a function of an arbitrary, small rigid transformation. The goal is to perform this analysis for a *collection* of points which lie on the surface. Recall from Equation 3.5, that for a single point:

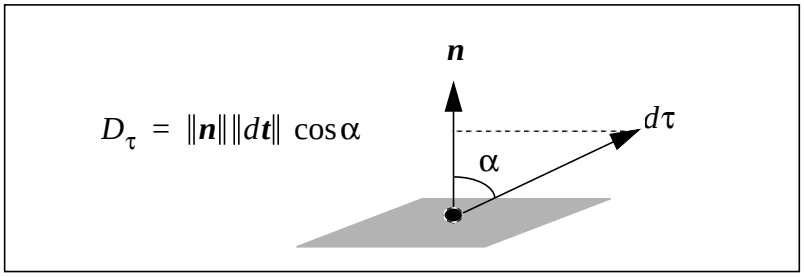


Figure 3-6: Geometric interpretation of the translational components of the sensitivity vector, $\boldsymbol{V}(\boldsymbol{x}_s)$.

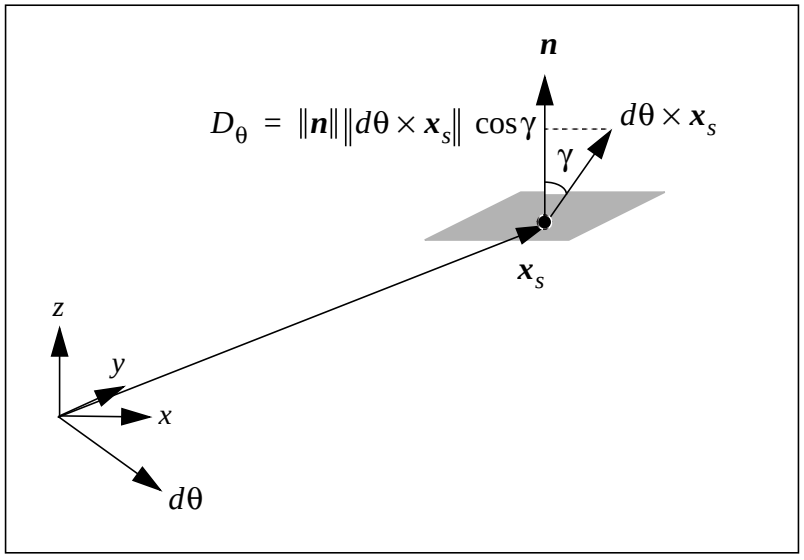


Figure 3-7: Geometric interpretation of the rotational components of the sensitivity vector, $\boldsymbol{V}(\boldsymbol{x}_s)$. A rotation about the vector $d\theta$ will induce a translation at the point $\boldsymbol{x}_s$ in the direction of the vector $(d\theta \times \boldsymbol{x}_s)$.

$$D(T(\boldsymbol{x}_s)) \;=\; \boldsymbol{V}^{\mathrm{T}}(\boldsymbol{x}_s)d\boldsymbol{t}$$

Squaring this equation, results in:

$$D^2(T(\boldsymbol{x}_s)) \;=\; d\boldsymbol{t}^{\mathrm{T}}\boldsymbol{V}(\boldsymbol{x}_s)\boldsymbol{V}^{\mathrm{T}}(\boldsymbol{x}_s)\,d\boldsymbol{t} \;=\; d\boldsymbol{t}^{\mathrm{T}}M(\boldsymbol{x}_s)d\boldsymbol{t} \tag{3.8}$$

where $M(\boldsymbol{x}_s) \;=\; \boldsymbol{V}(\boldsymbol{x}_s)\boldsymbol{V}^{\mathrm{T}}(\boldsymbol{x}_s)$ is a symmetric, positive semi-definite 6x6 matrix. Summing the quantity in Equation 3.8 over a set, *P*, of discrete surface points results in:

$$E_P(T(\boldsymbol{x}_s)) \;=\; \sum_{\boldsymbol{x}_s \in P} D^2(T(\boldsymbol{x}_s)) \;= d\boldsymbol{t}^{\mathrm{T}}\Big[\sum_{\boldsymbol{x}_s \in P} M(\boldsymbol{x}_s)\Big]d\boldsymbol{t} = d\boldsymbol{t}^{\mathrm{T}}\Psi_P d\boldsymbol{t} \tag{3.9}$$

where E_P is the sum of squared distance errors between the surface and the points in *P*, and Ψ_P is the sum of the $M(\boldsymbol{x}_s)$ matrices evaluated at each point. E_P is a first order approximation of the registration error metric of Equation 3.1; the points in *P* correspond to the discrete point measurements, D_i, of Equation 3.1. The matrix Ψ_P is a scatter matrix which contains information about the distribution of the original $\boldsymbol{V}(\boldsymbol{x}_s)$ vectors over all points in the set *P*. This matrix is similar to the *stiffness* matrix in robotic grasp analysis [Mishra and Silver, 1989].

Performing principal component analysis [Kendall and Stuart, 1977] upon Ψ_P results in a form which is more easily interpreted:

$$\Psi_P \;=\; Q\Lambda Q^{\mathrm{T}} \;=\; \begin{bmatrix}\boldsymbol{q}_1 & \boldsymbol{q}_2 & \boldsymbol{q}_3 & \boldsymbol{q}_4 & \boldsymbol{q}_5 & \boldsymbol{q}_6\end{bmatrix}\begin{bmatrix}\lambda_1 & & & & & \\ & \lambda_2 & & \varnothing & & \\ & & \lambda_3 & & & \\ & & & \lambda_4 & & \\ & \varnothing & & & \lambda_5 & \\ & & & & & \lambda_6\end{bmatrix}\begin{bmatrix}\boldsymbol{q}_1^{\mathrm{T}} \\ \boldsymbol{q}_2^{\mathrm{T}} \\ \boldsymbol{q}_3^{\mathrm{T}} \\ \boldsymbol{q}_4^{\mathrm{T}} \\ \boldsymbol{q}_5^{\mathrm{T}} \\ \boldsymbol{q}_6^{\mathrm{T}}\end{bmatrix} \tag{3.10}$$

where $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \lambda_4 \geq \lambda_5 \geq \lambda_6$ are the eigenvalues of Ψ_P, and $\boldsymbol{q}_i$ are the corresponding unit eigenvectors. Each eigenvector, $\boldsymbol{q}_i$, can be interpreted as a differential transformation represented as a 6-vector; the first three elements are the translation components, and the last three elements are the rotation components.

Substituting Equation 3.10 into Equation 3.9 results in:

$$E_P(T(\boldsymbol{x}_s)) = d\boldsymbol{t}^{\mathrm{T}} Q \Lambda Q^{\mathrm{T}} d\boldsymbol{t} = \sum_{i=1}^{6} \lambda_i (d\boldsymbol{t}^{\mathrm{T}} \boldsymbol{q}_i)^2 \tag{3.11}$$

From Equation 3.11 it can be seen that the eigenvector $\boldsymbol{q}_1$ corresponding to the largest eigenvalue represents the *transformation of maximum constraint*. Perturbing the points in the set *P* by the differential transformation $\boldsymbol{q}_1$ will result in the largest possible change in E_P from among all possible transformation perturbations. Similarly, the differential transformation represented by the eigenvector $\boldsymbol{q}_6$ corresponds to the *transformation of maximum freedom*. Perturbing the points by this transformation will result in the smallest possible change in E_P from among all possible transformation perturbations. In general, an eigenvalue, λ_i, is proportional to the rate of change of error, E_P, induced by a small transformation in the direction specified by $\boldsymbol{q}_i$. The result presented in Equation 3.11 is equivalent to the one presented by the Ohio State group [Menq et al., 1992].

A special situation occurs when some of the λ_i are close to or equal to zero. For each such eigenvalue, a singularity exists such that perturbing the points in the direction specified by the corresponding eigenvector will result in no change in E_P. Clearly, such singularities complicate registration since it becomes impossible to localize an object in the direction corresponding to the singularity. For example, as demonstrated in Section 3.4.3, the slotted cylinder example of Figure 3-2 has two zero eigenvalues, one corresponding to a rotation about the central axis and the other corresponding to a translation along it.

From the above derivation, it is also possible to see that the minimum number of points required to unambiguously localize an arbitrarily-shaped object using shape-based registration is six. In order for the Ψ_P matrix to contain no zero-eigenvalues (i.e., have a rank of 6), it must be derived from a minimum of six linearly independent $\boldsymbol{V}(\boldsymbol{x}_s)$ vectors. This is only possible when 6 or more Data points are available.

In Section 3.4, it is demonstrated that discrete point measurement configurations which have a well conditioned Ψ_P matrix tend to result in better registration accuracy. In Chapter 4, it is shown that the primary goal of constraint synthesis is to select a minimally-sized set of discrete point measurements, *P,* which results in a well-conditioned Ψ_P matrix for a given object. The problem of how to mathematically define “well-conditioned” is addressed in the next section.

At this point, the reader may choose to refer to the experimental results presented in Section 3.4.1 which may help to clarify the above derivation.

3.3.2 Criterion Measures

This section addresses the question of how to mathematically define a *well conditioned* Ψ_P matrix from Equation 3.10 for the purposes of constraint analysis. In this section, several scalar measures of matrix conditioning are discussed, and one measure is selected for use in constraint analysis.

In Equation 3.11, principal component analysis is applied to the matrix Ψ_P resulting in 6 eigenvectors, $\boldsymbol{q}_i$, and 6 corresponding eigenvalues, λ_i, where $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \lambda_4 \geq \lambda_5 \geq \lambda_6$. Given a 6-vector, $\boldsymbol{x}$, the equation

$$\boldsymbol{x}^{\mathrm{T}} \Psi_P \boldsymbol{x} = 1$$

defines a 6-dimensional ellipsoid which will be referred to as the *uncertainty hyperellipsoid*.[1] The principal axes of this hyperellipsoid are defined by the eigenvectors of Ψ_P, and the lengths of the principal axes are given by the reciprocals of the square roots of the corresponding eigenvalues of Ψ_P, as shown in Figure 3-8. The length of each principal axis can be interpreted as a measure of localization uncertainty in the direction specified by the corresponding eigenvector; longer axes correspond to directions in which localization uncertainty is larger. There are several attributes of this hyperellipsoid which can be characterized by the scalar measures of matrix conditioning described below. For the purposes of accurate object localization via shape-based registration, the most important attribute is that the length of the largest axis of the hyperellipsoid be as small as possible. As demonstrated by the cylinder example presented in Section 3.4.3, registration accuracy can be compromised by just one small eigenvalue. As explained below, another attribute of the hyperellipsoid which is relevant for this problem is its eccentricity.

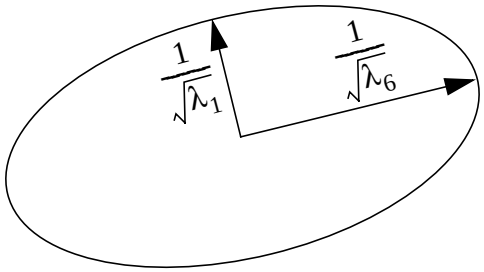


Figure 3-8: Two axes of the 6-dimensional uncertainty hyperellipsoid.

1. The uncertainty hyperellipsoid is similar to the manipulability hyperellipsoid from robot dexterity, and the observability hyperellipsoid from control systems.

In [Kim and Khosla, 1991] and [Nahvi and Hollerbach, 1996] a number of scalar measures of matrix conditioning are discussed in the context of robot dexterity measures. Due to the similarity between constraint analysis and robot dexterity, these measures are equally applicable to the work in this dissertation. Each of the criterion measures described in the above papers are scalar functions of the eigenvalues, λ_i. A summary of these measures as well as several others are presented in Table 3-1.

For each measure in the table (with the exception of eigenvalue variance) larger values result in better matrix conditioning and thus reduced localization uncertainty. For example, a large minimum eigenvalue results in a small maximum axis of the hyperellipsoid, and thus small localization uncertainty in the direction of maximum freedom. While the magnitude of the minimum eigenvalue is a weak indicator of hyperellipsoid volume, it provides no information regarding hyperellipsoid eccentricity.

Table 3-1: Scalar measures of matrix conditioning (6 dimensional case).

Measure Name	Equation	
Minimum Eigenvalue	λ_6	(3.12)
Inverse Condition Number	$\sqrt{\frac{\lambda_6}{\lambda_1}}$	(3.13)
Manipulability	$\sqrt{\lambda_1 \cdot \lambda_2 \cdot \lambda_3 \cdot \lambda_4 \cdot \lambda_5 \cdot \lambda_6}$	(3.14)
Geometric Mean of Eigenvalues	$\sqrt[6]{\lambda_1 \cdot \lambda_2 \cdot \lambda_3 \cdot \lambda_4 \cdot \lambda_5 \cdot \lambda_6}$	(3.15)
Arithmetic Mean of Eigenvalues	$\frac{\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + \lambda_5 + \lambda_6}{6}$	(3.16)
Eigenvalue Variance	$Var(\lambda_i)$	(3.17)
Isotropy Measure	$6 \cdot \frac{\sqrt[6]{\lambda_1 \cdot \lambda_2 \cdot \lambda_3 \cdot \lambda_4 \cdot \lambda_5 \cdot \lambda_6}}{\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + \lambda_5 + \lambda_6}$	(3.18)
Noise Amplification Index	$\frac{\lambda_6}{\sqrt{\lambda_1}}$	(3.19)

The inverse condition number is the square root of the ratio of the smallest to largest eigenvalues. Maximizing this measure tends to result in less eccentric hyperellipsoids (i.e., closer to hyperspherical). Since this measure is derived as a ratio of eigenvalues, it is dimensionless and thus is normalized for the number of measurements (i.e., points in the set P). However, the inverse condition number provides no information regarding the volume of the hyperellipsoid.

The manipulability measure [Yoshikawa, 1985] is the square root of the product of the eigenvalues. Similarly, the geometric mean of the eigenvalues is the mth root of this product, where m is the order of the matrix. It can be shown that both of these measures are related to the volume of the uncertainty hyperellipsoid. However, it is possible to have large values of manipulability despite one or more small eigenvalues. In addition, manipulability is not a good predictor of eccentricity. Note that manipulability has units of $[\text{length}]^m$.

The arithmetic mean is defined as the trace of the matrix divided by its order. This measure has an exacerbated version of the problem which is evident with the manipulability measure. For the arithmetic mean, it is possible to have eigenvalues which are equal to zero despite large values of the measure.

The variance of all of the eigenvalues can be computed as a measure of eccentricity. Small values correlate with low eccentricity. This measure is similar to the inverse condition number in that there is minimal information regarding the volume of the hyperellipsoid.

The isotropy measure proposed in [Kim and Khosla, 1991] is the ratio of the geometric to arithmetic means of the eigenvalues. The key point is that the arithmetic mean is always greater than or equal to the geometric mean. Equality is satisfied only when all of the eigenvalues are equal, which occurs in a hypersphere. Values of the isotropy measure close to 1 indicate a small hyperellipsoid with low eccentricity. Since both the geometric and arithmetic means have units of $[\text{length}]^2$, the isotropy measure is unitless.

The noise amplification index [Nahvi and Hollerbach, 1996] is the product of the inverse condition number and the square root of the minimum eigenvalue. The inverse condition number is an indicator of eccentricity, while the reciprocal of the square root of the minimum eigenvalue is the length of the longest principal axis of the hyperellipsoid. Therefore, the noise amplification index accounts for both the size and the eccentricity of the hyperellipsoid. Since the noise amplification index is computed from the largest and smallest eigenvalues, it implicitly includes information regarding the remaining eigenvalues (i.e., all eigenvalues lie between these two). In [Nahvi and Hollerbach, 1996] it is proven that the noise amplification index provides an upper bound on the amplification of unwanted noise in residual measurements (e.g., discrete point Data measurement noise and errors in the Model) to noise in the estimated parameters (e.g., registration pose parameters).

For the work presented in this dissertation, the noise amplification index (NAI) was used as a condition measure for the matrix Ψ_P. What follows is a justification of this selection which parallels a similar presentation in [Nahvi and Hollerbach, 1996].

The justification begins by rewriting Equation 3.9 as:

$$E_P(T(\boldsymbol{x}_s)) \;=\; \sum_{\boldsymbol{x}_s \in P} D^2(T(\boldsymbol{x}_s)) \;=\; \|\Delta D\|^2 \;=\; \Delta \boldsymbol{t}^{\mathrm{T}} \Psi_P \Delta \boldsymbol{t}, \tag{3.20}$$

in which ΔD is a column N-vector of the residuals associated with each of the points $\boldsymbol{x}_s \in P$. To use notation similar to that in [Nahvi and Hollerbach, 1996], the pose parameters relative to the global minimum pose can be represented as a column 6-vector, Δt (see Equation 3.3). Since the matrix Ψ_P relates ΔD to Δt for differential transformations, the following relation holds [Strang, 1980]:

$$\sqrt{\lambda_6} \le \frac{\|\Delta D\|}{\|\Delta t\|} \le \sqrt{\lambda_1} \tag{3.21}$$

where λ_1 and λ_6 are the maximum and minimum eigenvalues of Ψ_P, respectively. This equation can be rearranged into the inequality

$$\sqrt{\lambda_6}\,\|\Delta t\| \le \|\Delta D\|, \tag{3.22}$$

in which the left side places a lower bound on the magnitude of the registration residuals as a function of the magnitude of the pose parameter vector. This result is fundamental since it illustrates the need to maximize the minimum eigenvalue, λ_6, in constraint analysis. Large values of λ_6 will ensure that when the relative pose between Data and Model, Δt, is perturbed from the global minimum, the perturbation will be reflected in the residuals, ΔD, a requirement for accurate shape-based registration.

Maximizing the smallest eigenvalue, λ_6, will minimize the maximum length of the uncertainty hyperellipsoid. It is now shown that a second condition suggests the need to minimize the eccentricity of the hyperellipsoid as well. Assume that there is unwanted noise associated with the registration residuals, ΔD, due to sensing noise in the Data acquisition process or errors in the surface Model. This noise can be represented as a column N-vector, $\delta\Delta D$, which is added to the residual vector, ΔD. As a result of this noise, the transformation parameters output by the registration process, Δt, will also be noisy, and the resulting noise components can be represented as a column 6-vector, $\delta\Delta t$. Given these terms, the following inequality holds [Strang, 1980],

$$\frac{\|\delta\Delta t\|}{\|\Delta t\|} \le \sqrt{\frac{\lambda_1}{\lambda_6}}\,\frac{\|\delta\Delta D\|}{\|\Delta D\|}, \tag{3.23}$$

in which the square-root of the ratio of the eigenvalues is the condition number of Ψ_P (see Equation 3.13), and the two ratios are relative errors. This inequality places an upper bound on the relative error introduced in the pose parameters as a result of the relative error due to the residual noise. For shape-based registration, it is desirable to minimize the effect that the residual noise, $\delta\Delta D$, has upon pose parameter noise, $\delta\Delta t$. However, minimizing the condition number alone will not guarantee that this condition is satisfied. This is because Equation 3.23 only deals with the *relative* errors. In registration, it is the absolute error terms which are relevant (i.e., the ratio $\|\delta\Delta t\| / \|\delta\Delta D\|$ should be small, independent of the value of $\|\Delta D\|$). This problem can be reconciled by substituting the lower bound on $\|\Delta D\|$ from Equation 3.22 into the inequality of Equation 3.23 resulting in:

$$\frac{\|\delta\Delta t\|}{\|\Delta t\|} \leq \sqrt{\frac{\lambda_1}{\lambda_6}} \frac{\|\delta\Delta D\|}{\sqrt{\lambda_6}\|\Delta t\|} \tag{3.24}$$

which can be simplified to:

$$\|\delta\Delta t\| \leq \frac{\sqrt{\lambda_1}}{\lambda_6}\|\delta\Delta D\| \tag{3.25}$$

in which the ratio containing the eigenvalues is the reciprocal of the noise amplification index (NAI). This result shows that the NAI is the maximum amplification factor between noise in the residuals and noise in the pose parameters. Maximizing the NAI will ensure that the minimum eigenvalue is large (which satisfies the requirement of Equation 3.22), and that the uncertainty hyperellipsoid is not too eccentric (which satisfies the requirement of Equation 3.23). For these reasons, the NAI has been selected for use in this dissertation.

Additional details on the NAI, as well as experimental results which compare the NAI to other criterion measures can be found in [Nahvi and Hollerbach, 1996]. Examples of the use of the NAI as a scalar measure of conditioning in constraint analysis are presented in this dissertation beginning in Section 3.4.

3.3.3 Scale Dependence

In the derivation of constraint analysis in Section 3.3.1, an implicit weighting factor determines the relative importance of rotational versus translational constraint. This weighting factor is a function of the scale of the surface being analyzed. In this section, a normalization is described which reduces the dependence of surface size upon constraint analysis. It is no coincidence that the same problem has been noted in the context of robot dexterity measures [Doty et al., 1995] [Hollerbach, 1993] [Sugimoto, 1989].

Figure 3-7 indicates that the rotational component of the sensitivity vector is scale dependent. The magnitude of the translation induced at the point $\boldsymbol{x}_s$ by a fixed rotation about the origin is given by $\|d\theta \times \boldsymbol{x}_s\|$. This expression is a function of the distance between the point and the origin, $\|\boldsymbol{x}_s\|$. Thus, for a fixed rotation, $d\theta$, the induced distance, $D_\theta(\boldsymbol{x}_s)$, scales linearly as the surface is scaled about the origin, as illustrated below in Figure 3-9. Case I shows a planar curve which has undergone a rotation $d\theta$ about the origin.[1] After the rotation, the distance between the curve and the point $\boldsymbol{x}_s$ is $D_\theta(\boldsymbol{x}_s)$, as seen in Equation 3.7. In Case II, an identical planar curve which has been contracted about the origin has undergone a rotation about the origin of the *same* magnitude as in Case I. Due to the term $\boldsymbol{x}_s$ in Equation 3.7, the resulting induced distance $D'_\theta(\boldsymbol{x}_s)$ differs from $D_\theta(\boldsymbol{x}_s)$ by a factor which is proportional to the ratio of curve sizes. In general, for a fixed rotation about the origin, the induced distance, $D_\theta(\boldsymbol{x}_s)$, is proportional to the size of the surface, assuming that the surface is scaled about the origin.

Now consider the translational component of the sensitivity vector, which from Figure 3-6 is seen to be scale independent. From Equation 3.6 recall that:

$$D_\tau(\boldsymbol{x}_s) = \boldsymbol{n}^{\mathrm{T}} d\tau$$

Since the right side of this equation depends only upon the surface normal, the distance induced by a translation $D_\tau(\boldsymbol{x}_s)$, is independent of the surface size as illustrated in

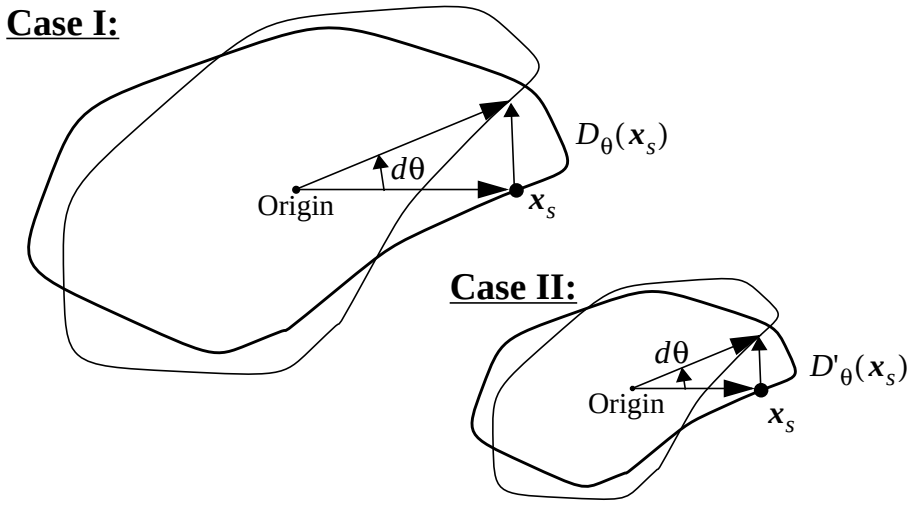


Figure 3-9: Scale dependence of the rotational component of constraint analysis. The distance induced at a point by a rotation through a fixed angle is a function of scale.

1. Planar curves are used for illustrative purposes; however, the same reasoning can be applied to 3-D surfaces. The magnitudes of the rotations shown in the figure are exaggerated for clarity.

Figure 3-10. In Case I, the large planar curve has been translated by a fixed amount $d\tau$ with a resulting induced distance, $D_\tau(\boldsymbol{x}_s)$. In Case II, the smaller curve has been translated by the same amount, $d\tau$, resulting in an induced distance which is *identical* to that for Case I. In contrast to the rotational case, the distance induced by a fixed translation is not proportional to the size of the surface.

If $\|d\tau\| = \|d\theta\|$, and x_s is a fixed surface point not near the origin, consider what happens to the ratio

$$\alpha = \frac{D_\theta(\boldsymbol{x}_s)}{D_\tau(\boldsymbol{x}_s)} \tag{3.26}$$

as the surface is scaled about its centroid. For very large surfaces $\alpha \gg 1$, while for very small surfaces $\alpha \ll 1$. The condition $\alpha = 1$ will be satisfied only when the surface size is such that $\|\boldsymbol{x}_s\| = 1$ for the given point. When $\alpha = 1$ then $D_\tau(\boldsymbol{x}_s) = D_\theta(\boldsymbol{x}_s)$, which has the satisfying interpretation that a translation of unit magnitude or a rotation of unit magnitude will result in identical induced distances.

Since the results of constraint analysis are intimately related to Equations 3.6 and 3.7 upon which the above ratio, α, is based, the results of constraint analysis are scale dependent as well. The more general problem is that constraint analysis attempts to treat rotations and translations in a unified framework, despite different underlying units. In the robot dexterity

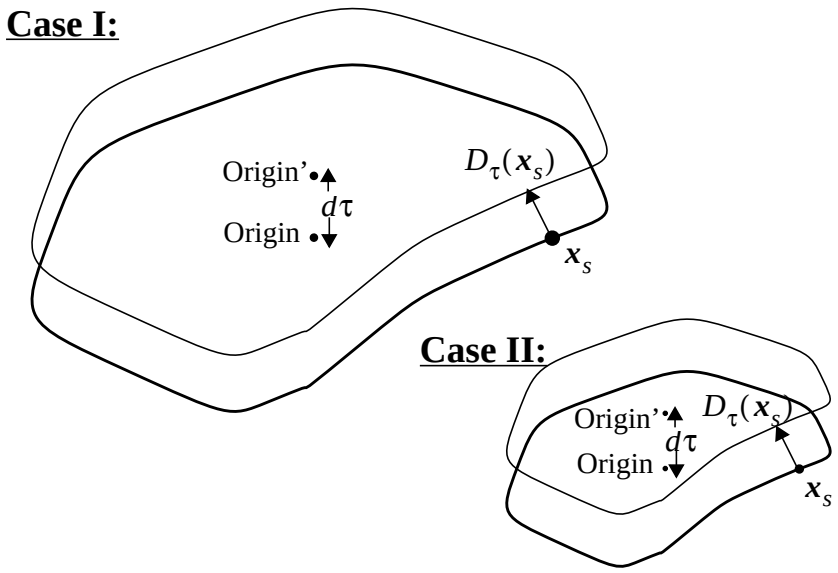


Figure 3-10: Scale independence of the translational component of constraint analysis. The distance induced at a point by a translation of fixed length is not a function of scale.

literature, various types of task-dependent normalization and scaling are proposed to deal with the scaling problems [Doty et al., 1995] [Hollerbach, 1993] [Sugimoto, 1989]. This same problem has also been noted in the context of visual tracking [Rehg and Witkin, 1991]. The solution used in this dissertation (which is also task-dependent) is to pre-normalize the size of surfaces to which constraint analysis is applied, as described below.

The first step in scale normalization is to shift the surface such that the origin is coincident with the centroid to ensure that scaling occurs about the centroid. Next, the surface is scaled such that the average distance between the surface and the origin is 1. When the surface is represented by a triangle mesh containing *N* vertices, this can be achieved by computing a scale factor:

$$k = \frac{1}{N}\sum_{V} \|\boldsymbol{x}_s\| \tag{3.27}$$

where the summation is over the set, *V,* of all vertices in the mesh. Each vertex is then scaled by *k,* so that:

$$\frac{1}{N}\sum_{V} \|\boldsymbol{x}_s^*\| = 1 \tag{3.28}$$

where $\boldsymbol{x}_s^* = \boldsymbol{x}_s / k$ are the scaled vertices. By pre-normalizing surface models in this manner, the results of constraint analysis can be made independent of scale. The motivation behind the average distance of 1 in Equation 3.28 is to equate translational and rotational units. Recall that the ratio, α, in Equation 3.26 will equal 1 only when the surface size is such that $\|\boldsymbol{x}_s\| = 1$. While this condition cannot be satisfied simultaneously for all surface points, Equation 3.27 ensures that the *average* distance from the surface to the origin is 1. Therefore, on average a unit translation will contribute the same to constraint analysis as a unit rotation. The results of an experiment which validates this normalization approach are presented in Appendix C. Unless otherwise mentioned, all constraint analysis results presented in this dissertation include pre-scaling by the method described above.

3.3.4 Coordinate System Dependence

This section shows that there is a dependence between the results of constraint analysis and the position of the origin of the constraint analysis coordinate system. Judicious selection of this coordinate system can improve the sensitivity of constraint analysis, and is shown to be a function of the geometry of the Model.

Figure 3-11 illustrates the effect of changing the origin of the constraint analysis coordinate system upon a displacement induced at a point on an object's surface. Rotations through an

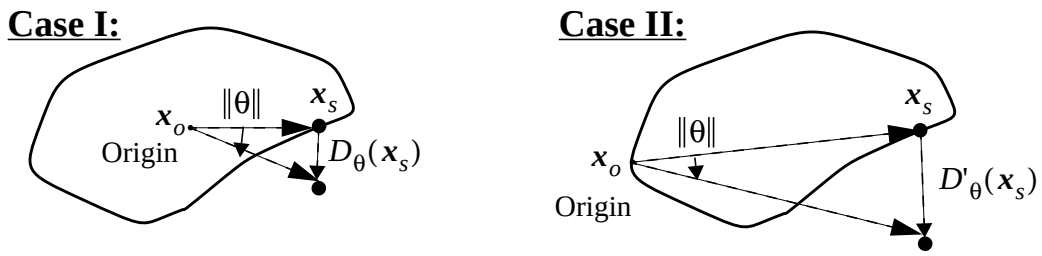


Figure 3-11: Coordinate system dependence of the displacement induced at a given point on a surface by a fixed rotation about the origin (rotations exaggerated for clarity). Displacement is a function of two factors: the distance between the point, $\boldsymbol{x}_S$, and the rotation center, $\boldsymbol{x}_O$; and the angle between the vector ($\boldsymbol{x}_S$- $\boldsymbol{x}_O$) and the surface normal, $\boldsymbol{n}$, at the point, $\boldsymbol{x}_S$ (for differential rotations only).

angle, $\|\theta\|$, about two different centers of rotation result in different induced distances between the point and the surface. This dependence on coordinate system only occurs for rotations and not for translations.

In the derivation of constraint analysis, it is shown in Equation 3.7 that the distance induced by a rotation about a given axis between a point and a surface can be expressed as:

$$D_\theta(\boldsymbol{x}_s) = (\boldsymbol{x}_s \times \boldsymbol{n})^{\mathrm{T}}\theta \tag{3.29}$$

where $D_\theta(\boldsymbol{x}_s)$ is the induced distance, $\boldsymbol{x}_s$ is a point on the surface, $\boldsymbol{n}$ is the unit normal to the surface evaluated at $\boldsymbol{x}_s$, $\boldsymbol{x}_s \times \boldsymbol{n}$ is the cross-product of the radius vector with the surface normal, and $\theta = \begin{bmatrix}\omega_x & \omega_y & \omega_z\end{bmatrix}^{\mathrm{T}}$ is a vector with components representing small rotations about the *x*, *y*, and *z* axes, respectively. Equation 3.29 can be generalized to incorporate the coordinate system origin, $\boldsymbol{x}_o$ as

$$D_\theta(\boldsymbol{x}_s - \boldsymbol{x}_o) = ((\boldsymbol{x}_s - \boldsymbol{x}_o) \times \boldsymbol{n})^{\mathrm{T}}\theta. \tag{3.30}$$

From this equation, it can be seen that the induced displacement, $D_\theta(\boldsymbol{x}_s - \boldsymbol{x}_o)$, is a function of the distance between the surface point, $\boldsymbol{x}_s$, and the origin, $\boldsymbol{x}_o$. Assuming a constant value of $\|\boldsymbol{x}_s - \boldsymbol{x}_o\|$, the displacement $D_\theta(\boldsymbol{x}_s - \boldsymbol{x}_o)$ is maximized when the two vectors, $(\boldsymbol{x}_s - \boldsymbol{x}_o)$ and $\boldsymbol{n}$, are orthogonal and minimized when they are parallel.

Following the constraint analysis derivation of Section 3.3.1, Equation 3.30 can be squared and summed over multiple surface points to create an energy measure similar to the one from Equation 3.9:

$$E = \sum_i D_\theta^2(\boldsymbol{x}_{si} - \boldsymbol{x}_o) = \sum_i [((\boldsymbol{x}_{si} - \boldsymbol{x}_o) \times \boldsymbol{n}_i)^{\mathrm{T}} \theta]^2 \tag{3.31}$$

where $\boldsymbol{x}_{si}$ are the individual Data points on the surface and $\boldsymbol{n}_i$ are the associated surface normals. In the remainder of this section, it is shown that placing the origin of the constraint analysis coordinate system, $\boldsymbol{x}_o$, at a location which is a function of the $\boldsymbol{x}_{si}$ and the $\boldsymbol{n}_i$ results in desirable behavior.

For a given Model and Data set, Equation 3.31 defines an energy surface which is a function of the rotation 3-vector, θ, centered about the registration global minimum (i.e., $\theta = 0$). This energy surface is illustrated schematically in Figure 3-12, in which the y-axis represents the energy, and the x-axis represents the 3-dimensional rotation space. The global minimum corresponds to the case in which the Model and Data are perfectly aligned.

From Equation 3.31 it can be seen that varying the location of $\boldsymbol{x}_o$ will affect the shape of the energy surface, and in particular will change the slope of the surface in the vicinity of the global minimum. To derive a mathematical measure of this slope, it is necessary to compute the second derivative of E with respect to θ (i.e., a Hessian matrix). As a first step, the gradient of E can be computed from Equation 3.31 as

$$\frac{dE}{d\theta} = 2\sum_i [(\boldsymbol{x}_{si} - \boldsymbol{x}_o) \times \boldsymbol{n}_i]^{\mathrm{T}} \theta [(\boldsymbol{x}_{si} - \boldsymbol{x}_o) \times \boldsymbol{n}_i]^{\mathrm{T}} \tag{3.32}$$

in which $dE/d\theta$ is a row 3-vector. The Hessian can then be computed by differentiating Equation 3.32 with respect to θ resulting in

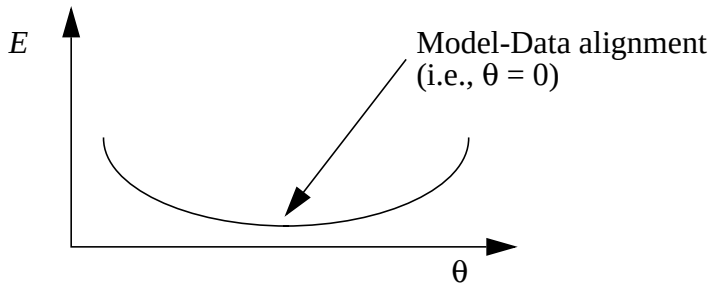


Figure 3-12: Schematic illustration of the energy surface as a function of the rotation 3-vector, θ.

$$\frac{d^2E}{d\theta} = 2\sum_i [(\boldsymbol{x}_{si} - \boldsymbol{x}_o) \times \boldsymbol{n}_i][(\boldsymbol{x}_{si} - \boldsymbol{x}_o) \times \boldsymbol{n}_i]^{\mathrm{T}} \tag{3.33}$$

which is a 3x3 real symmetric matrix. The eigenvalues of this matrix contain information about the slope of the energy surface in the region of the global minimum. A large eigenvalue is associated with a steep slope of the energy surface in the direction of its corresponding eigenvector. Similarly, a small eigenvalue is associated with a shallow slope in the direction of its corresponding eigenvector.

Recall from Figure 3-1 that constraint analysis provides a criterion measure to constraint synthesis for the purpose of Intelligent Data Selection (IDS). The goal of IDS is to select configurations of discrete point Data sets which result in the *largest* possible changes in energy as a result of arbitrary, small transformations. Given a choice between an energy surface which is locally steep near the global minimum (i.e., has a large maximum eigenvalue of the Hessian matrix of Equation 3.33) and one which is locally shallow (i.e., has a small maximum eigenvalue), for the purposes of constraint analysis, it is always desirable to choose the one which is locally *shallow.* This may seem counterintuitive; however, selection of $\boldsymbol{x}_o$ such that the energy surface is shallow will reflect the conditions under which a given Data set provides the least geometric constraint. This choice is desirable since constraint analysis should provide a criterion measure under the conditions for which a Data set will perform the worst (i.e., result in the least geometric constraint and largest registration error).

Based upon the above argument, one approach for selecting a coordinate system origin, $\boldsymbol{x}_o$, is to position it such that the smallest eigenvalue of the matrix in Equation 3.33 is minimized. This can be achieved by starting with the following observation. Given an *k x k*, real, symmetric matrix, $\boldsymbol{A}$, and an arbitrary *k*-vector, $\boldsymbol{X}$, the following inequality holds:

$$\boldsymbol{X}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{X} \geq \lambda_{min}\|\boldsymbol{X}\|^2 \tag{3.34}$$

where λ_{min} is the minimum eigenvalue of the matrix $\boldsymbol{A}$. In other words, the quadratic form $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{X}$ places an upper bound on the smallest eigenvalue of $\boldsymbol{A}$, λ_{min}. Assuming that $\|\boldsymbol{X}\| = 1$, minimizing the left side of Equation 3.34 will minimize an upper bound on the minimum eigenvalue, λ_{min}. Substituting the Hessian matrix of Equation 3.33 for $\boldsymbol{A}$ in Equation 3.34 results in the expression

$$2\sum_i \boldsymbol{X}^{\mathrm{T}}[(\boldsymbol{x}_{si} - \boldsymbol{x}_o) \times \boldsymbol{n}_i][(\boldsymbol{x}_{si} - \boldsymbol{x}_o) \times \boldsymbol{n}_i]^{\mathrm{T}}\boldsymbol{X} \geq \lambda_{min}. \tag{3.35}$$

Since the two inner product terms in this expression are scalars, the left side of this equation can be rewritten as

$$2\sum_i [\boldsymbol{X}^{\mathrm{T}}[(\boldsymbol{x}_{si} - \boldsymbol{x}_o) \times \boldsymbol{n}_i]]^2 . \tag{3.36}$$

Applying the Schwarz inequality to this expression results in

$$2\sum_i [\boldsymbol{X}^{\mathrm{T}}[(\boldsymbol{x}_{si} - \boldsymbol{x}_o) \times \boldsymbol{n}_i]]^2 \leq 2\sum_i \left\|(\boldsymbol{x}_{si} - \boldsymbol{x}_o) \times \boldsymbol{n}_i\right\|^2 \tag{3.37}$$

where the fact that $\|\boldsymbol{X}\| = 1$ has been used to eliminate this term from the right side. The cross product term can be represented using the following skew-symmetric matrix

$$\tilde{\boldsymbol{n}}_i = \begin{bmatrix} 0 & -n_i^z & n_i^y \\ n_i^z & 0 & -n_i^x \\ -n_i^y & n_i^x & 0 \end{bmatrix} \tag{3.38}$$

where n_i^x, n_i^y, n_i^z are the elements of the *i*th normal vector, $\boldsymbol{n}_i$. Therefore, the right side of Equation 3.37 can be rewritten as

$$2\sum_i \left\|\tilde{\boldsymbol{n}}_i \boldsymbol{x}_o - \tilde{\boldsymbol{n}}_i \boldsymbol{x}_i\right\|^2 . \tag{3.39}$$

This expression provides an upper bound on the left side of Equation 3.35, and therefore, an upper bound on the minimum eigenvalue, λ_{min}. Therefore, the goal is to minimize Equation 3.39 via judicious selection of $\boldsymbol{x}_o$, which can be done using a least squares formulation. The value of $\boldsymbol{x}_o$ which minimizes Equation 3.39 in the least squares sense is identical to the least squares solution of

$$\begin{bmatrix} \tilde{\boldsymbol{n}}_1 \\ \tilde{\boldsymbol{n}}_2 \\ : \\ \tilde{\boldsymbol{n}}_k \end{bmatrix} \boldsymbol{x}_o = \begin{bmatrix} \tilde{\boldsymbol{n}}_1 \boldsymbol{x}_1 \\ \tilde{\boldsymbol{n}}_2 \boldsymbol{x}_2 \\ : \\ \tilde{\boldsymbol{n}}_k \boldsymbol{x}_k \end{bmatrix} \tag{3.40}$$

which is of the form $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$. The least squares solution can be computed as

$$\tilde{\boldsymbol{x}} = [\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}]^{-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{b} . \tag{3.41}$$

Applying Equation 3.41 to Equation 3.40, and expressing the inner products as summations results in

$$\tilde{\boldsymbol{x}}_o = \left[\left(\sum_i \tilde{\boldsymbol{n}}_i^{\mathrm{T}} \tilde{\boldsymbol{n}}_i\right)^{-1}\left(\sum_i \tilde{\boldsymbol{n}}_i^{\mathrm{T}} \tilde{\boldsymbol{n}}_i \boldsymbol{x}_i\right)\right]. \tag{3.42}$$

This result means that locating the origin of the constraint analysis coordinate system, $\boldsymbol{x}_o$, according to Equation 3.42 will tend to minimize the smallest eigenvalue of the Hessian matrix of Equation 3.33, and therefore will result in the shallowest slope of the corresponding energy surface near the global minimum.

This result is intuitively satisfying. Minimizing the right side of Equation 3.37 by setting $\boldsymbol{x}_o = \tilde{\boldsymbol{x}}_o$ tends to minimize the angular differences between corresponding pairs of vectors $(\boldsymbol{x}_{si} - \boldsymbol{x}_o)$ and $\boldsymbol{n}_i$, since the cross product terms will be smallest when these vector pairs are parallel. By positioning $\boldsymbol{x}_o$ at $\tilde{\boldsymbol{x}}_o$, the surface will appear as "spherical" as possible from the point of view of the origin. Due to the $(\boldsymbol{x}_{si} - \boldsymbol{x}_o)$ term in the cross product, points which are farther from the origin will be more strongly weighted when evaluating the "spherical" criterion. In the case when the surface is a sphere, $\tilde{\boldsymbol{x}}_o$ will correspond to the sphere's centroid. Referring to Figure 3-7, it can be seen that choosing the origin according to Equation 3.42 will minimize the change in energy, *E*, as a function of small, arbitrary rotations about the global minimum. Thus, the solution of Equation 3.42 satisfies the goal of placing the coordinate system origin in the "worst" possible location.[1]

3.4 Experimental Results

This section contains a number of experimental results which demonstrate the utility of constraint analysis. In Section 3.4.1, two simple examples of the intermediate and final results of constraint analysis are presented for a torus and helicoid. In order to relate the measure computed by constraint analysis to registration accuracy, it is necessary to perform large numbers of registration experiments. Section 3.4.2 outlines a framework for performing registration experiments which is used throughout the remainder of the dissertation. Section 3.4.3 presents experimental results which demonstrate the effectiveness of constraint analysis as a predictor of relative registration accuracy. These experiments are performed using a cube, a slotted cylinder, a femur, a pelvis and a face. Section 3.4.4 presents the results of an experiment which demonstrates the relation between Data noise and constraint analysis.

1. The result of Equation 3.42 was not discovered until most of the experiments for this dissertation were completed. Therefore, in the experimental results presented throughout the dissertation, it was the Model centroid, and not $\tilde{\boldsymbol{x}}_o$, which was used as the origin of the constraint analysis coordinate system. In practice, this substitution had minimal effect upon the results, since the distances between the $\tilde{\boldsymbol{x}}_o$ values and the centroids were small for the Models used in the experiments.

3.4.1 Simple Examples of Constraint Analysis

In Section 3.3.1, the sensitivity gradient, $\boldsymbol{V}(\boldsymbol{x}_s)$, is defined in Equation 3.4. This 6 dimensional vector represents the rate of change of point-to-surface distance between a point and an underlying surface as the point is perturbed by a small transformation. Figure 3-13 displays this gradient vector graphically over the surface of a torus. Before computation, the torus was scaled using Equation 3.27, and the origin was placed at the centroid of the vertices of the triangle mesh as described in the footnote on page 69. Each surface in the figure represents a different element of $\boldsymbol{V}(\boldsymbol{x}_s)$ corresponding to the 6 parameters of the transformation. The $\boldsymbol{V}(\boldsymbol{x}_s)$ vector was evaluated for every vertex of the triangle mesh representing the torus. Intensity encodes the rate of change of point-to-surface distance induced by the indicated transformation at a particular point; brighter regions indicate higher rates of change. For example, the surface labeled $\partial D/\partial t_x$ corresponds to the first element of $\boldsymbol{V}(\boldsymbol{x}_s)$, the component which measures sensitivity to a translation along the x-axis. For the translational components, surface points at which the induced motion is parallel to the surface normal result in a high rate of change, while points at which the induced motion is perpendicular to the surface normal result in a low rate of change. For rotational components, the rate of change at a surface point is related to the direction of the surface normal with respect to the axis of rotation (AOR) and the distance of the point from the AOR. For example, consider the surface labeled $\partial D/\partial \omega_y$ corresponding to a rotation about the y-axis. Surface points near the axis of rotation have a smaller rate of change than points distant from it. However, even points which are distant from the rotational axis may have small rates of change if the induced motion at a point is perpendicular to the direction of the surface normal at the point. An especially interesting case is the surface labeled $\partial D/\partial \omega_x$, rotation about the x-axis. Since the torus is symmetric about this axis, the rate of change should be zero everywhere. In practice, the small intensity variations seen in the figure are due to surface discretization errors which cause the computed surface normals to differ from those of an ideal torus.

Given the $\boldsymbol{V}(\boldsymbol{x}_s)$ vectors displayed in Figure 3-13, the eigenvalues and eigenvectors of Equation 3.11 can be computed. Table 3-2 presents the analysis results for the case in which all vertices of the triangle mesh surface model of the torus are included in the set P of Equation 3.9. The magnitude of each eigenvalue is shown in the horizontal bar graph on the left of the table. Each row of the table corresponds to an eigenvalue - eigenvector pair. The eigenvectors corresponding to the smallest and largest eigenvalues represent the transformations of maximum freedom and constraint, respectively. In this example, the transformation of maximum freedom corresponds to a rotation about the x-axis, which is the axis of rotational symmetry noted above. Theoretically, the smallest eigenvalue in this example should have a magnitude of zero; however, a small non-zero value is present due to discretization errors introduced by the triangle mesh. These discretization errors result in computed surface normals and radius vectors which differ from those of an ideal torus, thus causing devi-

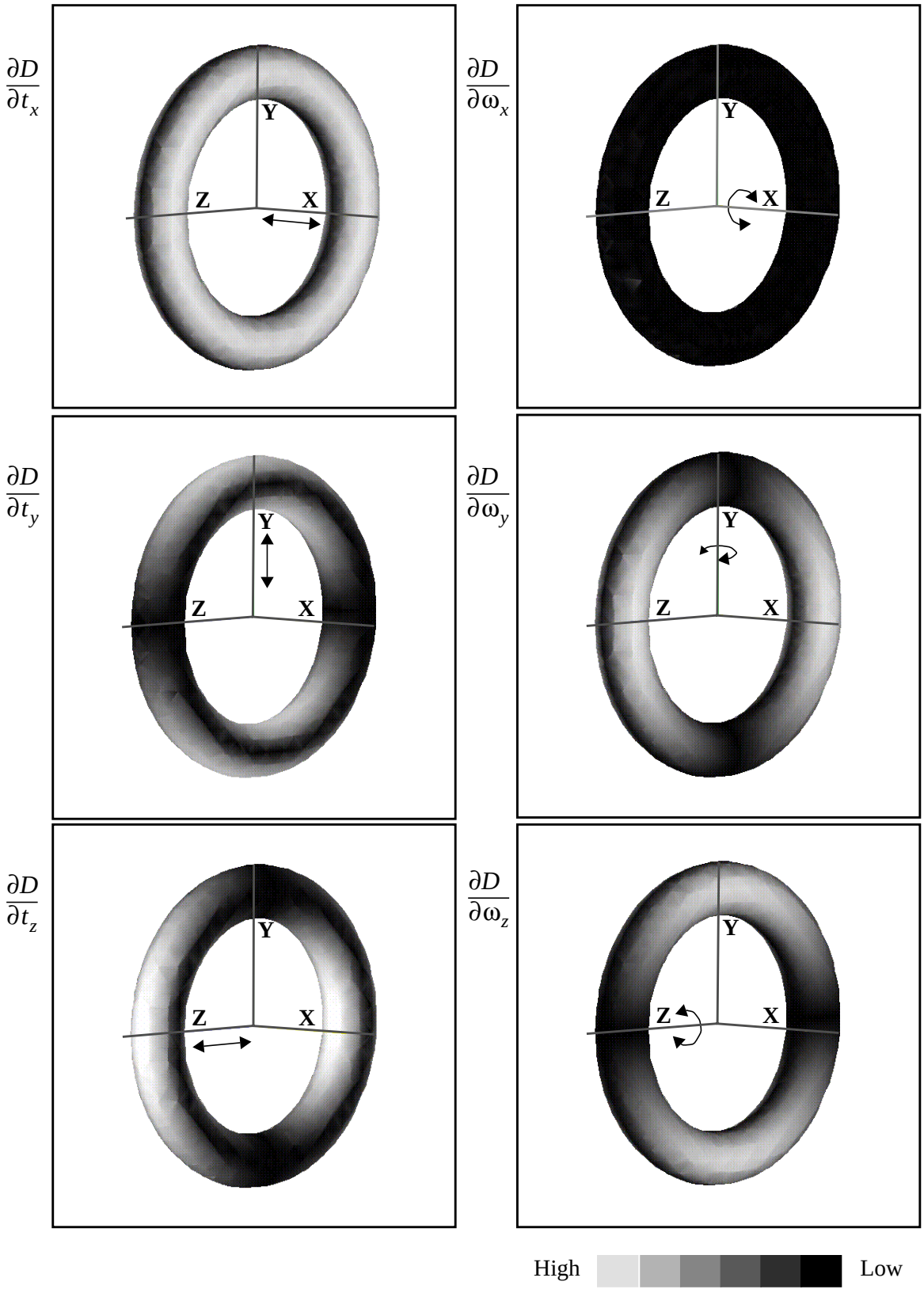


Figure 3-13: Components of the sensitivity gradient, $\boldsymbol{V}(\boldsymbol{x}_s)$, for a torus.

ations from the expected result. From Table 3-2, it can be seen that the transformation of maximum constraint corresponds to a translation along the x-axis. This agrees with the intuition from Figure 3-13 that the $\partial D / \partial t_x$ surface has the largest bright region. The remaining 4 eigenvectors in the table complete a basis set which spans the space of differential transformations. Due to the symmetry of the torus, these 4 eigenvectors are only unique to a rotation about the x-axis. For this example, constraint analysis correctly determines that it is impossible to localize the torus rotationally about the x-axis.

Table 3-2: Constraint analysis results for a torus.

Eigenvalue	Index	Eigenvectors					
		Translation Elements			Rotation Elements		
	1	0.00	0.00	0.00	1.00	0.00	0.00
	2	0.01	0.02	-0.08	0.00	0.61	0.78
	3	0.00	-0.01	-0.12	0.00	0.78	-0.62
	4	0.00	-0.72	0.68	0.00	0.10	0.01
	5	0.00	0.69	0.71	0.00	0.10	-0.03
	6	1.00	0.00	0.00	0.00	0.00	-0.01

1500.0 1000.0 500.0 0.0

In general, the eigenvalues resulting from constraint analysis are not dependent upon the *orientation* of the constraint analysis coordinate system. For example, the coordinate system shown in Figure 3-13 can be arbitrarily rotated about its origin with no change in the eigenvalues of Table 3-2 (and thus no change in the criterion function). The eigenvectors do vary as a function of coordinate system orientation, but remain fixed relative to the object itself (disregarding symmetries). *Translations* of the reference coordinate system will affect the results of constraint analysis for the reasons outlined in Section 3.3.4. As demonstrated in that section, placing the coordinate system origin at the location specified by Equation 3.42 will result in maximal sensitivity of constraint analysis. Nevertheless, it has been experimentally verified that coordinate system translations with magnitudes similar to the object size can be tolerated by constraint analysis. In particular, the singularities identified by constraint analysis are invariant to such small translational changes in coordinate system.

The results of constraint analysis can be computed at very high speeds. Computation is performed in three steps: calculation of the $\boldsymbol{M}(\boldsymbol{x}_s)$ matrices of Equation 3.8 for each vertex in the Model; computation of the matrix Ψ_P of Equation 3.9 over all vertices in the current Data configuration (i.e., the set *P*); and computation of the principal components as in Equation 3.10. Once the first step has been performed for a given Model, only the second and third steps must be computed for each new configuration. For the constraint analysis results of Table 3-2, computation times were roughly 20 ms for the first step, and 10 ms for

the second and third steps on an SGI Indigo 2 with a 200 MHz IP22 processor. High speed computation of constraint analysis is important since the constraint synthesis algorithm described in Chapter 4 requires a large number of constraint analysis evaluations (e.g., 10^6).

A second example of constraint analysis is demonstrated for the helicoidal surface shown in Figure 3-14. All of the vertices of the triangle mesh, except for those near the ends of the helicoid, are included in the set *P* for the analysis. The resulting eigenvalues and eigenvectors are presented in Table 3-3. In this example, the transformation of maximum freedom corresponds to an invariant twist about the z-axis. This twist corresponds to a translation along the z-axis, coupled with a rotation about the z-axis. The *pitch* of the helicoid can be recovered from this eigenvector as the ratio of translation to rotation, in this case 0.1. The transformation of maximum constraint is parallel to the z-axis. Once again, constraint analysis correctly determines that it is impossible to localize the pose of the helicoid along the identified twist.

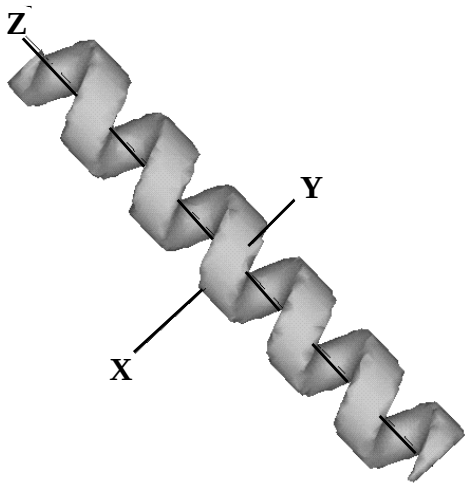


Figure 3-14: Surface model of a helicoid.

3.4.2 Framework for Registration Experiments

Throughout the remainder of the dissertation, experimental results from large numbers of controlled registration trials are reported. These experiments are used to demonstrate the relation between constraint analysis and registration, and to demonstrate the efficacy of constraint synthesis. This section describes the framework used in these registration experiments.

A registration experiment is defined as a set of registration trials. Each registration trial is a single execution of the speed-enhanced iterative closest point (EICP) algorithm described in Chapter 2. For each trial, all of the EICP parameter values described in Table 2-1 must be

specified, and these parameters remain constant for all trials within an experiment. Rather than reporting these parameters for every experiment, a set of default EICP parameters is presented in Table 3-4. EICP parameters will only be reported when the values differ from the defaults specified in the table.

The flow of a single registration experiment is outlined in Figure 3-15. The rectangles in the flowchart represent processing operations, and the ovals represent experiment parameters. An experiment begins by reading a description of the Model from a file, and initializing the experimental parameters which are also read from a file. Within an experiment, there are three possible methods of fixed-size discrete point Data set creation: random generation by

Table 3-3: Constraint analysis results for a helicoid.

Eigenvalue	Index	Eigenvectors					
		Translation Elements			Rotation Elements		
	1	0.00	-0.01	0.10	0.00	0.00	1.00
	2	-0.48	-0.14	0.03	0.79	-0.36	-0.01
	3	-0.05	-0.65	0.00	0.19	0.73	0.00
	4	0.78	0.23	0.03	0.56	0.11	0.00
	5	0.40	-0.71	0.03	-0.14	-0.57	-0.01
	6	-0.02	0.02	0.99	-0.03	0.02	-0.10

1500.0 1000.0 500.0 0.0

Table 3-4: Default EICP parameters for registration experiments. Only deviations from these parameters are reported in subsequent experiments.

Parameter	Description / Value
Termination criterion	Which ICP stopping condition is used? 1) change in relative transformation magnitude
Termination thresholds: ε_{Rr}, ε_{Tr}	$\varepsilon_{Rr} = 10^{-4}$, $\varepsilon_{Tr} = 10^{-4}$
k_{max}	Maximum EICP iteration count: $k_{max} = 500$
Minima suppression	Disabled
Outlier elimination	Disabled

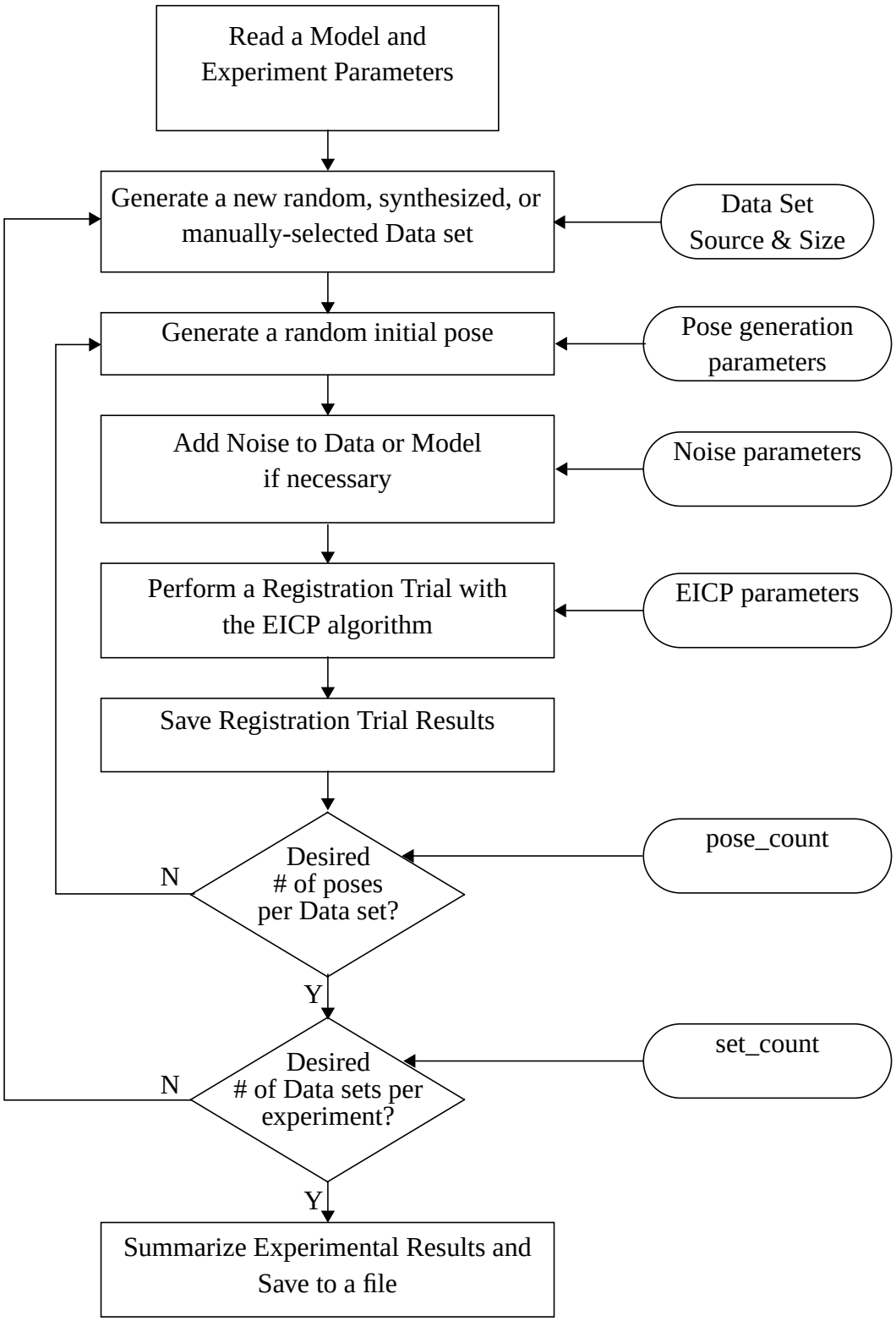


Figure 3-15: The flow of a single registration experiment. Rectangles represent processing operations and ovals represent experiment parameters.

randomly selecting vertices of the triangle mesh surface Model; optimal selection via the constraint synthesis methods of Chapter 4; and manual-selection by a human.

To evaluate the performance of a given Data set for use in registration, it is necessary to perform many registration trials beginning from different initial starting poses. Initial poses are randomly generated using two parameters: the maximum translation distance, τ_{imax}, and the maximum rotation angle, θ_{imax}. In order to ensure a fair comparison between Data sets, the same random initial poses are repeated for each set. A random translation is generated as follows:

$$\begin{aligned} t_x &= ran(-1.0, 1.0) \cdot \frac{\tau_{imax}}{\sqrt{3}} \\ t_y &= ran(-1.0, 1.0) \cdot \frac{\tau_{imax}}{\sqrt{3}} \\ t_z &= ran(-1.0, 1.0) \cdot \frac{\tau_{imax}}{\sqrt{3}} \end{aligned} \tag{3.43}$$

where (t_x, t_y, t_z) are the generated translation parameters, and $ran(x, y)$ is a function which returns a uniformly distributed random number in the range $[x, y]$. The effect of normalizing by the term $\tau_{imax}/\sqrt{3}$ is to ensure that the maximum possible magnitude of the resulting translation is τ_{imax}.

A random rotation can be generated in a similar manner by representing the rotation in the angle-axis representation discussed in Section 2.5.1. The random angle is generated as:

$$\theta = ran(-\theta_{imax}, \theta_{imax}) \tag{3.44}$$

Generating a uniformly distributed random vector on a unit sphere (or hemisphere in this case) requires several steps. First, a uniformly distributed random vector is generated using parameters specified by the bounding box of a unit hemisphere centered at the origin:

$$\begin{aligned} k_x^* &= ran(-1.0, 1.0) \\ k_y^* &= ran(-1.0, 1.0) \\ k_z^* &= ran(0.0, 1.0) \end{aligned} \tag{3.45}$$

where k_x^*, k_y^*, k_z^* are elements of a random vector, $\boldsymbol{k}^*$. If the magnitude of this vector, $\|\boldsymbol{k}^*\|$, is greater than 1, then the vector lies outside of the hemisphere and is discarded. Otherwise, the vector is projected onto the unit hemisphere:

$$\begin{aligned} k_x &= k_x^* / \|\boldsymbol{k}^*\| \\ k_y &= k_y^* / \|\boldsymbol{k}^*\| \\ k_z &= k_z^* / \|\boldsymbol{k}^*\| \end{aligned} \tag{3.46}$$

where k_x, k_y, k_z are elements of the transformed unit vector after normalization by $\|\boldsymbol{k}^*\|$. The reason for generating a vector on the unit hemisphere rather than the unit sphere is that in angle-axis representation, there is an equivalence between ($\boldsymbol{k}$, -θ) and (-$\boldsymbol{k}$, θ). Therefore, since θ is generated in the range $[-\theta_{imax}, \theta_{imax}]$, it is only necessary to generate positive values of $\boldsymbol{k}$ (i.e., values on a unit hemisphere). It has been empirically verified that this simple procedure generates uniformly distributed vectors on the unit hemisphere.

The next step in the registration experiment flow of Figure 3-15 is corruption of the Data, and (less frequently) the Model with random Gaussian noise. When physically measured Data points are used in an experiment, it is not necessary to corrupt the Data with noise; however, when the Data sets are created, corruption with noise results in Data points which more closely resemble real measurements. Given a 3-D point, $[x, y, z]^T$, the problem is to generate a random Gaussian noise vector such that the magnitude of this random vector has a specified expected value. Noise is added to each point as:

$$\begin{aligned} x' &= x + \delta_x \\ y' &= y + \delta_y \\ z' &= z + \delta_z \end{aligned} \tag{3.47}$$

where $\delta_x, \delta_y, \delta_z$ are the elements of the added noise vector, and the magnitude of this vector is given by:

$$\Delta = \sqrt{\delta_x^2 + \delta_y^2 + \delta_z^2} \tag{3.48}$$

The goal is to generate random Gaussian noise vectors such that:

$$E[\Delta] = \mu_\Delta \tag{3.49}$$

where $E[\ ]$ is the expected value operator and μ_Δ is a constant specified by the experimenter. It is shown in [Papoulis, 1984] that if $\delta_x, \delta_y, \delta_z$ are independent Gaussian random variables with zero mean and variance σ_δ^2, then Δ has a Maxwell distribution and:

$$E[\Delta] = 2\sqrt{\frac{2\sigma_\delta^2}{\pi}} \tag{3.50}$$

Therefore, the standard deviation of the random Gaussian noise variables can be expressed in terms of the expected vector magnitude, μ_Δ, as:

$$\sigma_\delta = \mu_\Delta\sqrt{\frac{\pi}{8}} = 0.627 \cdot \mu_\Delta \tag{3.51}$$

The desired behavior can be attained by generating the random variables δ_i from a Gaussian distribution with zero mean and standard deviation σ_δ. When generating noise in this manner, *maximum* values for the noise vector magnitude tend to be roughly a factor of 3 greater than the expected magnitude, μ_Δ.

Once noise has been added, a registration trial is performed using the specified EICP parameters. At the completion of the trial, all of the registration accuracy measures and transformations described in Section 2.5.1 are recorded. The process of generating random initial poses, adding noise, performing registration, and recording results is repeated until the desired number of poses is reached. A new Data set is then read from a file or created, and the entire process repeats until the desired number of Data sets have been processed. Finally, statistics are calculated for all trials within the experiment and the results are saved to a file for subsequent analysis. A complete list of the experimental parameters discussed in this section is presented in Table 3-5 for reference.

3.4.3 Relation of Constraint Analysis to Registration

In Section 3.1 above, registration problems using a cube and slotted cylinder are presented to motivate the need for constraint analysis. This section uses these and other objects to demonstrate the utility of constraint analysis as a predictor of relative registration accuracy.

Cube Examples

In Figure 3-3 on page 47, there are three Data configurations shown for registering a cube. Configuration C1 is a dense sampling of 25 points per face, and configurations C2 and C3 are samplings of 4 points per face with the points in C2 near the corners and the points in C3 near the centers. Tables 3-6, 3-7 and 3-8 show the eigenvalues and eigenvectors resulting from constraint analysis for these configurations. The first observation from these results is that the eigenvectors can be divided into two sets: those containing only translation components and those containing only rotation components. This behavior does not occur for arbitrarily-shaped objects or random Data configurations, but appears in this case due to the Data configuration symmetry with respect to the cube. A second observation is that eigenvalue magnitudes are larger for configuration C1 than for the other two configurations since

Table 3-5: Registration experiment parameters.

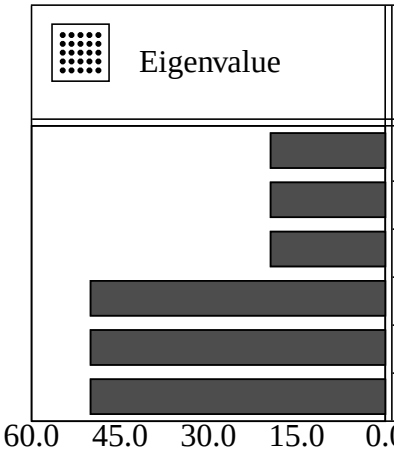
Parameter	Description
Data source	Method of Data creation: 1) Randomly generated. 2) Synthesized via constraint synthesis. 3) Manually-selected.
Data set size, N	Number of discrete points in the Data set.
τ_{imax}	Maximum translation distance used to generate random initial poses.
θ_{imax}	Maximum rotation angle used to generate random initial poses.
Noise state	Specifies whether noise is enabled or disabled.
μ_{Δ}	Expected value of noise vector magnitude.
pose_count	Number of poses per Data set.
set_count	Number of Data sets per experiment.

the eigenvalues are not normalized for the total number of Data points. A third observation is that the results of configurations C2 and C3 only differ in the eigenvalue magnitudes of the rotation components; C2 is more sensitive than C3 to rotations, while both configurations are equally sensitive to translations. This observation is verified in the experimental registration results below.

Using the framework from the previous section, a registration experiment was performed with the three Data configurations. The EICP and registration experiment parameters are shown in Table 3-9. In particular, note that local minima suppression (described in Section 2.3) is enabled. In these experiments, the length of an edge of the cube is 50.0 mm, so the expected value of the noise magnitude is 2.5% of the length of an edge, and the maximum noise magnitude is roughly 7.5% of this length.

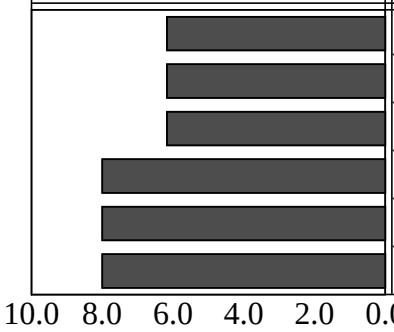
The results of this experiment are presented in Figure 3-16. The three graphs correspond to three different measures of registration accuracy: the maximum correspondence error (MCE); the translation error norm, $\|E_t\|$; and the rotation error norm, $\|E_\omega\|$ (see Section 2.5.1 for a discussion of these accuracy measures). In each case, the accuracy measure is plotted versus the noise amplification index (NAI) criterion measure described in Section 3.3.2. Recall that the NAI is computed as the ratio of the smallest eigenvalue to the square-root of the largest eigenvalue, and provides a scalar measure of localization sensitivity for a given Model and Data set. Each datum in the plot consists of a circle, an error bar,

Table 3-6: Constraint analysis results for cube with Data configuration C1



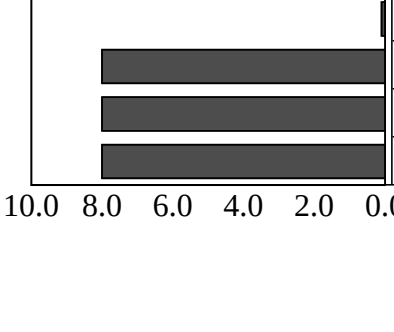
Eigenvalue	Index	Eigenvectors					
		Translation Elements			Rotation Elements		
	1	0.00	0.00	0.00	-0.02	-0.01	1.00
	2	0.00	0.00	0.00	1.00	-0.02	0.02
	3	0.00	0.00	0.00	0.02	1.00	0.01
	4	0.50	-0.50	-0.71	0.00	0.00	0.00
	5	0.15	0.85	-0.50	0.00	0.00	0.00
	6	0.85	0.15	0.50	0.00	0.00	0.00

Table 3-7: Constraint analysis results for cube with Data configuration C2



Eigenvalue	Index	Eigenvectors					
		Translation Elements			Rotation Elements		
	1	0.00	0.00	0.00	1.00	0.00	0.00
	2	0.00	0.00	0.00	0.00	1.00	0.00
	3	0.00	0.00	0.00	0.00	0.00	1.00
	4	1.00	0.00	0.00	0.00	0.00	0.00
	5	0.00	1.00	0.00	0.00	0.00	0.00
	6	0.00	0.00	1.00	0.00	0.00	0.00

Table 3-8: Constraint analysis results for cube with Data configuration C3



Eigenvalue	Index	Eigenvectors					
		Translation Elements			Rotation Elements		
	1	0.00	0.00	0.00	1.00	0.00	0.00
	2	0.00	0.00	0.00	0.00	1.00	0.00
	3	0.00	0.00	0.00	0.00	0.00	1.00
	4	1.00	0.00	0.00	0.00	0.00	0.00
	5	0.00	1.00	0.00	0.00	0.00	0.00
	6	0.00	0.00	1.00	0.00	0.00	0.00

Table 3-9: Parameters of cube registration experiment #1.

Parameter	Value
Data source	Read from file
τ_{imax}	5.0 mm
θ_{imax}	10 deg
Noise state	Enabled
μ_{Δ}	1.25 mm
pose_count	500
set_count	3
Minima suppression	Enabled
$\tau_{p\text{-}max}$	2.5 mm
$\theta_{p\text{-}max}$	5.0 deg
I_p	7 iterations

and two triangles. The circle corresponds to the mean of the associated accuracy measure computed over the total number of poses (i.e., 500) for the given Data set. The error bars represent the standard deviation, and the upward and downward facing triangles correspond to the minimum and maximum values of the accuracy measure, respectively. The graph of MCE provides an indication of the combined effect of rotation and translation error. It is apparent that configuration C1, which contains 150 Data points, has the smallest mean, maximum and minimum errors, as well as the smallest standard deviation. Configuration C2, which has roughly a factor of six fewer Data points than configuration C1, exhibits comparable registration accuracy, while the accuracy results for configuration C3 are relatively poor. An important observation from these results is that constraint analysis provides a good predictor of the relative registration accuracy of the three Data configurations; larger values of the NAI result in smaller registration errors. Note that constraint analysis provides no explicit measure of the *absolute* error which can be expected from registration. Absolute registration error is a function of factors which are not included in constraint analysis, including noise in the Data and Model. Therefore, the value of constraint analysis is derived from its ability to estimate the *relative* registration accuracy resulting from multiple Data configurations.

The results of Figure 3-16 demonstrate that for the cube example, rotation errors contribute more than translation errors to the overall error measured by the MCE. While this effect is

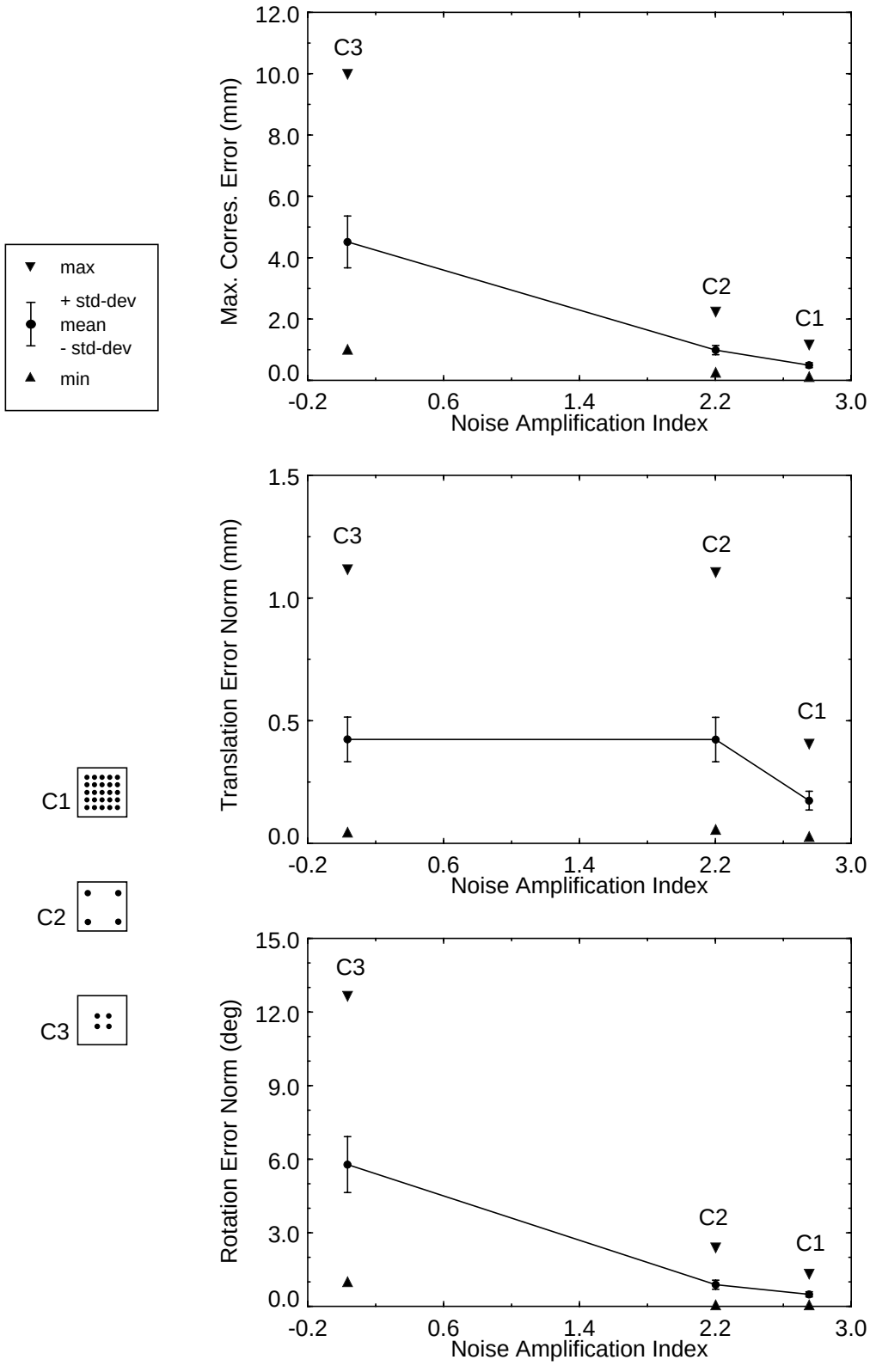


Figure 3-16: Results of cube registration experiment #1. Maximum correspondence error, translation error norm, and rotation error norm plotted versus the NAI for three Data configurations on a cube.

not predicted by the NAI values for the three configurations, it can be explained with reference to the eigenvalues of Tables 3-6, 3-7 and 3-8. From these tables, note that the translational eigenvalues for configurations C2 and C3 are identical, and the translation errors in Figure 3-16 for these two configurations are also identical. In contrast, the rotational eigenvalues for configuration C2 are larger than those for C3, and as expected, the rotation errors for C2 are smaller than those for C3. It is difficult to directly compare the eigenvalues for configuration C1 with the other two configurations since they were derived using different numbers of Data points. However, the larger eigenvalue magnitudes for configuration C1 and corresponding larger NAI, correctly predict the improved registration accuracy.

Recall from Section 2.5.1 that the use of correspondence error measures such as the maximum correspondence error (MCE) and the average correspondence error (*ACE*) requires the availability of a ground-truth registration transformation. Residual-based error measures such as the average residual error (*ARE*) described in Section 2.5.1, do not require ground-truth. It is interesting to compare plots of *ACE* and *ARE* for the above cube registration experiments. Figure 3-17 contains plots of *ACE* and *ARE* versus the NAI for the three cube Data configurations. As expected, the absolute error magnitudes in the plots differ since the *ACE* is computed using corresponding points, and the *ARE* is computed using closest points. Despite the magnitude differences, the similarity in general features is apparent. This similarity partially explains why the ICP algorithm converges; minimizing the *ARE* in the registration process also minimizes the *ACE*. If this condition were not satisfied for a given object, ICP would not converge to the global minimum. A convergence theorem for ICP which depends upon this observation appears in [Besl and McKay, 1992].

Slotted Cylinder Examples

This section presents constraint analysis and registration experiment results for a slotted cylinder using the two Data configurations, each containing 10 points, shown in Figure 3-18. The configuration on the left, S1, does not contain any points within the central slot or in the two end regions, while the configuration on the right, S2, contains points within both of these regions.

Constraint analysis results for the two slotted cylinder Data configurations are presented in Tables 3-10 and 3-11. Configuration S1 contains two singularities corresponding to translation along the central axis and rotation about it as demonstrated by the two zero eigenvalues in Table 3-10. As seen in Table 3-11, these singularities are not present for configuration S2 since Data are available in the end regions and within the slot. Note that the eigenvectors cannot be decoupled into pure rotational and translational components in this example.

Registration experiments were performed for the slotted cylinder with each Data configuration using the parameters specified in Table 3-12. The length of the cylinder's axis is 40 mm and the radius is 10 mm, so the expected magnitude of the noise is 2.0% of the radius. A plot

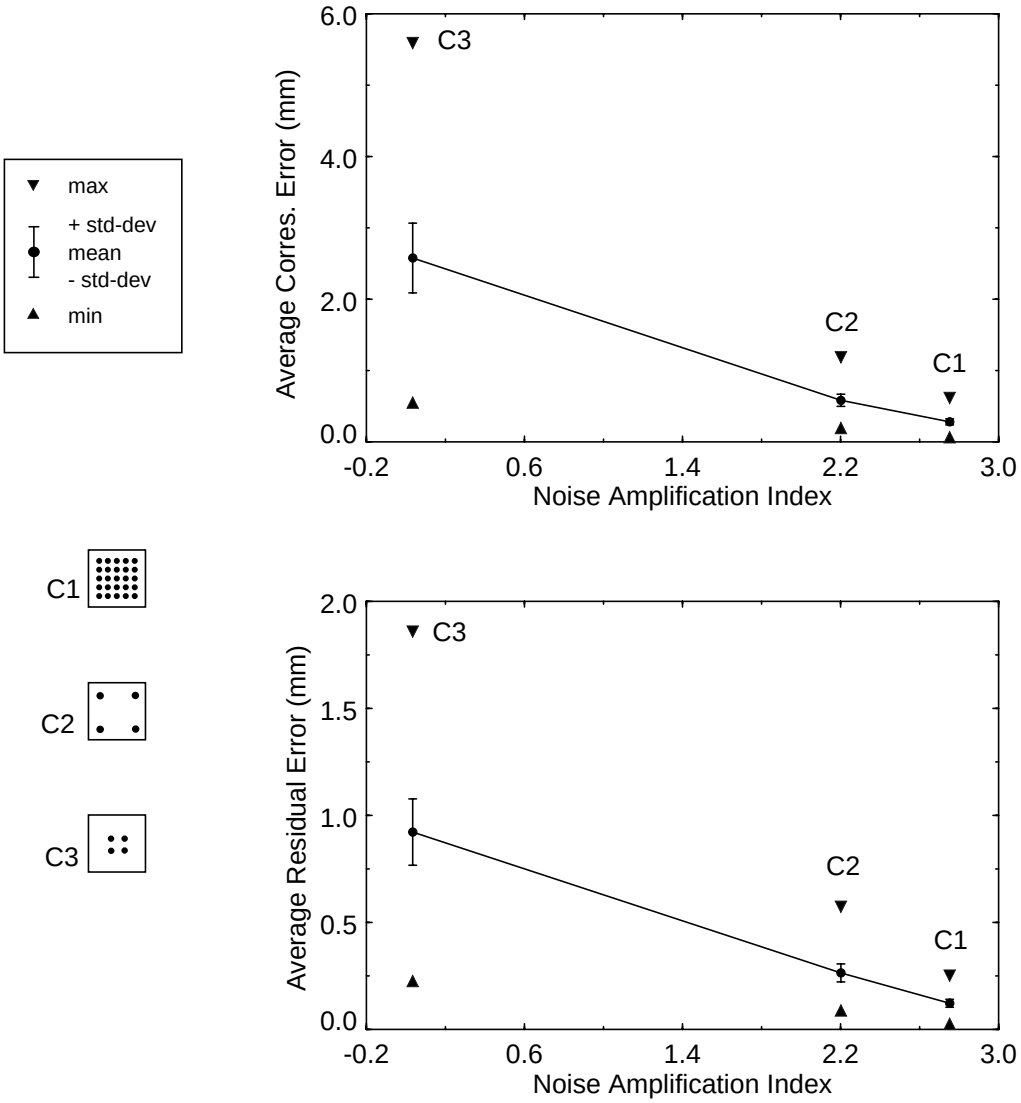


Figure 3-17: Results of cube registration experiment #1: Average correspondence error and average residual error plotted versus the NAI for three Data configurations on a cube.

Table 3-10: Constraint analysis results for the slotted cylinder Data configuration S1.

Eigenvalue	Index	Eigenvectors					
		Translation Elements			Rotation Elements		
	1	0.00	0.00	1.00	0.00	0.00	0.00
	2	0.00	-0.01	0.00	-0.02	0.00	1.00
	3	0.29	-0.27	0.00	0.56	0.73	0.01
	4	0.64	-0.33	0.00	0.31	-0.62	0.00
	5	0.61	-0.08	0.00	-0.74	0.29	-0.01
	6	0.38	0.90	0.00	0.22	0.02	0.02

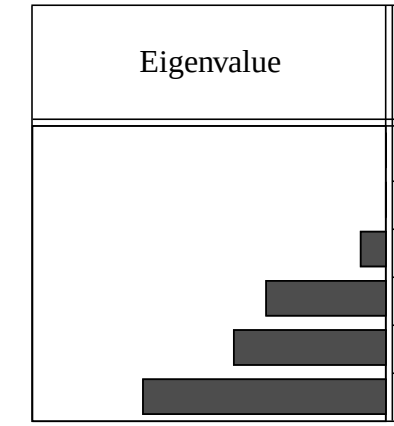
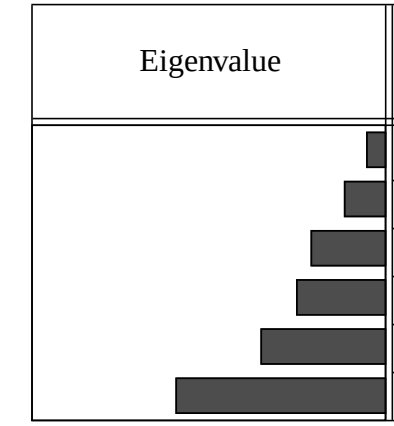


Table 3-11: Constraint analysis results for the slotted cylinder Data configuration S2.

Eigenvalue	Index	Eigenvectors					
		Translation Elements			Rotation Elements		
	1	0.18	-0.31	-0.01	-0.02	0.04	0.93
	2	0.09	-0.12	0.35	0.76	-0.53	-0.02
	3	-0.07	0.14	0.90	-0.39	0.00	0.07
	4	0.08	-0.42	0.24	0.35	0.78	-0.18
	5	0.82	0.54	0.02	0.09	0.17	0.01
	6	0.52	-0.64	0.00	-0.38	-0.30	-0.30



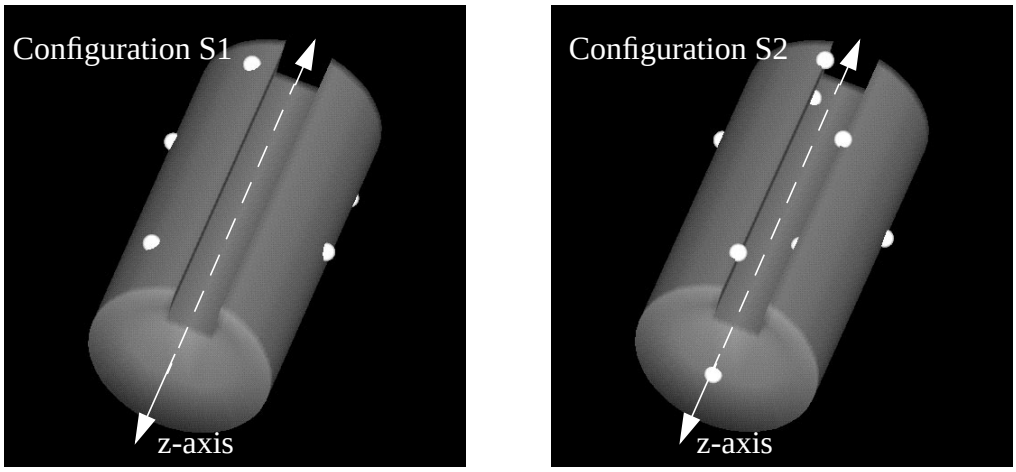


Figure 3-18: Slotted cylinder with 2 Data configurations, each with a total of ten points.

of MCE versus NAI for this experiment is shown in Figure 3-19. Due to the singularities in configuration S1, the value of the NAI is zero for this configuration. From the plot it is apparent that the mean, standard deviation, minimum and maximum values of the MCE are smaller for configuration S2 than S1, as predicted by the NAI. Since the NAI for configuration S1 is zero, the resulting MCE values are very dependent upon the particular initial poses used in the registration trials. Initial poses which displace the Data in the direction of a singularity would result in worse MCE values than initial poses which do not displace the Data in these directions. Alternate measures of registration error, the individual components of

Table 3-12: Parameters of the slotted cylinder registration experiment #1

Parameter	Value
Data source	Read from file
Data set size, N	10
τ_{imax}	2.0 mm
θ_{imax}	5 deg
Noise state	Enabled
μ_{Δ}	0.2 mm
pose_count	500
set_count	2
Minima suppression	Disabled

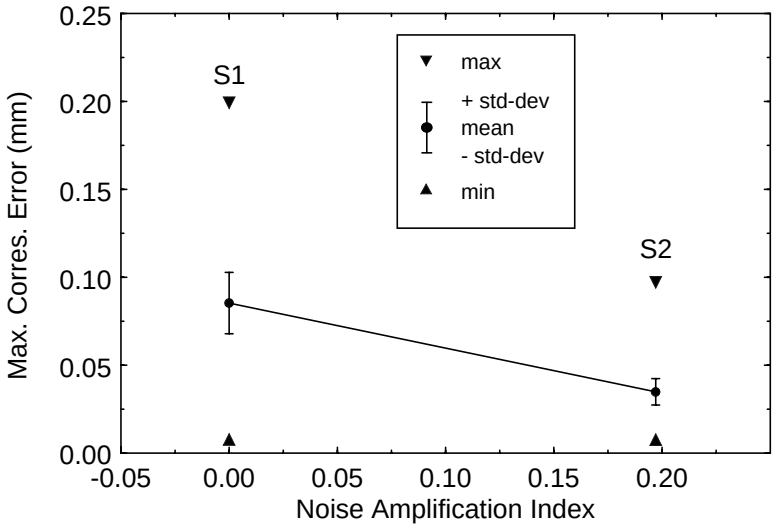


Figure 3-19: Results of the slotted cylinder registration experiment #1. Maximum correspondence error plotted versus the NAI for two Data configurations on the slotted cylinder.

the registration error transformation of Figure 2-8, are plotted in Figure 3-20. From these plots, it is apparent that the registration inaccuracies resulting from configuration S1 are due to translations and rotations about the z-axis of the cylinder as expected from the results of constraint analysis for this shape.

Femur, Pelvis and Venus Examples

Three additional demonstrations of the relation between constraint analysis and registration accuracy are presented for a human femur, a human pelvis, and a bust of the goddess Venus, all of which are shown in Figure 3-21. The Femur and Pelvis Models were constructed from CT images by extracting contours of the bones and then linking these contours into triangle mesh surfaces using techniques described in [Geiger, 1993]. The Venus Model was created from multiple range data views using a technique based on deformable surfaces [Delingette et al., 1992]. The number of vertices in each of the models is 4527 for the Femur, 2375 for the Pelvis, and 2432 for the Venus.

Registration experiments were performed for each of the Models using the parameters specified in Table 3-13. Each registration experiment was performed using 6 Data configurations containing 20 points each. Five of these configurations were randomly generated, and one was automatically synthesized to maximize the NAI using methods described in detail in Chapter 4. The maximum initial pose misalignments, 20.0 mm and 10 degrees, were based

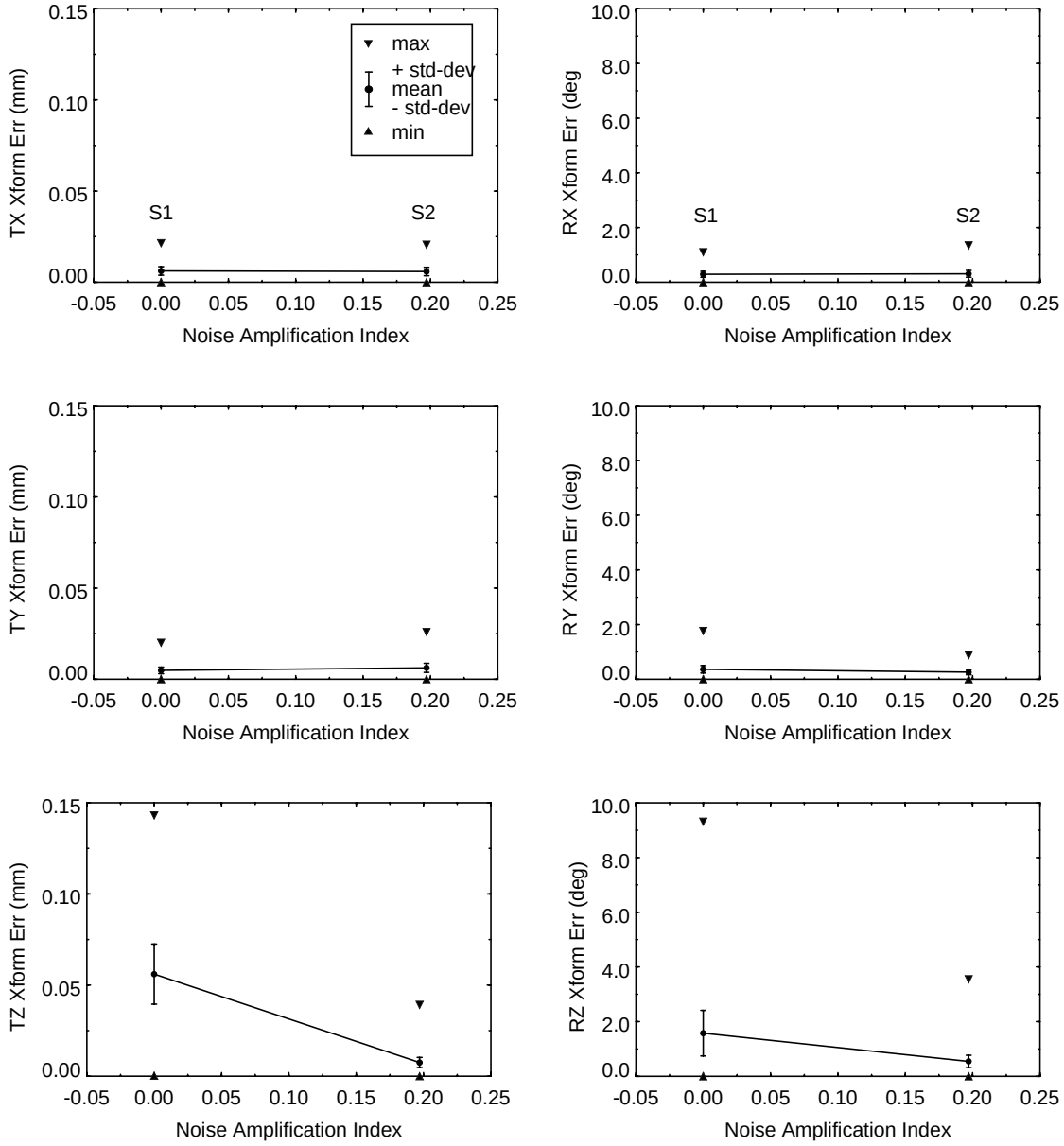
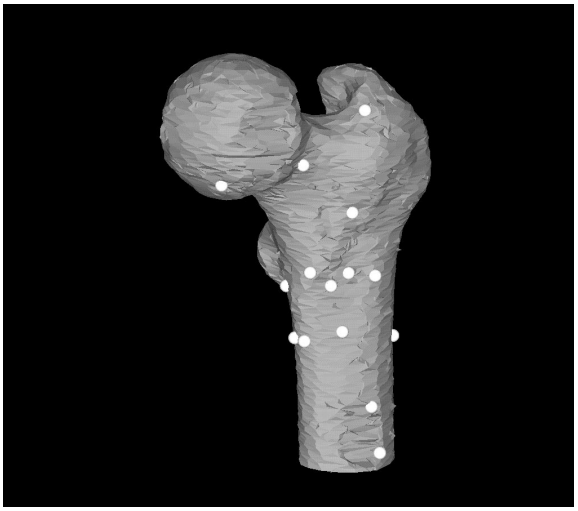
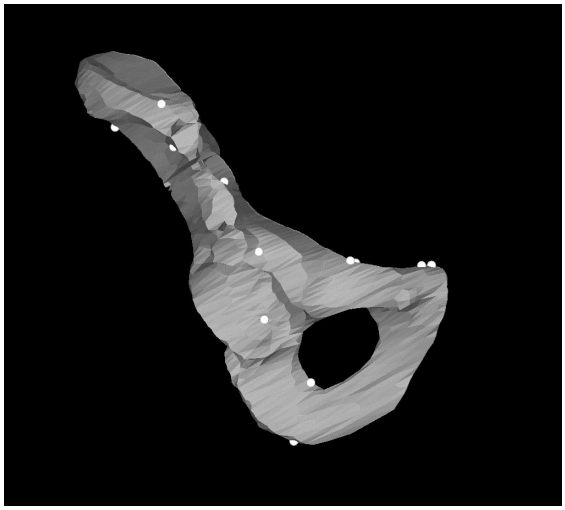


Figure 3-20: Results of slotted cylinder registration experiment #1. Individual translation and rotation error components plotted versus the NAI for two Data configurations on the slotted cylinder.

Femur:



Pelvis:



Venus:

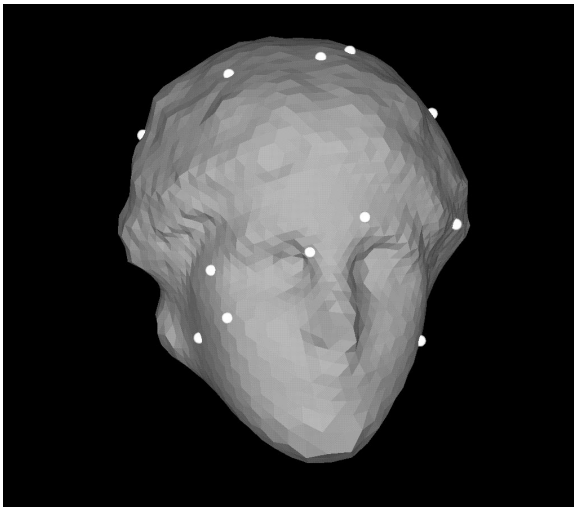


Figure 3-21: Surface Models of a human femur, a human hemi-pelvis and a bust of the goddess Venus, each superimposed with 20 randomly selected Data points.

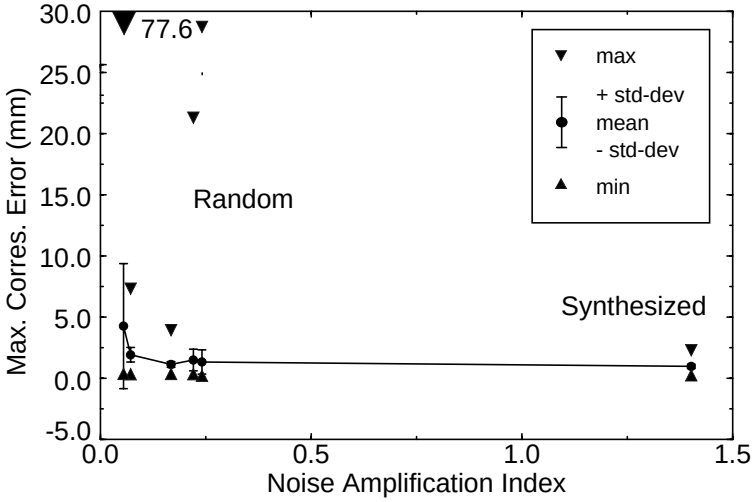
Table 3-13: Parameters for Femur, Pelvis and Venus registration experiments #1.

Parameter	Value
Data source	Random & File
Data set size, *N*	20
τ_{imax}	20.0 mm
θ_{imax}	10 deg
Noise state	Enabled
μ_{Δ}	1.0 mm
pose_count	500
set_count	5 random, 1 file
Minima suppression	Enabled
$\tau_{p\text{-}max}$	10.0 mm
$\theta_{p\text{-}max}$	8.0 deg
I_p	6 iterations

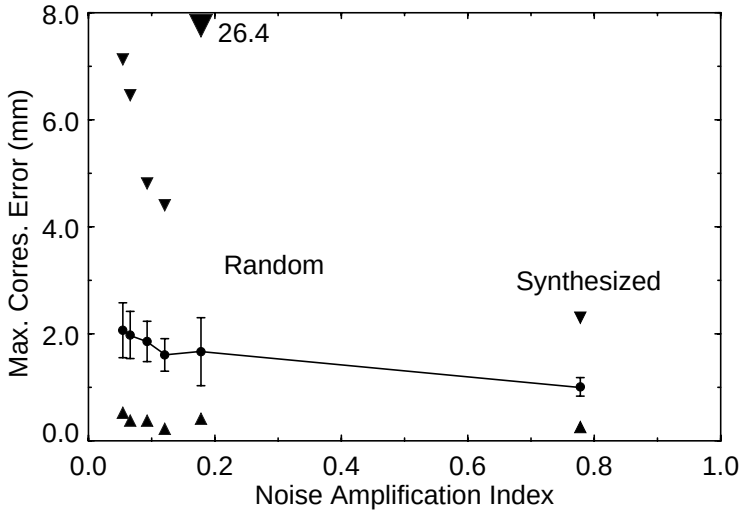
on empirically derived estimates of the accuracy of anatomical landmark-based corresponding point registration (see Figure 2-3). The expected noise magnitude of 1.0 mm is a very conservative estimate of expected measurement noise from the optical digitizing system used to collect Data in subsequent experiments. Local minima suppression was always enabled for these experiments.

The results of these experiments are reported in Figure 3-22 as graphs of NAI versus MCE. The general trend between NAI and registration error is similar to the relation observed for the cube and the slotted cylinder; smaller registration errors are generally associated with larger values of the NAI. For each Model, the 5 randomly generated Data configurations all have smaller values of the NAI than the single synthesized configuration. It is important to note that the NAI provides an estimate of *worst case* MCE value for a given configuration. For example, the minimum MCE values for the random Data configurations are smaller than the mean MCE values for the synthesized configurations. Due to variations in initial pose and added noise, it is possible for a Data configuration with a small NAI value to have a lower MCE value than a configuration with a large NAI value in a given registration trial. The utility of constraint analysis is due to the relation between NAI and the maximum and mean MCE values over multiple registration trials for a given Data configuration. These relations are well illustrated in the Pelvis graph in which mean and maximum MCE are

Femur:



Pelvis:



Venus:

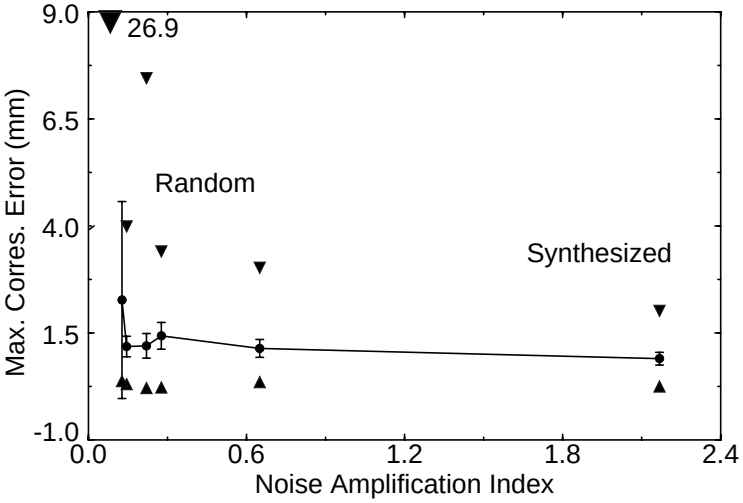


Figure 3-22: Results of Femur, Pelvis and Venus registration experiments #1. MCE plotted versus the NAI for five random Data configurations and one synthesized Data configuration. Note the different scales along the y-axes. Large triangles at the upper borders indicate data which would lie off the graph (y-value adjacent).

almost monotonically decreasing functions of NAI. Additional results which demonstrate the relation between maximum MCE and NAI are presented below.

In the experiments of Figure 3-22, several Data configurations resulted in very large maximum MCE values due to registration trials which converged to local minima in the registration pose space. This is true despite the use of local minima suppression. As suggested in Section 2.3, manual verification may be required in critical applications to identify such registration local minima. In the graphs of Figure 3-22, the maximum values of MCE corresponding to some of these local minima are not drawn to scale along the y-axes, but rather are indicated by the large downward facing triangles with adjacent numbers indicating the values. Two additional registration local minima are apparent for the Femur, both with NAI values near 0.25. It is important to note that the relation between NAI and maximum MCE does *not* hold for registration trials which have converged to local minima. Constraint analysis is formulated using a local approximation of the registration cost function which only holds in the vicinity of the registration global minimum. When registration converges to a local minimum, no relation between the NAI and registration error is expected.

The distinction between poor registration accuracy due to local minima in the registration pose space and poor registration accuracy due to poor geometric constraint is a subtle yet important difference. It is possible for a Data configuration with a large NAI value to converge to a local minimum, and for a Data configuration with a small NAI value to converge to the global minimum with poor accuracy due to poor constraint. Both are undesirable situations resulting in poor registration accuracy; however, the mechanisms which cause these results are quite different. The former situation must be addressed via methods for avoiding local minima in the registration pose space, while the latter can be addressed via intelligent selection of registration Data.

To better demonstrate the relation between MCE and NAI, an additional set of experiments was performed using the Femur, Pelvis and Venus models. For these experiments, 1000 configurations were generated randomly, and the number of registration trials per configuration was reduced to 3, as seen in Table 3-14. The results of these experiments are presented in the scatter plots of Figure 3-23 which graph the NAI versus the maximum value of MCE achieved over the 3 registration trials. To eliminate the effects of registration local minima in these graphs, all Data configurations with maximum MCE values greater than 10 mm were eliminated. The total numbers of configurations eliminated out of 1000 were 22 for the Femur, 8 for the Pelvis and 1 for the Venus. From the figure, the relation between registration error and constraint analysis is clear; the NAI places an upper bound on registration error as indicated by the hand-drawn upper bound curves in each graph. For each Model, the upper bound on error falls off sharply with increasing NAI.

Table 3-14: Parameters for Femur, Pelvis and Venus registration experiments #2.

Parameter	Value
Data source	Random
Data set size, N	20
τ_{imax}	20.0 mm
θ_{imax}	10 deg
Noise state	Enabled
μ_{Δ}	1.0 mm
pose_count	3
set_count	1000
Minima suppression	Enabled
$\tau_{p\text{-}max}$	10.0 mm
$\theta_{p\text{-}max}$	8.0 deg
I_p	6 iterations

In the Pelvis Model graph of Figure 3-23, a single outlier is apparent with an NAI value of roughly 0.8. This outlier is caused by changes to the NAI value resulting from the added sensor noise. With the addition of noise, the points within a Data configuration will be perturbed from their nominal locations. Due to this perturbation, the surface normals and radius vectors used to compute the NAI may change (see Equation 3.4). The term *ideal* NAI will be used to refer to the NAI computed from the unperturbed, nominal Data points. The term *effective* NAI will be used to refer to the NAI computed from the perturbed, noisy Data points. The effective NAI can be computed from Equation 3.4 using the Model points which are closest to the actual registration Data points at the termination of the registration process. Differences between ideal and effective NAI values result not only from sensor noise, but also from uncertainty in the Data collection process of Figure 3-1. This uncertainty may result in discrepancies between the locations of the desired points (i.e., from the Data collection plan) and locations of the points actually collected. In order to demonstrate the usefulness of the effective NAI, the results of Figure 3-23 are re-plotted in Figure 3-24 using this measure. Note that the upper bound curves are much steeper and have no outliers when the effective NAI is used. Complete discussions of Data collection uncertainty and the effective NAI appear in Chapter 4.

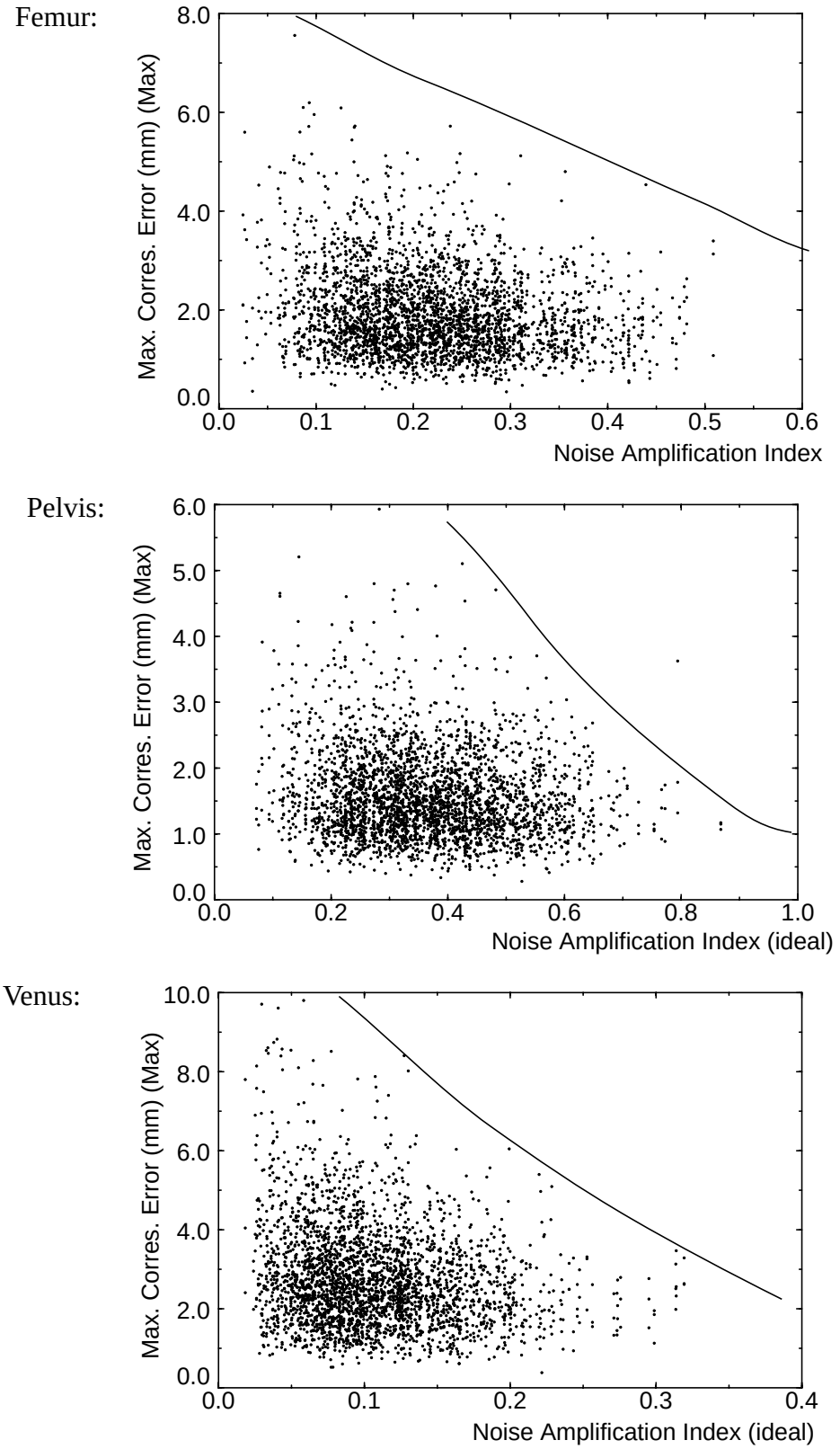
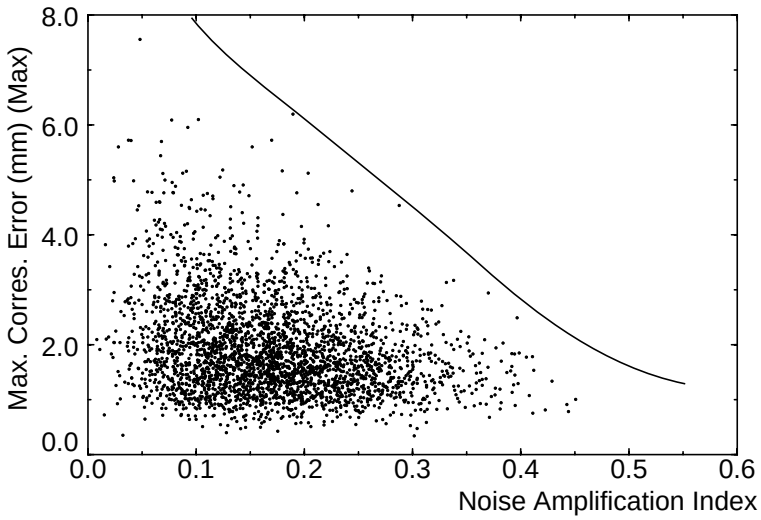
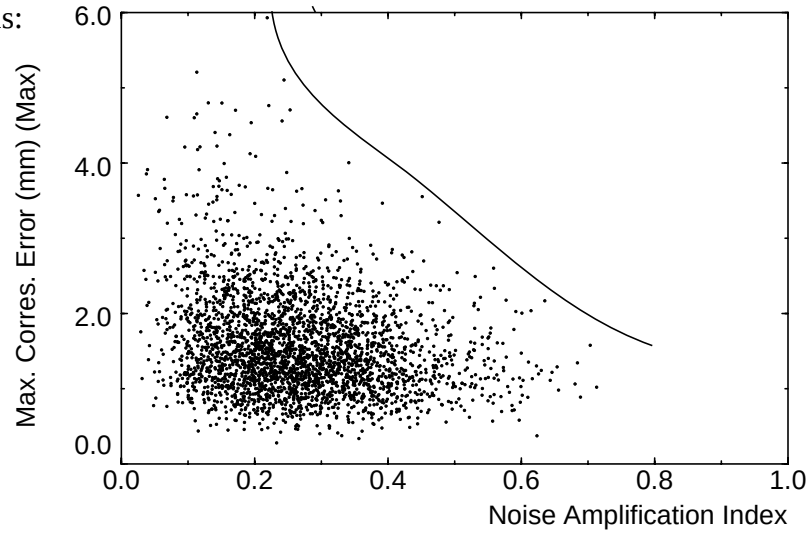


Figure 3-23: Results of Femur, Pelvis and Venus registration experiments #2. Maximum MCE plotted versus the *ideal* NAI for 1000 random Data configurations. Upper bound curves drawn by hand. Note the different scales along the x- and y-axes.

Femur:



Pelvis:



Venus:

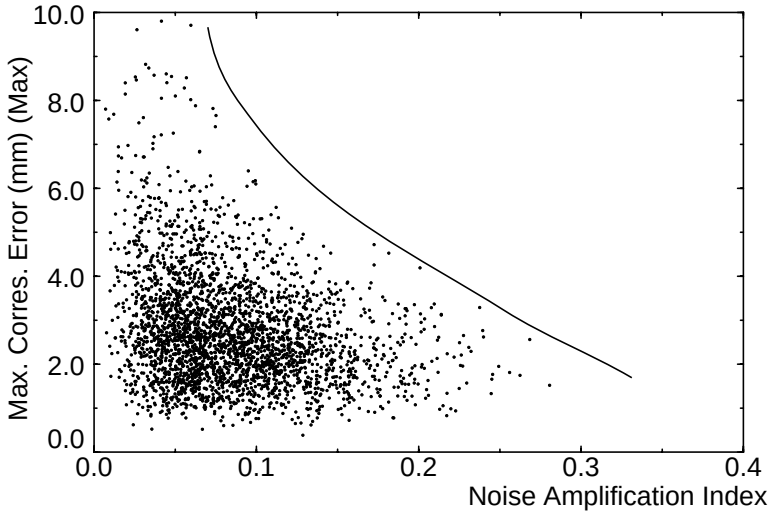


Figure 3-24: Results of Femur, Pelvis and Venus registration experiments #2. Maximum MCE plotted versus *effective* NAI for 1000 random Data configurations. Upper bound curves drawn by hand. Note the different scales along the x- and y-axes.

The results obtained in the Femur, Pelvis and Venus experiments, and in particular the relation between maximum MCE and NAI have a resemblance to results reported previously by other researchers. In the area of optimal pose selection for kinematic calibration of robots, several papers report results which are quite similar to those shown in Figures 3-23, 3-23 and 3-24 [Borm and Menq, 1991] [Nahvi and Hollerbach, 1996]. These papers demonstrate a strong similarity between errors in a robot's kinematic model, and criterion measures such as the NAI. This similarity is expected due to the fundamental resemblance between constraint analysis and robot pose selection as explained in Section 3.2. The group at Ohio State University working in the area of registration has also reported similar results [Menq et al., 1992].

3.4.4 Noise, Registration Accuracy, and Constraint Analysis

Added Data noise can influence registration accuracy and the effective NAI value described in the previous subsection. This subsection presents experimental results which demonstrate the effects of Data noise upon registration accuracy for a simple Model, the cube. Subsequent chapters present similar results for more complex Models.

Table 3-15 contains the parameters of the experiments performed. Each of the three cube Data configurations, C1, C2 and C3 described earlier, was registered to the cube Model

Table 3-15: Parameters: cube experiment #2: Noise versus registration accuracy

Parameter	Value
Data source	File
τ_{imax}	5.0 mm
θ_{imax}	10 deg
Noise state	Enabled
μ_{Δ}	0.2-2.2 mm by 0.2 mm
pose_count	500
set_count	3
Minima suppression	Enabled
$\tau_{p\text{-}max}$	2.5 mm
$\theta_{p\text{-}max}$	5.0 deg
I_p	6 iterations

using different noise magnitudes. A total of 11 registration experiments were performed with each Data configuration using expected noise magnitudes ranging from 0.2 mm to 2.2 mm in 0.2 mm increments. The length of a cube edge was 50 mm, so the expected noise magnitudes ranged from 0.4% to 4.4% of the size of the cube, with maximum noise magnitudes roughly a factor of 3 larger.

The results of this set of experiments are presented in Figure 3-25. Each column of graphs in the figure corresponds to a particular Data configuration, C1, C2, or C3. Each row of graphs corresponds to an error measure: maximum correspondence error, MCE; translation error norm, $\|E_t\|$; and rotation error norm, $\|E_\omega\|$. X-axis scales are the same for all graphs, while y-axis scales are the same within each row of graphs. As expected, registration errors increase as a function of increasing Data noise. There is a strong relation between the slopes of the error curves and the NAI value for a given configuration; larger NAI values correspond to smaller slopes, and thus lower sensitivity of a configuration to noise. For example, the mean MCE curve is roughly a factor of 6 steeper for C3 than for C1, and a factor of 3 steeper for C3 than for C2. Most of the difference in sensitivity between the three configurations can be attributed to rotation errors as seen by comparing mean error slopes in the bottom two rows of graphs. Also note that the slopes of the mean translation error norm curves are roughly the same for C2 and C3, while the slopes of the mean rotation error norm curves for these configurations differ by a factor of roughly 3.5. Both of these observations are expected based upon the constraint analysis results from Tables 3-6, 3-7 and 3-8.

These experiments demonstrate a strong relation between the NAI and sensitivity of registration accuracy to Data noise. This relation was expected based upon the proof in [Nahvi and Hollerbach, 1996] (see Section 3.3.2) which shows that the noise amplification index (NAI) provides an upper bound on the amplification of unmodeled errors and noise (i.e., discrete point Data measurement noise and errors in the Model) to the estimated parameters (i.e., registration transformation parameters). In Section 4.4.4, the results of similar experiments for the Femur, Venus and Pelvis objects are presented.

3.5 Discussion

This chapter demonstrates that the criterion measure provided by constraint analysis is a good predictor of registration accuracy. Similar observations have been made by other researchers; however, the presentation in this chapter is significant for several reasons.

1. It experimentally demonstrates the relationship between sensor noise, constraint analysis, and registration accuracy.
2. It identifies and analyzes the dependence of constraint analysis upon object scale, and proposes a normalization method which reduces this effect.

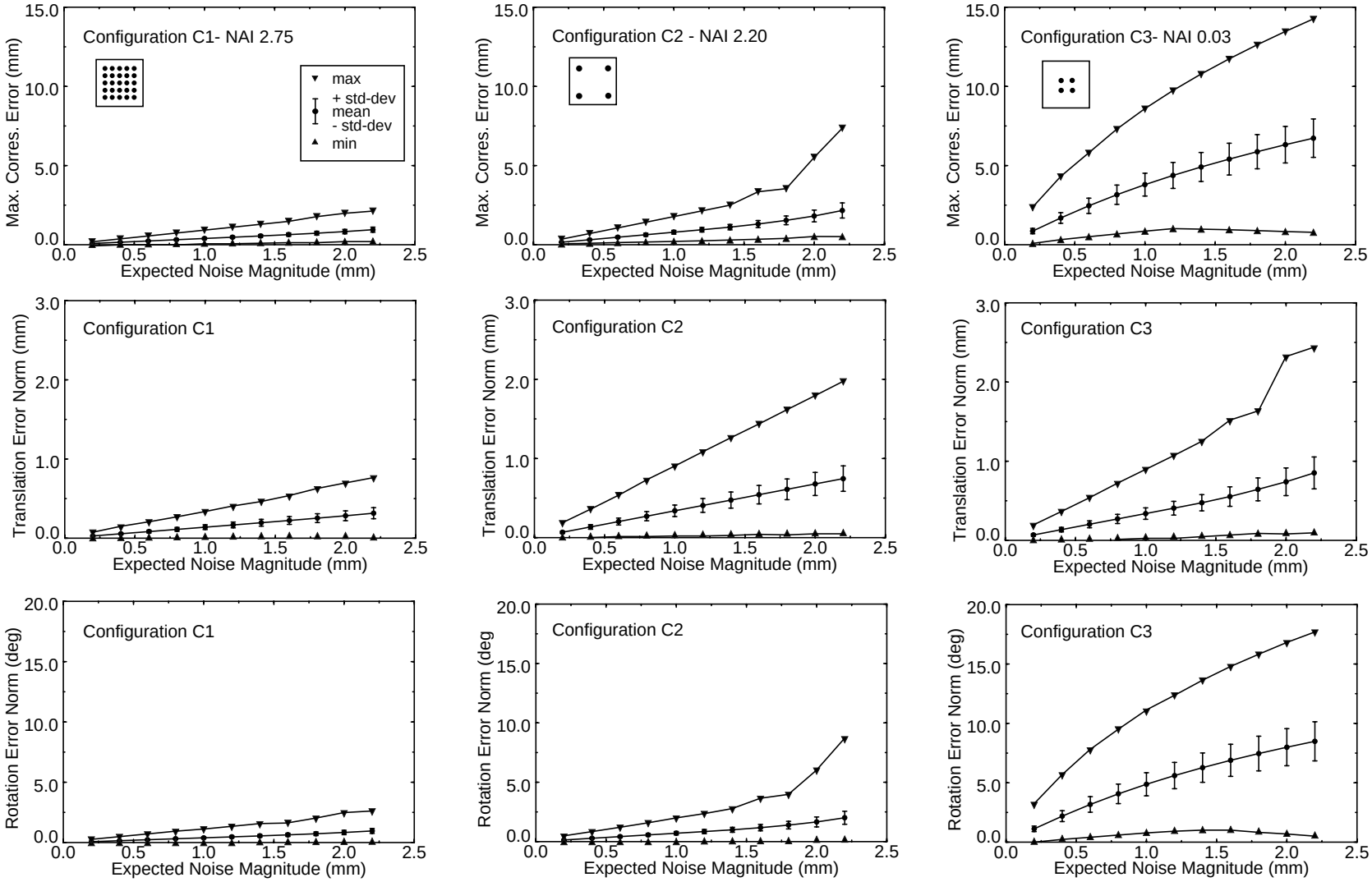


Figure 3-25: Registration errors (MCE, translation norm, rotation norm) as a function of expected Data noise magnitude for the three cube configurations C1, C2 and C3. Top row: MCE, middle row translation error, bottom row: rotation error.

3. It identifies and analyzes the dependence of constraint analysis upon coordinate system origin, and proposes the value specified in Equation 3.42 as the location of the coordinate system origin which maximizes constraint analysis sensitivity.
4. It proposes and justifies the selection of the noise amplification index as a suitable criterion measure for constraint analysis.

Constraint analysis plays two important roles in the object localization framework of Figure 3-1. For Intelligent Data Selection, constraint analysis provides the criterion measure to be maximized by constraint synthesis in order to generate Data configurations which are near-optimal in terms of expected registration accuracy. In this role, it is important that constraint analysis be computed very quickly, since it may be necessary to compute the noise amplification index on the order of 10^6 times to compute a single near-optimal Data configuration. Constraint synthesis and Intelligent Data Selection are discussed in detail in Chapter 4.

In its second role in the object localization framework, constraint analysis provides a measure which is useful for estimating online registration error, a topic explored in Chapter 5. It is important to note that constraint analysis does not provide measures of *absolute* expected registration accuracy, since factors which are unmodeled by constraint analysis such as noise in the Data or Model will affect absolute accuracy. Because of this limitation, constraint analysis alone can not be used to determine the *quantity* of Data required to satisfy a given registration accuracy requirement. Methods for addressing this problem are discussed in Chapter 5.

Development of the infrastructure necessary to perform the hundreds of thousands of registration trials required to generate the results reported in this chapter was a non-trivial endeavor. For example, the results of Figure 3-24 alone required roughly 50,000 registration trials. Software was written to automate much of the experimental process. Experiments were often distributed over 30 or more computer workstations, and systems were developed to automatically invoke and record the experiments. Analysis of the data was also semi-automated, and software was written to automatically convert the voluminous output of a single registration experiment to the various graphs presented in this chapter. Performing these experiments, as well as the ones reported in Chapters 4 and 5, would have been impossible without distributing the computation over multiple workstations, and without the automated infrastructure for collecting and analyzing results.

Chapter 4

Constraint Synthesis

The constraint analysis methods described in the previous chapter provide estimates of the relative registration accuracy among multiple, fixed-size Data configurations. The constraint synthesis methods described in this chapter use the output of constraint analysis to automatically generate near-optimal Data configurations with respect to the noise amplification index (NAI). Constraint synthesis is shown to be a high-dimensionality optimization problem in which the objective is to maximize the NAI, and the independent variables are the locations of points included in a Data configuration. The relation of constraint synthesis to the other elements of the object localization framework is shown in Figure 4-1.

This chapter is organized as follows. Section 4.1 defines the constraint synthesis problem. Section 4.2 describes four search algorithms which have been used to perform constraint synthesis: steepest ascent hillclimbing (SAH), next ascent hillclimbing (NAH), Population-Based Incremental Learning (PBIL) and a hybrid PBIL - hillclimbing approach. Section 4.3 introduces the problem of Data collection uncertainty. Due to uncertainty regarding the pose of the registration object, it is impossible for the Data collection process of Figure 4-1 to precisely collect the exact Data specified by the Data collection plan. This section examines the implications of this uncertainty. Section 4.4 presents experimental results which demonstrate the effectiveness of constraint synthesis and the robustness of the resulting Data configurations. Section 4.4.1 describes the framework used to evaluate constraint synthesis. Section 4.4.2 compares the performance of the constraint synthesis search algorithms. Section 4.4.3 demonstrates the effects of Data collection uncertainty upon the NAIs of synthesized Data configurations. Section 4.4.4 examines the effects of sensor noise and Data collection uncertainty upon registration accuracy based upon synthesized Data configurations. The chapter concludes with a discussion in Section 4.5.

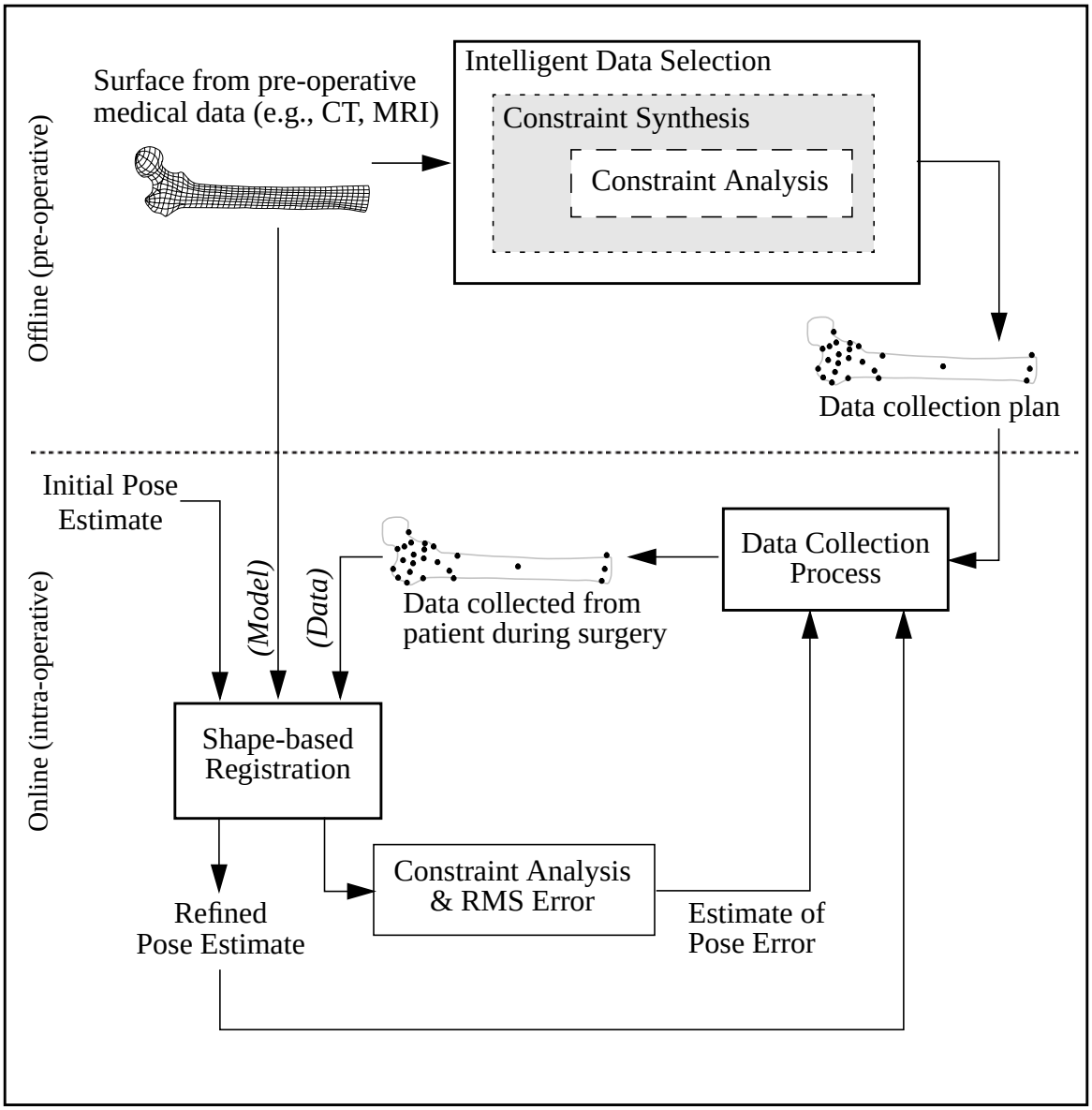


Figure 4-1: The role of constraint synthesis (shaded) in Intelligent Data Selection and the object localization framework.

4.1 Problem Description

The constraint synthesis problem can be stated as follows:

> Select N discrete points from a set, *V,* and place these points in the Data configuration set, *P,* of Equation 3.9 such that the noise amplification index of Equation 3.19 is maximized. The resulting optimal set, *P,* is referred to as the Data collection plan (DCP).

This is a combinatorial search problem, and for all but artificially small problems the solution space is much too large to search exhaustively. For the experimental results reported in this dissertation, the set, *V,* is composed of the vertices of triangle mesh registration Models; however, any sufficiently dense tessellation of a surface can be used to derive *V.* The number of points in a Data configuration, N, remains fixed within a given constraint synthesis trial. The problem of finding Data configurations of minimum size is addressed in Chapter 5.

In general, it may be desired to constrain points within a DCP to lie within specified regions of a Model. This is useful, for example, if surgical access is limited to certain regions of a patient's anatomy, thus preventing the collection of Data outside of those regions. Such access constraints can easily be incorporated into constraint synthesis by including in the set *V,* only points within the accessible regions.

In DCPs generated by constraint synthesis, it is possible to allow a single vertex to appear more than once. Mathematically, this choice is justified since repeated points in the set *P* can increase the value of the NAI. In practice, each time a Data point is collected it will provide new, and slightly different information for the shape-based registration process.

4.2 Search Algorithms for Constraint Synthesis

This section presents four search algorithms for solving the constraint synthesis problem: two hillclimbing methods, an approach related to genetic algorithms known as Population-Based Incremental Learning (PBIL) and a hybrid PBIL / hillclimbing approach.

4.2.1 Steepest Ascent Hillclimbing

The first type of hillclimbing is steepest ascent hillclimbing (SAH). Initially, N vertices are randomly chosen from the set of possible vertices, *V,* and placed into the set *P.* Let NAI**(***P***)** represent the value of the noise amplification index computed using the points in the set *P* **(**see Equation 3.19). Select a vertex, *v,* from *P.* In turn, substitute *v* with every vertex in *V.* Repeat this procedure for each *v* in *P.* After all substitutions have been attempted, replace the vertex in *P* with the single substitution which leads to the largest increase in NAI(*P***)**. Iterate this process, and continue until no substitutions which improve the result are found.

Steepest ascent hillclimbing is a "greedy" method which, on each iteration, performs the substitution which results in the maximal change in NAI(*P***)**. Each iteration requires N(M-1) evaluations of Equations 3.4, 3.8, 3.9, 3.10 and 3.19, where M is the number of vertices in the set of possible vertices, *V,* and N is the number of vertices in the set of selected vertices, *P.*[1] The total number of iterations required for convergence depends upon the size of the problem, but is typically less than 100 for the experiments performed in this dissertation.

1. Evaluation time for these equations is approximately 1 ms on an SGI R4400 200 MHz processor.

4.2.2 Next Ascent Hillclimbing

The second type of hillclimbing is next ascent hillclimbing (NAH). This method is similar to steepest ascent hillclimbing, except that the set P is immediately changed when a vertex substitution is found which increases NAI(P). This is in contrast to the previous method in which the chosen substitution is the one which maximizes NAI(P) over all possible substitutions. In NAH, the sequence of trial vertex substitutions within an iteration is randomized.

Next ascent hillclimbing is less greedy than the steepest ascent version. It usually requires more iterations than the previous method, but there are fewer NAI evaluations per iteration. The number of attempted substitutions is typically small during the initial iterations, and increases during the later iterations when there are fewer possible substitutions which will increase the value of NAI(P). Although rarely reached in practice, the maximum number of evaluations within an iteration is the same as the number required for each iteration of steepest ascent hillclimbing, N(M-1). The total number of iterations depends upon the size of the problem, but is typically less than 700 for the experiments performed in this dissertation.

4.2.3 Population-Based Incremental Learning

As demonstrated below in Section 4.4, the above hillclimbing methods are susceptible to local minima in the search space. Genetic algorithms (GAs) are biologically motivated adaptive systems based upon principles of natural selection and genetic recombination which attempt to avoid such local minima. Good overviews of GAs can be found in [Goldberg, 1989] and [DeJong, 1975].

Recently, a simplified statistical model of the GA has been introduced in [Baluja and Caruana, 1995] termed Population-Based Incremental Learning (PBIL). This method has been compared to standard GAs on a variety of benchmarks with promising results [Baluja, 1995]. Like the standard GA, the version of PBIL presented below operates on solutions encoded as binary vectors. A brief introduction to the PBIL method is given below, and the algorithm is shown in Figure 4-2 (description and figure with permission from [Baluja, 1995]):

> PBIL is a combination of evolutionary optimization and hillclimbing. The object of the algorithm is to create a real valued probability vector which, when sampled, reveals high quality solution vectors with high probability. For example, if a good solution to a problem can be encoded as a string of alternating 0's and 1's, a suitable final probability vector would be 0.01, 0.99, 0.01, 0.99, etc.
>
> Initially, the values of the probability vector are set to 0.5. Sampling from this vector yields random solution vectors because the probability of generating a 1 or 0 is equal. As search progresses, the values in the probability vector gradually shift to represent high evaluation solution vectors. This is accomplished as follows: A number of solution vectors are generated based upon the probabilities specified in the probability vector. The probability vector is pushed towards the generated solution vector(s) with the highest evaluation. The distance the probability vector is pushed depends upon the learning rate parameter. After the probability vector is updated, a new set of solution vectors is produced by sampling from the updated probability vector, and the cycle is continued. As the search progresses, entries in the probability vector

move away from their initial settings of 0.5 towards either 0.0 or 1.0. The probability vector can be viewed as a prototype vector for generating solution vectors which have high evaluations with respect to the available knowledge of the search space.

The manner in which the updates to the probability vector occur is similar to the weight update rule in supervised competitive learning networks, or the update rules used in Learning Vector Quantization (LVQ) [Hertz et al., 1991]. Many of the heuristics used to make learning more effective in supervised competitive learning networks (or LVQ), or to increase the speed of learning, can be used with the PBIL algorithm. This relationship is discussed in greater detail in [Baluja and Caruana, 1995].

```
****** Initialize Probability Vector *****
for i :=1 to LENGTH do P[i] = 0.5;

while (NOT termination condition)
    ***** Generate Samples *****
    for i :=1 to SAMPLES do
      sample_vectors[i]:=generate_sample_vector_with_probabilities (P);
      evaluations[i] :=Evaluate_Solution (sample[i]);
      best_vector := best_evaluation (sample_vectors, evaluations);

    ***** Update Probability Vector Towards Best Solution *****
    for i :=1 to LENGTH do
      P[i] := P[i] * (1.0 - LR) + best_vector[i] * (LR);

    ***** Mutate Probability Vector *****
    for i :=1 to LENGTH do
      if (random (0,1) < MUT_PROBABILITY) then
            if (random (0,1) > 0.5) then mutate_direction := 1
            else mutate_direction := 0;
            P[i]:=P[i]*(1.0-MUT_SHIFT)+ mutate_direction*(MUT_SHIFT);
```

USER DEFINED CONSTANTS (Values Used in this Study):
SAMPLES: vectors generated before update of the probability vector (100).
LR: the learning rate, how fast to exploit the search performed (0.1).
MUT_PROBABILITY: probability for a mutation in each position (0.02).
MUT_SHIFT: amount a mutation alters the value in the bit position (0.05).
LENGTH: length of encoded solution (problem dependent).

Figure 4-2: Single vector PBIL algorithm for binary encoded solution strings (with permission from [Baluja, 1995]).

The application of PBIL to the constraint synthesis problem was performed in collaboration with Shumeet Baluja, and additional details of this work can be found in [Baluja and Simon, 1996]. In particular, that document describes issues related to the problem encoding and search space representations which were used to map the constraint synthesis problem into the PBIL framework.

For the purposes of this dissertation, PBIL can be thought of as a black-box as outlined in Figure 4-3. The inputs to PBIL are: the surface Model; the set of allowable configuration vertices, *V*; the number of discrete points in the configuration to be generated by PBIL; the

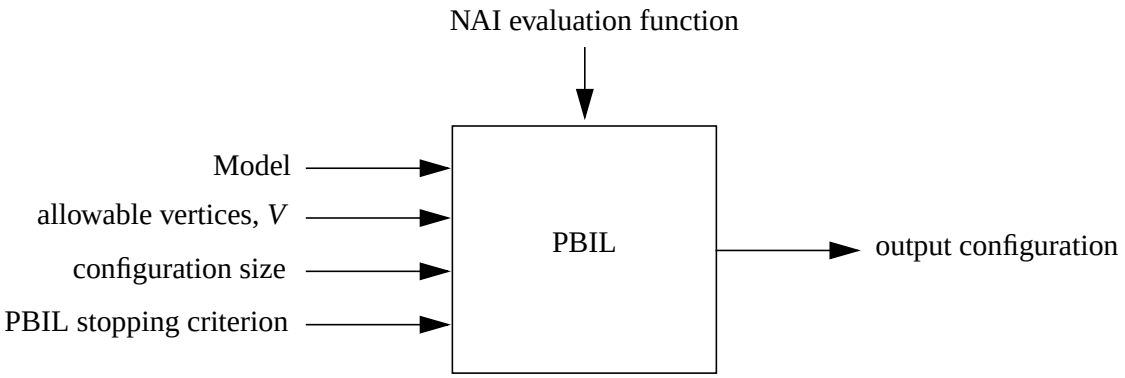


Figure 4-3: Population-Based Incremental Learning (PBIL) as a black-box.

stopping criterion; and the function to be optimized (in this case the NAI). The output of PBIL is the particular configuration which maximizes the NAI among all configurations evaluated by PBIL within a given trial.

4.2.4 Hybrid PBIL / Hillclimbing Approach

As suggested above, PBIL is better than hillclimbing methods at guiding the search towards the region of global optimality while avoiding sub-optimal local minima. However, additional performance improvements can often be realized by initializing a hillclimbing search using the output of PBIL as a starting point. By combining these two approaches, it is possible to take advantage of the strengths of each method, as illustrated in Figure 4-4. In the top graph, hillclimbing begins at a random configuration indicated by the open circle, and converges to a locally maximal configuration indicated by the closed circle. In this case, the resulting configuration is clearly not globally optimal. In the lower graph, the open circle represents the configuration resulting from a trial of PBIL, and the closed circle is the result of hillclimbing from the starting configuration provided by PBIL.

Due to the size and complexity of the search problem presented by constraint synthesis, even the hybrid PBIL hillclimbing method is not guaranteed to converge to the globally optimal solution. In an attempt to improve the NAI value of the selected configuration, it is possible to run the hybrid search method multiple times, as illustrated in Figure 4-5. The output of each trial of PBIL (indicated by the open circles) provides a starting point for the hillclimber which converges to a locally maximal configuration (indicated by the closed circles). A single optimal configuration can then be selected as the one with the overall maximal NAI value. However, as described in Section 4.3 and demonstrated in Section 4.4.3, maximizing the NAI value may not be the only criterion for selecting the “best” configuration. Due to uncertainty in the Data collection process, it is also important to assess the *stability* of the configuration with respect to the NAI.

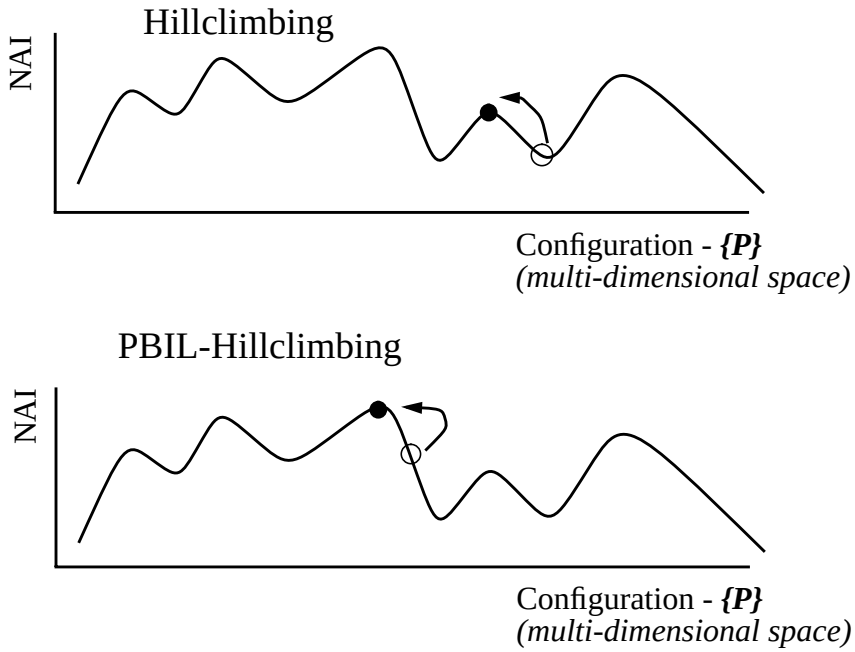


Figure 4-4: Schematics of the energy landscape for the constraint synthesis problem: hillclimbing alone versus a PBIL-hillclimbing hybrid. Open (closed) circles represent the starting (ending) points of the hillclimbing process. In the hillclimbing method, the starting point is randomly selected. In the hybrid method, hillclimbing begins at the best configuration found by PBIL.

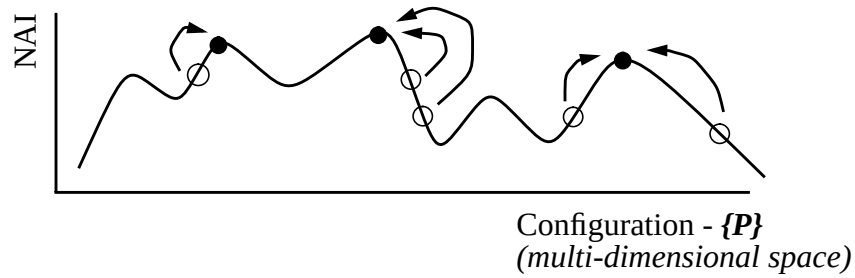


Figure 4-5: Multiple PBIL trials followed by hillclimbing. The open circles represent the outputs of multiple PBIL trials and the starting points for hillclimbing. The closed circles represent the outputs of the hillclimbing process.

Experimental results which demonstrate the performance of each of the constraint synthesis solution methods are presented in Section 4.4.2. A hybrid PBIL/NAH method is shown to be superior to the other methods in terms of selecting configurations which maximize the NAI.

4.3 Data Collection Uncertainty

As seen in Figure 4-1, the output of constraint synthesis is a *Data collection plan* (DCP). Unfortunately, it is impossible for a person (i.e., the Data collector) to physically collect the exact points specified by the DCP. (If this were possible, it would be unnecessary to perform registration, since the pose of the object would already be known.) This situation is illustrated in Figure 4-6, and is referred to as the *Data collection uncertainty problem*. In general, the distance between a desired and collected point depends upon several factors including:

- The availability of recognizable features on the object in the vicinity of the desired point which the Data collector can use as landmarks.
- The availability of feedback to the Data collector based upon the current pose estimate which indicates the position of the Data collection sensor relative to the desired point.
- The visibility of the region in which the desired point is located (during surgery, some regions may be obscured by soft-tissue, fluids, etc.)

As mentioned in Section 3.4.3, there are two variants of the NAI which can be computed. The *ideal* NAI is computed using the Data points in the DCP. These Data are uncorrupted by sensor noise and Data collection uncertainty. The *effective* NAI, on the other hand, uses the Data which have actually been collected and registered to the Model. The effective NAI is computed from the Model points which are closest to the collected Data points at the termination of the registration process. Therefore, the effective NAI is a better measure than the ideal NAI of the geometric constraint between a Model and the actual registration Data.

Due to Data collection uncertainty, the effective NAI value may be smaller than the ideal NAI value. The ratio of effective to ideal NAI is a measure of NAI *stability*, and is a function

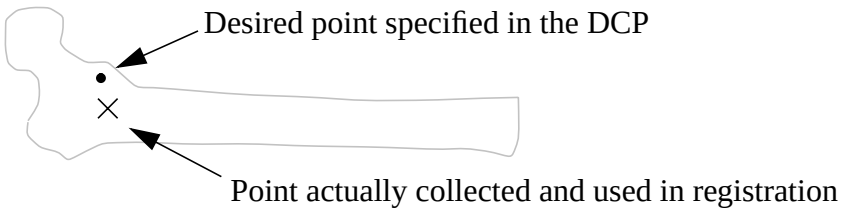


Figure 4-6: The Data collection uncertainty problem.

of several factors including the geometry of the surface, the number of Data points, the magnitude of the Data collection uncertainty, and the particular DCP. As demonstrated in Section 4.4.3, some Data configurations have greater NAI stability than others. Ideally, constraint synthesis would be formulated to maximize not only the NAI, but also Data configuration stability. Unfortunately, there are problems with such a formulation as discussed below.

A possible model for Data collection uncertainty is illustrated in Figure 4-7. For each desired point in the DCP, there is an associated region of collection uncertainty within which there is a likelihood that the actual Data point will be collected. This likelihood can be modeled as a function of the Euclidean distance between the desired point and all *connected* points which lie within the uncertainty region.[1] In the experiments reported below, the uncertainty region is defined by the intersection of the surface and a sphere of fixed radius centered at the desired point. All points within an uncertainty region have equal likelihood of being selected.

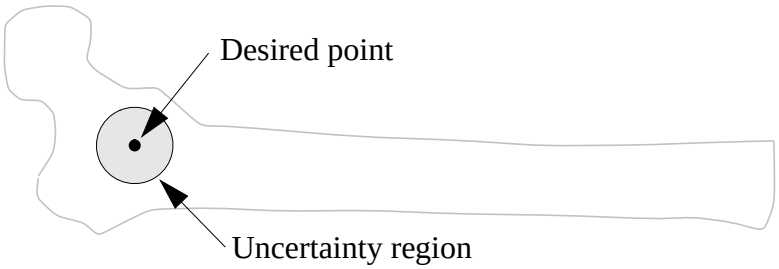


Figure 4-7: Data collection uncertainty for a single point. Any point within the uncertainty region has an equal likelihood of being collected.

Assuming that Data collection uncertainty information is available for each point (i.e.,vertex) on a Model, it is possible to re-state the constraint synthesis problem as follows:

> Select N discrete points with known Data collection uncertainties from a set, *V*, and place these points in the Data configuration set, *P*, of Equation 3.9 such that the *minimum value* of the noise amplification index, *when evaluated over all combinations of points within the associated regions of uncertainty*, is maximized.

The modified constraint synthesis problem can be explained with reference to Figure 4-8. In this formulation, each evaluation of a candidate set, *P*, requires multiple computations of the

1. Connected points are those which are topologically connected to the desired point via surface patches which lie completely within the uncertainty region.

NAI to determine the minimum value over the uncertainty regions. This minimum NAI value can then be used as the criterion measure for the candidate set, *P*. Unfortunately, this evaluation has exponential complexity. Assuming N Data configuration points, and an average of K points within each uncertainty region (i.e, K triangle mesh vertices), the number of NAI evaluations required for a single configuration evaluation would be K^N. The value of K depends upon the size of the uncertainty region and the resolution of the triangle mesh Model. However, even for relatively conservative values (N = 20, K = 5), the number of evaluations would be prohibitive ($9.5x10^{13}$).

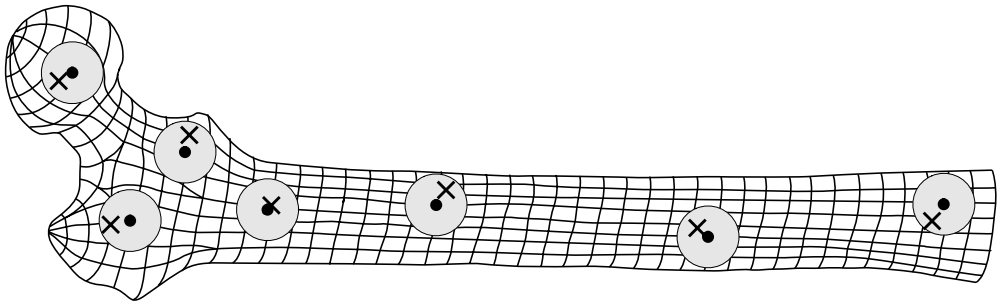


Figure 4-8: The modified constraint synthesis problem: NAI computation must be performed for all combinations of vertices within the regions of uncertainty. Small circles represent the evaluation points, large circles represent the uncertainty regions, and crosses represent the points which minimize the NAI.

It may be possible to reduce the total number of NAI evaluations by picking representative points within each uncertainty region based on similarity of surface orientation. For example, uncertainty regions which are relatively planar could be represented by a single point, while regions with larger variation in surface orientation might require 2 or 3 points to represent. Unfortunately, this still leads to unacceptable complexity. For example, for a problem in which N = 20, and for which 10 of these points can be represented by a single point (i.e., K=1), 5 can be represented by two points (i.e., K=2) and 5 can be represented by three points (i.e., K=3), the total number of evaluations is $(10^1)(5^2)(5^3) = 31250$. As demonstrated below, current run times for solving the conventional constraint synthesis problem vary from roughly 5 to 200 minutes. Multiplying even the shorter of these run times by the factor above would lead to unacceptable run times (110 days).

Due to the unacceptable complexity of the above approach, an alternative solution has been implemented which results in improved configuration stability for certain Models. From Equation 3.4 it can be seen that Data collection uncertainty will affect the NAI via a discrepancy between the *desired* and *collected* position and normal direction associated with a given Data point. It has been empirically shown that the NAI is much more sensitive to variation in the surface normal than to the position of the point. Therefore, an alternate strategy

for reducing the effect of Data collection uncertainty is to eliminate from the constraint synthesis process local patches of a Model which have high surface curvature. The resulting synthesized Data configurations will therefore consist only of points in regions of low curvature. Thus, the effects of Data collection uncertainty upon the stability of the NAI will be reduced since surface orientations in the vicinity of each desired Data point will be similar. An algorithm for eliminating regions of high curvature from a Model is presented below.

High-Curvature Point Elimination Algorithm

- For each vertex, *v*, in the Model:
    - For every vertex directly connected to *v* which lies within a sphere of radius *r*, group the associated surface normals into a set *L*. (Connected vertices are those which are topologically connected to *v* via only edges of the triangle mesh which lie completely within the sphere.)
    - Perform a pair-wise comparison between all surface normals in *L*, and determine if the angle between any two surface normals in the set is greater than a threshold, δ. If so, then do not include the vertex *v* in the filtered version of the Model.

This process eliminates vertices which are in regions of high curvature. In the experiments performed in this dissertation, typical values of δ and *r* are 90 degrees and 10 mm, respectively. As demonstrated below in Section 4.4.3, high curvature elimination before constraint synthesis can be effective at improving the stability of synthesized Data configurations for Models in which there is substantial local variation in surface orientation (e.g., the Femur Model).

4.4 Experimental Results

This section focuses on the experimental validation of the algorithms and methods described in this chapter. All experimental registration results reported in this Chapter use simulated Data (i.e., Data which are not physically collected from an object, but generated by adding noise and random pose transformations to nominal Data sets, as described in Section 3.4). In Chapter 5, results are reported for similar experiments which use physically collected Data.

4.4.1 Framework for Constraint Synthesis Experiments

In order to demonstrate the efficacy of constraint synthesis, registration experiments using the methods described in Section 3.4.2 were performed with three types of Data: random, synthesized and hand-picked. The generation of random Data sets is described in Section 3.4.2. The generation of synthesized Data sets was performed using various combinations of the constraint synthesis search algorithms described in Section 4.2. Hand-picked Data sets were manually selected by humans.

In the registration experiments presented in Section 3.4, a method is described for simulating the effects of sensor noise in the Data collection process. The same method is used in the experiments presented below. In addition, a second "noise" component is used to model the effects of Data collection uncertainty described in Section 4.3. Data collection uncertainty noise is generated by randomly transforming a selected Model vertex along the surface, such that the Euclidean distance between the original and perturbed point is uniformly distributed. An algorithm for generating this noise is presented below.

Data Collection Uncertainty Noise Generation Algorithm

- Preprocessing - performed once during experiment initialization.
    - For every vertex, *v*, in the Model:
        - Compute a set, *F*, of all facets of the triangle mesh which are connected to *v* via edges which lie completely within a sphere of radius r_u centered at *v*. Facets which intersect the sphere are also included in the resulting set, *F*.
        - For each facet *f* in *F*, compute Area(*f*) / Σ Area(*f*), the fractional area of the facet relative to all facets in *F*. The area of each facet which intersects the sphere is approximated.
        - Cache the set *F* and the fractional area information together with the vertex *v*.
- Generation - performed for each Data set to which collection uncertainty noise is added.
    - For every Model vertex, *v*, in the nominal Data set:
        - Randomly select a facet *f* from the set *F* associated with the vertex *v*. The likelihood of selecting a particular facet from the set *F* is proportional to the fractional area computed during the preprocessing step.
        - Randomly select a point within the selected facet, *f*, from a uniform distribution. Perform proper clipping to ensure that this point lies within the sphere of radius r_u centered at *v*.

This algorithm adds noise to the Data which approximates the effects of Data collection uncertainty described in Section 4.3. After uncertainty noise has been added, sensor noise can be added using the methods described in Section 3.4. The flow of a single registration experiment for evaluating constraint synthesis is similar to the experimental flow presented in Figure 3-15, with the addition of a step to add uncertainty noise before sensor noise is added. Refer to Section 3.4 and Figure 3-15 for a complete description of the experimental method.

4.4.2 Evaluation of Constraint Synthesis Search Algorithms

This subsection compares the various search algorithms in terms of their ability to solve the constraint synthesis problem. It also demonstrates the superiority of automatically synthesized Data configurations compared to randomly or manually generated configurations in terms of the resulting NAI values.

To evaluate search algorithm effectiveness, each algorithm described in Section 4.2 was used to synthesize fixed-size configurations of 10, 25, 50 and 75 points for the Venus, Femur and Pelvis surface Models of Figure 3-21. For each configuration size, 5 different Data collection plans (DCPs) were generated. The results of the experiment for the Venus Model are reported in Table 4-1. Seven methods were used for generating configurations: random, manual, SAH, NAH, PBIL, hybrid PBIL-SAH and hybrid PBIL-NAH. The manually selected configurations were hand-picked by a surgeon, two graduate students with strong backgrounds in 3-D computer vision, and the author. The randomly generated configurations were evaluated over 1000 trials, while each of the other configurations were evaluated over 5 trials.

Table 4-1: Constraint synthesis results for the Venus Model as a function of synthesis method and configuration size. Results are reported as maximum/minimum NAI values over 5 trials.

Method	Configuration Size			
	10	25	50	75
Random Generation (1000 trials)	0.19 / 0.00	0.33/ 0.02	0.51 / 0.08	0.60 / 0.13
Manual Selection	0.08 / 0.05	0.44 / 0.12	0.40 / 0.19	0.45 / 0.29
Steepest Ascent Hillclimbing (SAH)	0.61 / 0.44	1.02 / 0.71	1.48 / 1.14	1.72 / 1.49
Next Ascent Hillclimbing (NAH)	0.63 / 0.55	1.19 / 1.09	1.65 / 1.49	2.06 / 1.96
PBIL	0.71 / 0.64	1.25 / 1.18	1.76 / 1.72	2.13 / 2.06
PBIL-SAH	0.73 / 0.65	1.27 / 1.23	1.80 / 1.75	2.21 / 2.14
PBIL-NAH	0.75 / 0.65	1.25 / 1.22	1.81 / 1.76	2.17 / 2.12

From the table it can be seen that the three search methods which use PBIL result in larger values of the NAI than any of the other automatic generation methods, and significantly larger values than randomly or manually generated configurations. From these results, there is no strong evidence as to which of the three PBIL-based approaches is preferable. For randomly generated configurations, the maximum NAI values tend to be larger than the corre-

sponding values for manually generated configurations; however, minimum NAI values are larger for manually generated configurations. The NAI values for each of the automatic generation methods are significantly larger than the corresponding manually generated configurations. This is true even when the configuration size is large. This suggests that humans are not very good at solving the constraint synthesis problem for complex geometries. Additional results which support this conclusion are presented in Chapter 5.

The plots of Figure 4-9 demonstrate the evolution of the hybrid PBIL-NAH search algorithm in which the NAI criterion measure (top), and eigenvalue magnitudes from Equation 3.11 (bottom), are plotted as a function of search algorithm iteration. For this example, most of the improvement in NAI results from the PBIL portion of the search, while the NAH algorithm “fine-tunes” the result. Note that during the evolution of the search, the eigenvalues which are initially small become large, while those which are initially large become smaller. This behavior is expected based upon the goal of maximizing the volume and hypersphericity of the sensitivity ellipsoid discussed in Section 3.3.2.

The experiments used to generate the results of Table 4-1 were repeated using the Femur and Pelvis Models, and the results are presented in Tables 4-2 and 4-3, respectively. Due to the difficulty of manually selecting Data sets, this method of generation was omitted from the remaining results in this chapter (although additional results with manually-selected Data sets are presented in Chapter 5). From the tables below, it can be seen that the hybrid methods consistently perform better than any of the other methods with respect to both maximum and minimum NAI values over the 5 trials.

Execution times for the five constraint synthesis search algorithms are reported in Table 4-4, and range from about 4 minutes per configuration to about 5 hours per configuration, depending upon method and configuration size. All times are reported in minutes and represent the times required to synthesize 5 independent configurations. These constraint synthesis trials were performed on several different computers, all SGI Indigo2 Extremes with R4400 200 MHz processors. Execution times for the NAH approach are consistently faster than for the SAH approach, both as a stand-alone algorithm, and as a component of the hybrid PBIL-hillclimbing approach, and therefore the NAH approach is preferred over SAH. The hybrid PBIL-NAH approach tends to result in slightly larger NAI values than the NAH approach; however, NAH is significantly faster than the hybrid approach. For the experimental results reported in the remainder of this dissertation, the hybrid PBIL-NAH approach is used unless otherwise indicated. In the future, it may be desirable to use the NAH approach alone if it can be shown that the resulting reduction in NAI value does not have a significant impact upon registration accuracy.

As suggested earlier, due to clinical access limitations, it may not be possible or desirable to collect registration Data within certain regions of an anatomical object during surgery. For

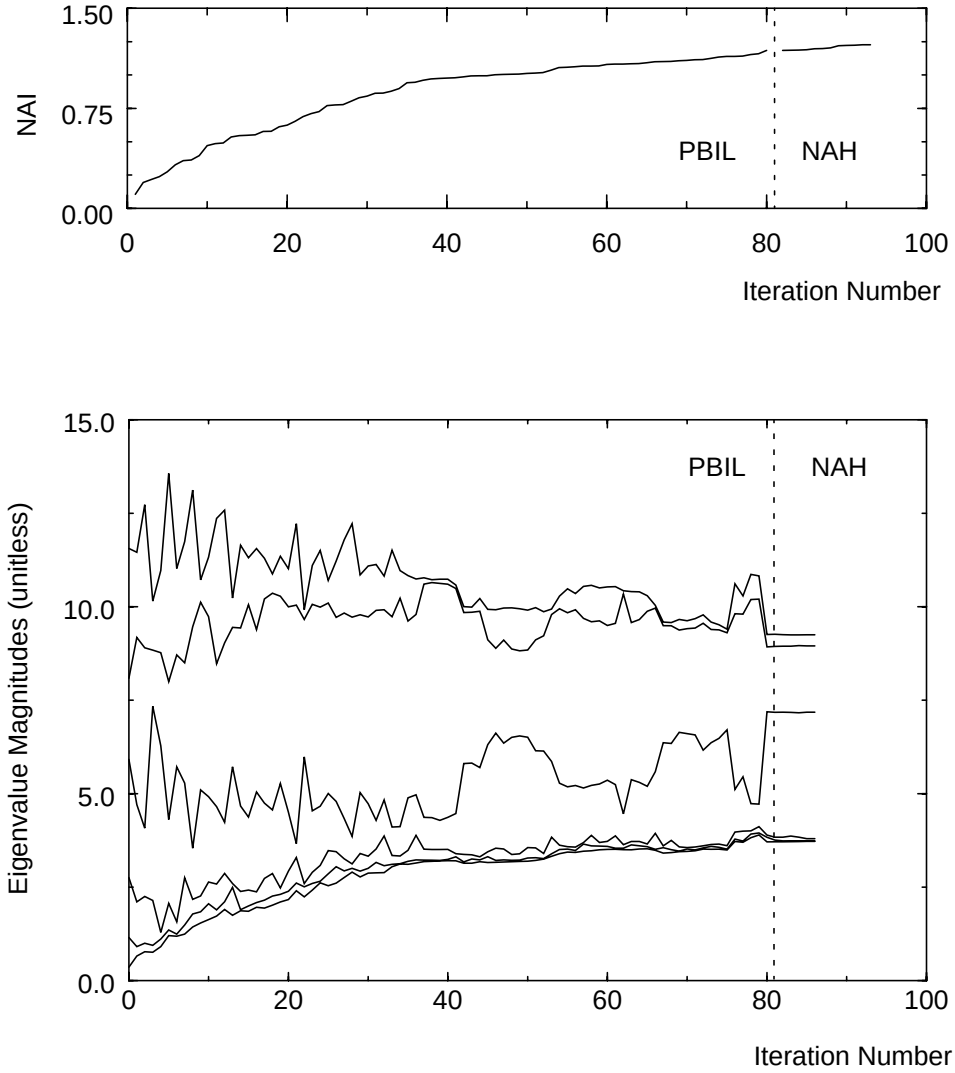


Figure 4-9: Evolution of the hybrid PBIL-NAH search algorithm for a single trial using the Venus Model and a configuration size of 25 points. The iterations to the left of the vertical line correspond to PBIL, while those to the right correspond to NAH. Top: NAI vs. iteration. Bottom: eigenvalue magnitude vs. iteration.

Table 4-2: Constraint synthesis results for the Femur Model as a function of synthesis method and configuration size. Results are reported as maximum/minimum NAI values over 5 trials.

Method	Configuration Size			
	10	25	50	75
Random Generation (1000 trials)	0.31 / 0.00	0.59 / 0.48	0.80 / 0.21	0.97 / 0.32
Steepest Ascent Hillclimbing (SAH)	0.94 / 0.79	1.76 / 1.52	2.48 / 2.38	3.20 / 2.38
Next Ascent Hillclimbing (NAH)	0.98 / 0.80	1.73 / 1.52	2.49 / 2.29	3.07 / 2.96
PBIL	0.97 / 0.91	1.77 / 1.68	2.51 / 2.44	3.02 / 2.98
PBIL-SAH	1.04 / 0.93	1.82 / 1.72	2.59 / 2.54	3.19 / 3.14
PBIL-NAH	1.06 / 0.97	1.80 / 1.74	2.59 / 2.56	3.20 / 3.11

Table 4-3: Constraint synthesis results for the Pelvis Model as a function of synthesis method and configuration size. Results are reported as maximum/minimum NAI values over 5 trials.

Method	Configuration Size			
	10	25	50	75
Random Generation (1000 trials)	0.41 / 0.00	1.18 / 0.08	1.52 / 0.33	1.68 / 0.53
Steepest Ascent Hillclimbing (SAH)	1.39 / 1.15	2.67 / 2.41	3.86 / 3.55	4.87 / 4.33
Next Ascent Hillclimbing (NAH)	1.42 / 1.28	2.62 / 2.43	3.97 / 3.76	4.90 / 4.79
PBIL	1.41 / 1.35	2.70 / 2.58	3.88 / 3.81	4.84 / 4.76
PBIL-SAH	1.51 / 1.36	2.76 / 2.68	4.00 / 3.89	4.96 / 4.91
PBIL-NAH	1.52 / 1.36	2.75 / 2.65	4.02 / 3.92	4.94 / 4.90

Table 4-4: Execution times for constraint synthesis of the Femur Model as a function of synthesis method and configuration size. Results are reported in minutes and represent the synthesis of 5 configurations. (* = not available)

Method	Configuration Size			
	10	25	50	75
Steepest Ascent Hillclimbing (SAH)	79	289	664	*
Next Ascent Hillclimbing (NAH)	21	99	209	424
PBIL	315	350	402	704
PBIL-SAH	+ 25	+ 124	+ 245	+ 757
PBIL-NAH	+ 22	+ 97	+ 174	+ 383

example, during total hip replacement surgery, only a few regions on a pelvis can be accessed with a digitizing probe for Data collection, as indicated in Figure 4-10. These clinical access limitations can be easily incorporated into the constraint synthesis process by using a surface Model which only includes the accessible regions. Constraint synthesis will operate without modification on a surface Model containing multiple disconnected patches, such as the one shown in Figure 4-10.

Table 4-5 shows the results of applying the five constraint synthesis search methods to the clinically accessible regions of the Pelvis surface Model. Despite the simplicity of this

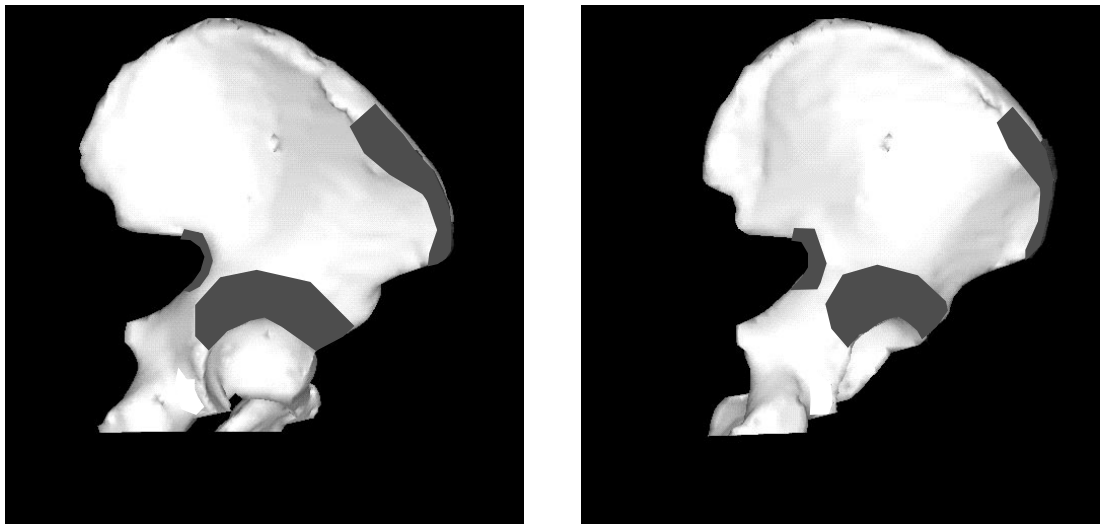


Figure 4-10: Two views of clinically accessible regions for Data collection from the surface of the Pelvis during total hip replacement surgery (accessible regions indicated in dark gray). The resulting Model is referred to as the Clinical Pelvis.

Table 4-5: Constraint synthesis results for the Clinical Pelvis Model as a function of synthesis method and configuration size. Results are reported as maximum/minimum NAI values over 5 trials.

Method	Configuration Size			
	10	25	50	75
Random Generation	0.24 / 0.00	0.44 / 0.02	0.60 / 0.12	0.72 / 0.24
Steepest Ascent Hillclimbing (SAH)	0.55 / 0.48	0.94 / 0.89	1.35 / 1.33	1.66 / 1.61
Next Ascent Hillclimbing (NAH)	0.54 / 0.53	0.98 / 0.94	1.42 / 1.36	1.72 / 1.66
PBIL	0.58 / 0.57	1.00 / 0.98	1.44 / 1.44	1.77 / 1.75
PBIL-SAH	0.58 / 0.57	1.00 / 0.98	1.45 / 1.44	1.78 / 1.77
PBIL-NAH	0.58 / 0.57	1.00 / 0.98	1.45 / 1.44	1.78 / 1.77

Model, the various automatic approaches for synthesizing configurations all perform significantly better than the random method.

This subsection has evaluated the various constraint synthesis search algorithms using several surface Models. The next subsection examines the NAI *stability* of generated configurations as a function of Data collection uncertainty.

4.4.3 NAI Stability of Synthesized Configurations

As discussed in Section 4.3, object pose uncertainty during the Data collection process can reduce the effective NAI value (computed using collected Data) relative to the ideal NAI value (computed using the DCP). To demonstrate this effect, experiments were performed in which Data collection uncertainty noise (see Section 4.4.1) was added to the nominal Data points specified by a DCP, and the resulting effective NAI values were computed. The results of such an experiment are illustrated in the two graphs of Figure 4-11. These graphs show the ideal NAI values, and the effect on the NAI of adding Data collection uncertainty noise to five nominal DCPs over 1000 trials. The mean, 5th and 95th percentiles of the effective NAI computed over 1000 trials are plotted for the 5 synthesized configurations. The graphs were generated using the Pelvis Model and 25-point Data configurations. The top graph was generated using a Data collection uncertainty noise radius, r_u, of 5 mm, and the bottom one was generated with an r_u of 10 mm. Note that the configuration which has the largest ideal NAI value is not necessarily the one which is most stable. As discussed in Section 4.3, the ultimate goal of constraint synthesis is to maximize the *effective* NAI. The results of experiments such as these can be used to determine which configurations are the most stable and are therefore the most likely to have the largest effective NAI values.

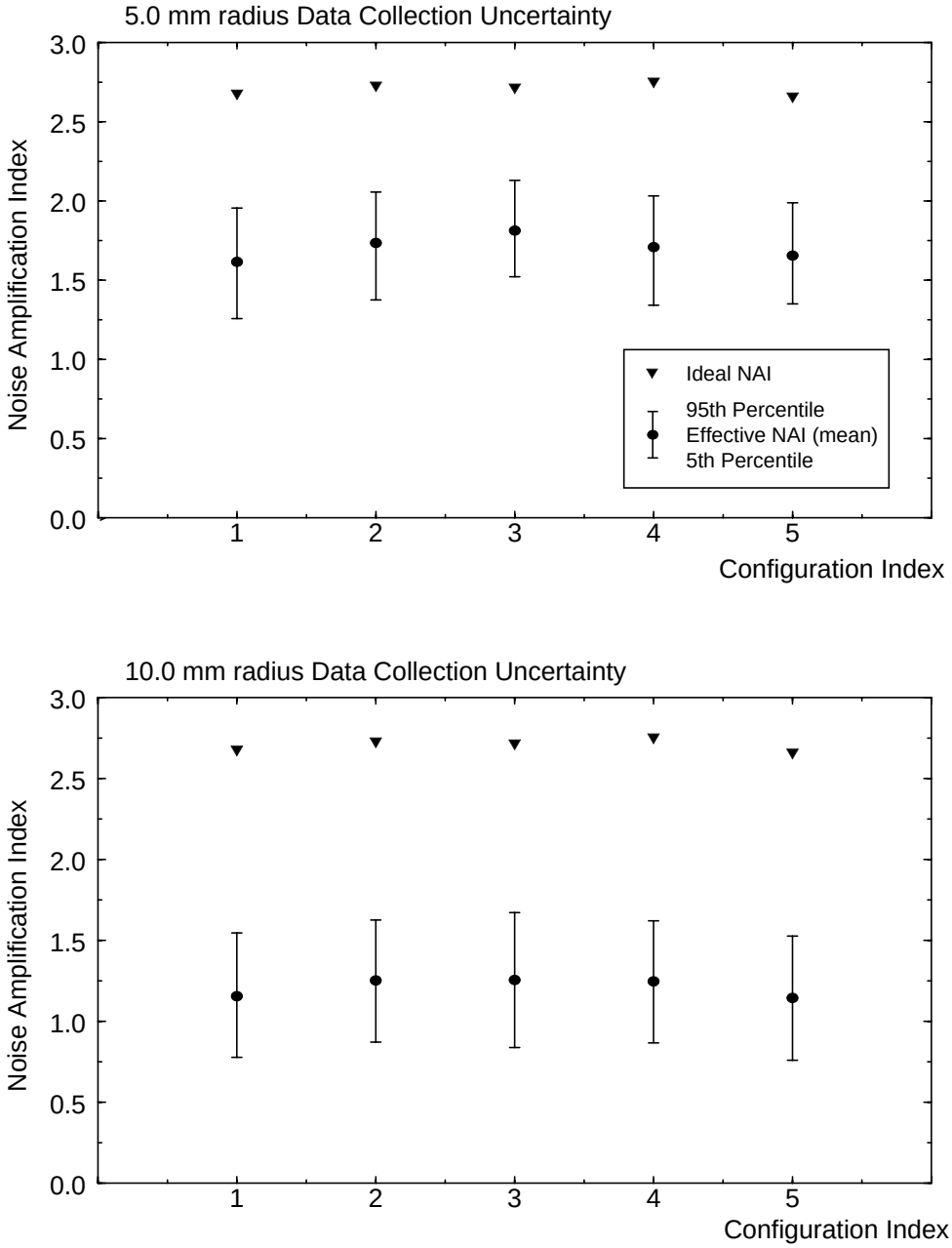


Figure 4-11: Effect of Data collection uncertainty on the NAI over 5 synthesized, 25-point configurations for the Pelvis Model. The ideal NAI is computed from the DCP. The effective NAI is computed over 1000 trials of adding Data collection uncertainty noise to the DCP. Top: 5.0 mm radius collection uncertainty. Bottom: 10.0 mm radius collection uncertainty.

In the graphs of Figure 4-11, the difference between ideal and effective NAI is a function of the magnitude of Data collection uncertainty. Larger values of Data collection uncertainty result in larger differences between ideal and effective NAI values. If Data collection uncertainty is too large, the resulting effective NAI values will be no better than those for randomly generated configurations. For comparison, the 5th and 95th percentiles of effective NAI values generated from one thousand 25-point *random* configurations for the Pelvis Model are 0.21 and 0.64, respectively. Therefore, even for 10 mm Data collection uncertainty, the synthesized configurations perform significantly better than random configurations.

As suggested in Section 4.3, filtering a surface Model by eliminating regions of high curvature before constraint synthesis can sometimes significantly improve the stability of the resulting Data configuration. An example of a filtered surface Model is shown in Figure 4-12 in which the filtering parameters, δ and *r,* are 90 degrees and 5 mm, respectively. The effect of high curvature filtering upon configuration stability is illustrated in the graphs of Figure 4-13 which plot ideal and effective NAI values versus configuration size for the Femur Model. To generate these graphs, five synthesized configurations were generated from a filtered Model, and another five configurations were generated from an unfiltered Model. Data collection uncertainty noise was added to each configuration and the statistical means of the effective NAI values were computed over 1000 trials. For each configuration size, the single configuration corresponding to the median of the effective NAI

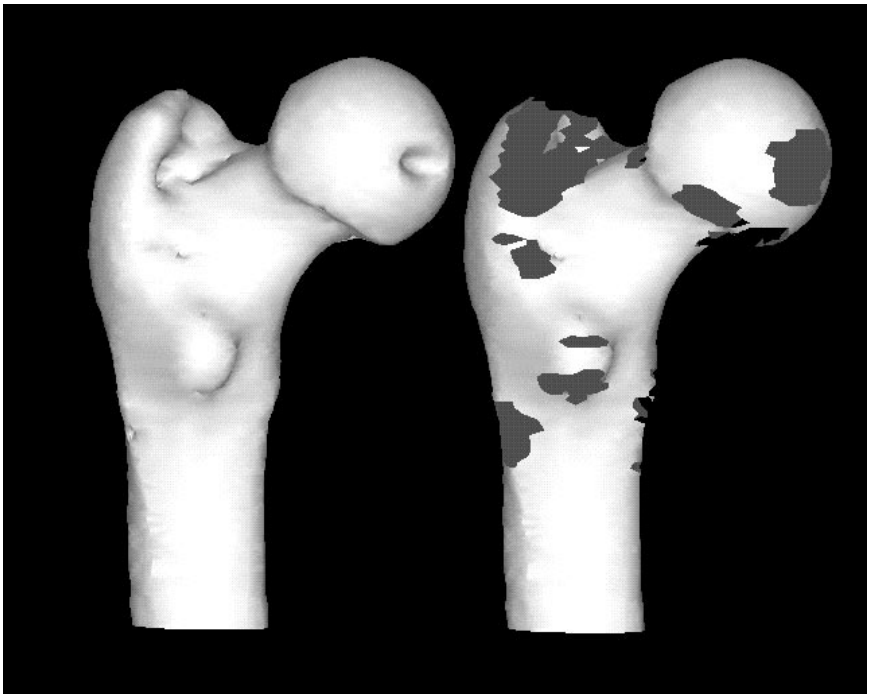


Figure 4-12: Views of the unfiltered (left) and filtered (right) Femur Model.

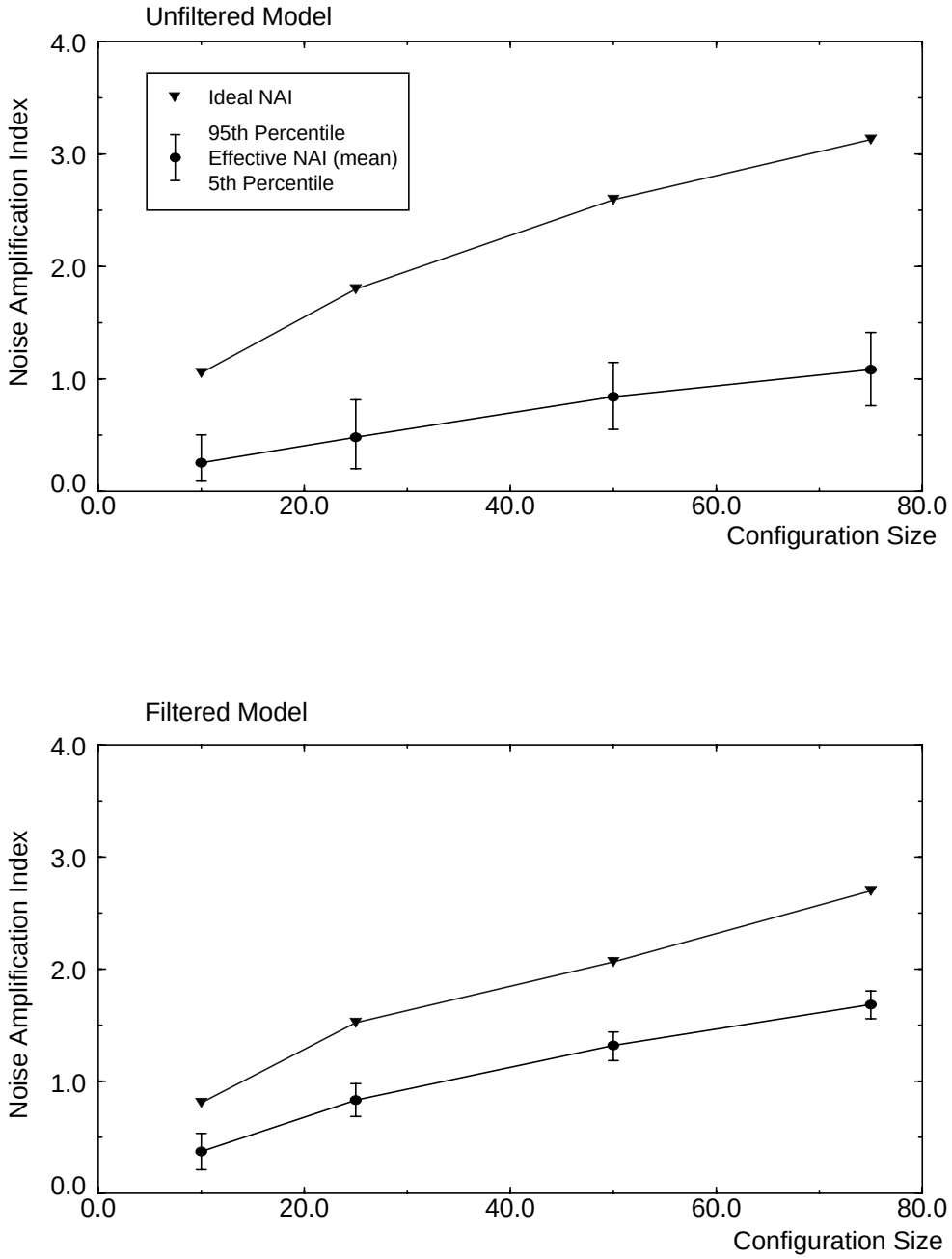


Figure 4-13: Effect of high-curvature filtering on Data collection uncertainty for the Femur Model with 5.0 mm radius Data collection uncertainty. Effective and ideal NAI values plotted vs. configuration size. The ideal NAI is computed from the DCP. The effective NAI is computed over 1000 trials of adding Data collection uncertainty noise to the DCP. Top: unfiltered Model. Bottom: filtered Model.

mean value (over the 5 configurations) is plotted. There are several important trends in these graphs. First, both ideal and effective NAI increase as a function of Data configuration size. Second, ideal NAI values are smaller for the configurations synthesized from the filtered Model. This is because removal of high curvature regions from the Model reduces the variation in facet orientation, and thus the potential for large values of the NAI. Third, despite the reduction in ideal NAI values for the filtered Model, the effective NAI values for this Model are larger than those for the unfiltered Model. This is why high curvature filtering is useful; removing high curvature regions from a Model may increase configuration stability with respect to Data collection uncertainty.

High curvature filtering is not always necessary to achieve stable Data configurations. Among the 4 Models used in the current experiments (Pelvis, Clinical Pelvis, Femur and Venus), only the Femur Model significantly benefited from application of the filtering process. Configurations generated using the other Models were fairly stable without filtering. Whether or not filtering will improve configuration stability is related to the geometry of the object, the resolution of the Model (i.e., size of the planar facets) and the noise characteristics of the Model. Figure 4-14 shows an unfiltered and filtered Pelvis Model using the same filtering parameters as above. For this Model, there are only a small number of high curvature regions which could potentially reduce stability. Therefore, configurations synthesized from the filtered Pelvis are no more stable than those generated from the unfiltered Pelvis as seen in the graphs of Figure 4-15.

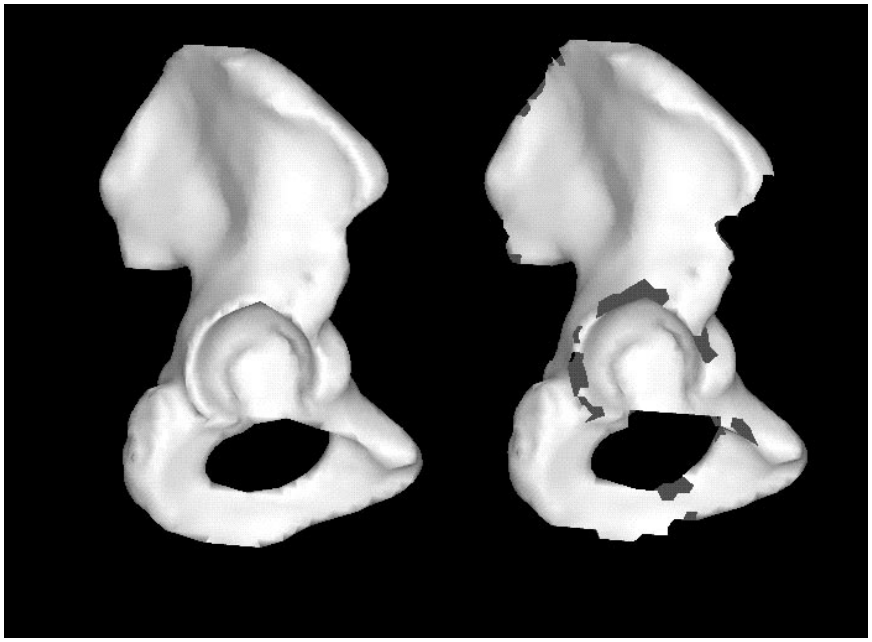


Figure 4-14: Views of the unfiltered (left) and filtered (right) Pelvis Model.

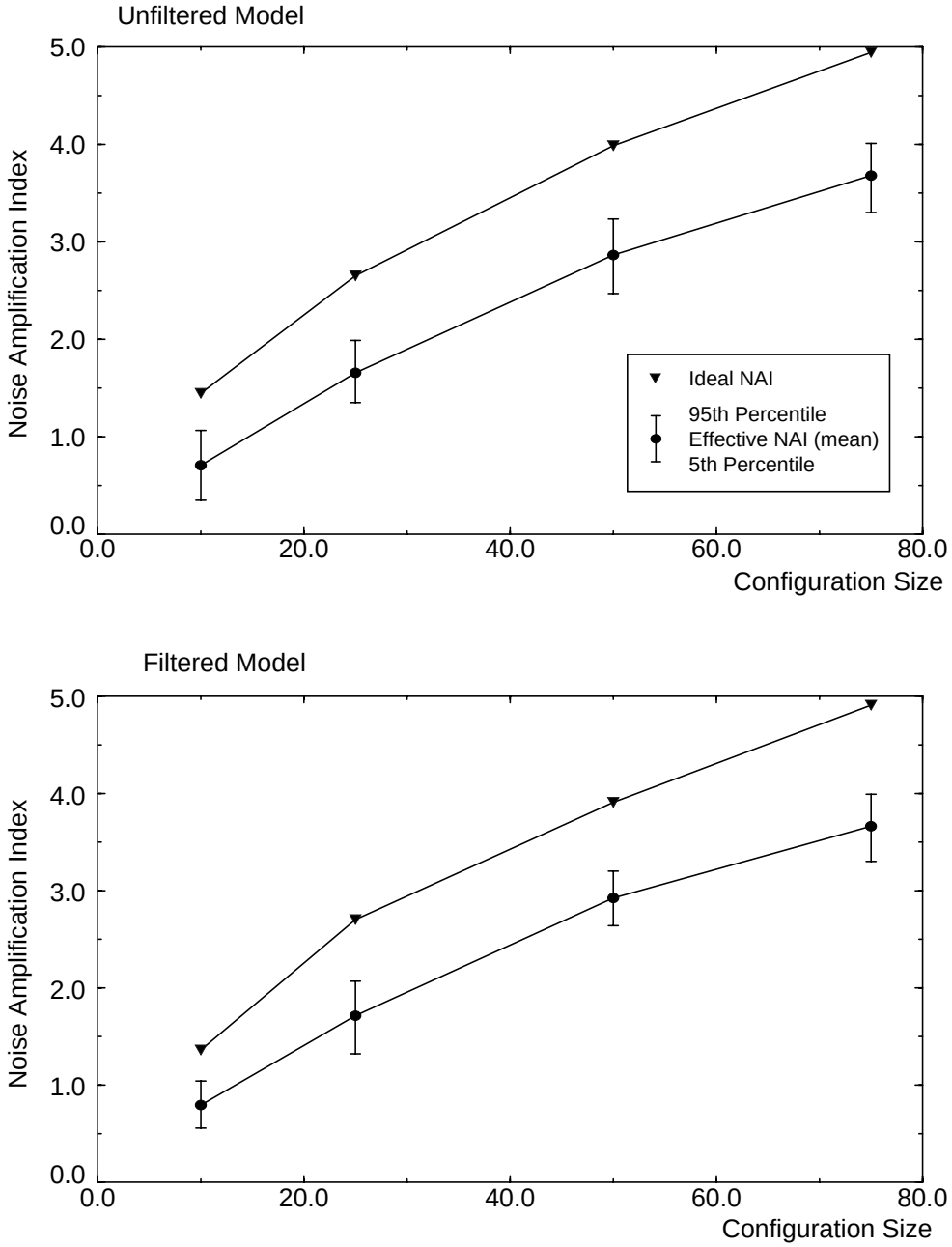


Figure 4-15: Effect of high-curvature filtering on Data collection uncertainty for the Pelvis Model with 5.0 mm radius Data collection uncertainty. Effective and ideal NAI values plotted vs. configuration size. The ideal NAI is computed from the DCP. The effective NAI is computed over 1000 trials of adding Data collection uncertainty noise to the DCP. Top: unfiltered Model. Bottom: filtered Model.

The results of Figures 4-13 and 4-15 demonstrate that larger configuration sizes result in improved NAI stability. In both sets of results, the ratios of mean effective NAI to ideal NAI increase with increasing configuration size. Figure 4-16 illustrates the relation between configuration stability and the magnitude of Data collection uncertainty. In this figure, ideal and effective NAI are plotted versus Data collection uncertainty noise radius, r_u. It can be seen that larger values of Data collection uncertainty result in reduced configuration stability. An important observation can be made based upon the results of Figures 4-13, 4-15 and 4-16. Given a registration accuracy requirement which must be satisfied, the number of Data points needed to satisfy this requirement will depend in part upon the magnitude of the associated Data collection uncertainty. Due to the Data collection guidance mechanism described in Chapter 5, high quality initial pose estimates will tend to reduce Data collection uncertainty, and thus the number of Data points required to localize an object to within fixed accuracy tolerances.

4.4.4 Registration Accuracy Resulting from Synthesized Configurations

The previous subsection examined the NAI stability of synthesized configurations with respect to Data collection uncertainty. The current subsection demonstrates that synthesized configurations result in superior registration accuracy, despite the presence of Data collection uncertainty and sensor noise. The results presented in this section are based upon the experimental method outlined in Figure 3-15 which uses synthesized Data. Similar results using Data collected by a human Data collector with a digitizing probe can be found in Chapter 5. A complete description of the experimental method and relevant parameters can be found in Sections 3.4.2 and 4.4.1. A summary of the registration parameters for the experiments of this section appear in Table 4-6.

Figure 4-17 shows the results of 9 registration accuracy experiments using the Venus Model. Each graph represents the results of a single experiment, and the title in the center of each graph indicates the associated noise parameters and Data configuration sizes. For example, the graph titled “n0.5_s5_25_Venus” contains the results of an experiment which had an expected noise magnitude, μ_Δ, of 0.5, an uncertainty radius, r_u, of 5.0 mm, and a configuration size of 25 points. For all graphs, the x-axis represents the ideal noise amplification index (NAI) computed from the nominal DCP, and the y-axis represents statistics of the maximum correspondence error (MCE) computed over 500 registration trials (i.e., mean, standard deviation, 5th and 95th percentiles). Note that the scales along the x-axes within a column of graphs are the same, but the scales of the y-axes differ for each graph. Within each graph, there are two distinct groupings of results; the grouping on the left is for randomly generated configurations (as indicated by the small NAI values), while the grouping on the right is for synthesized configurations. Within the 3x3 grid of graphs, moving from left to right across a row corresponds to an increase in the Data configuration size, while moving from top to bottom within a column corresponds to an increase in sensor noise. Data

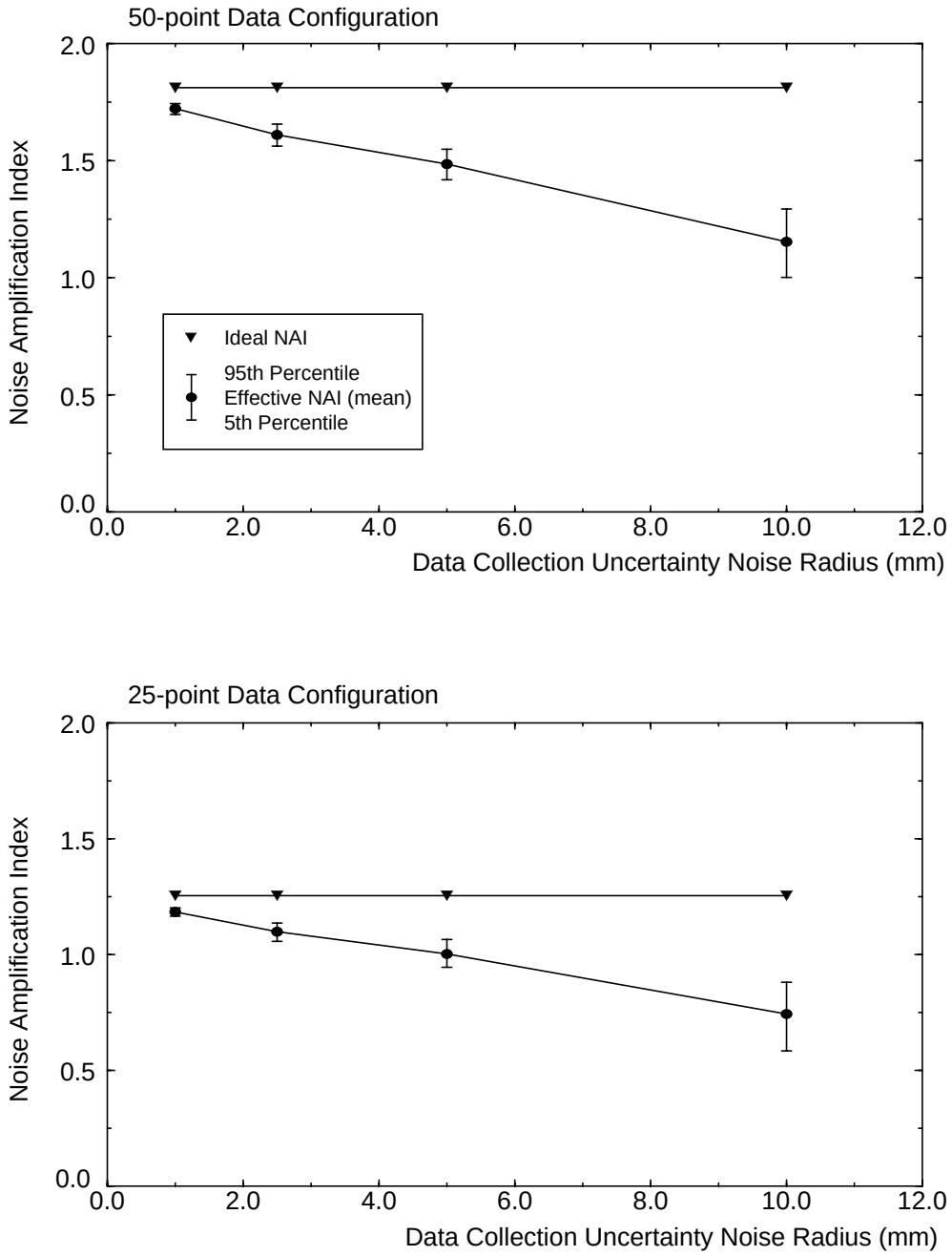


Figure 4-16: Effect of Data collection uncertainty radius, $r_{u,}$ on NAI for the unfiltered Venus Model. Ideal and effective NAI values are plotted versus Data collection uncertainty radius. Top: 50-point configuration. Bottom: 25-point configuration.

Table 4-6: Registration parameters for experiments reported in Section 4.4.4.

Parameter	Value
Data source	Random & Synthesized
Data configuration size	10, 25, 50 points
τ_{imax}	20.0 mm
θ_{imax}	10 deg
Noise state	Enabled
μ_{Δ}	0.1, 0.5, 1.0 mm
Uncertainty noise	Enabled
r_u - uncertainty radius	5.0, 10.0 mm
pose_count	500
set_count	5 random, 5 synthesized
Minima suppression	Enabled
$\tau_{p\text{-}max}$	10.0 mm
$\theta_{p\text{-}max}$	5.0 degrees
I_p	6 iterations

collection uncertainty is the same for all graphs within the grid; the uncertainty radius, r_u, is 5.0 mm.

Several trends are apparent from these results. First, for every experiment in Figure 4-17, the synthesized configurations perform better than the randomly generated configurations with respect to the MCE statistics. Moving down within a single column of graphs (i.e., increasing the magnitude of sensor noise) results in an increase in the MCE values. However, the ratio of MCE values for random configurations to MCE values for synthesized configurations remains roughly constant as a function of increasing sensor noise. Moving left to right across a row (i.e., increasing the number of points in a configuration) results in larger values of the NAI for both synthesized and random configurations, and corresponding smaller values of the MCE. These results suggest that constraint synthesis is useful despite uncertainty in the Data collection process. This conclusion is supported by the remaining results in this section, and the results based upon actual Data reported in Chapter 5.

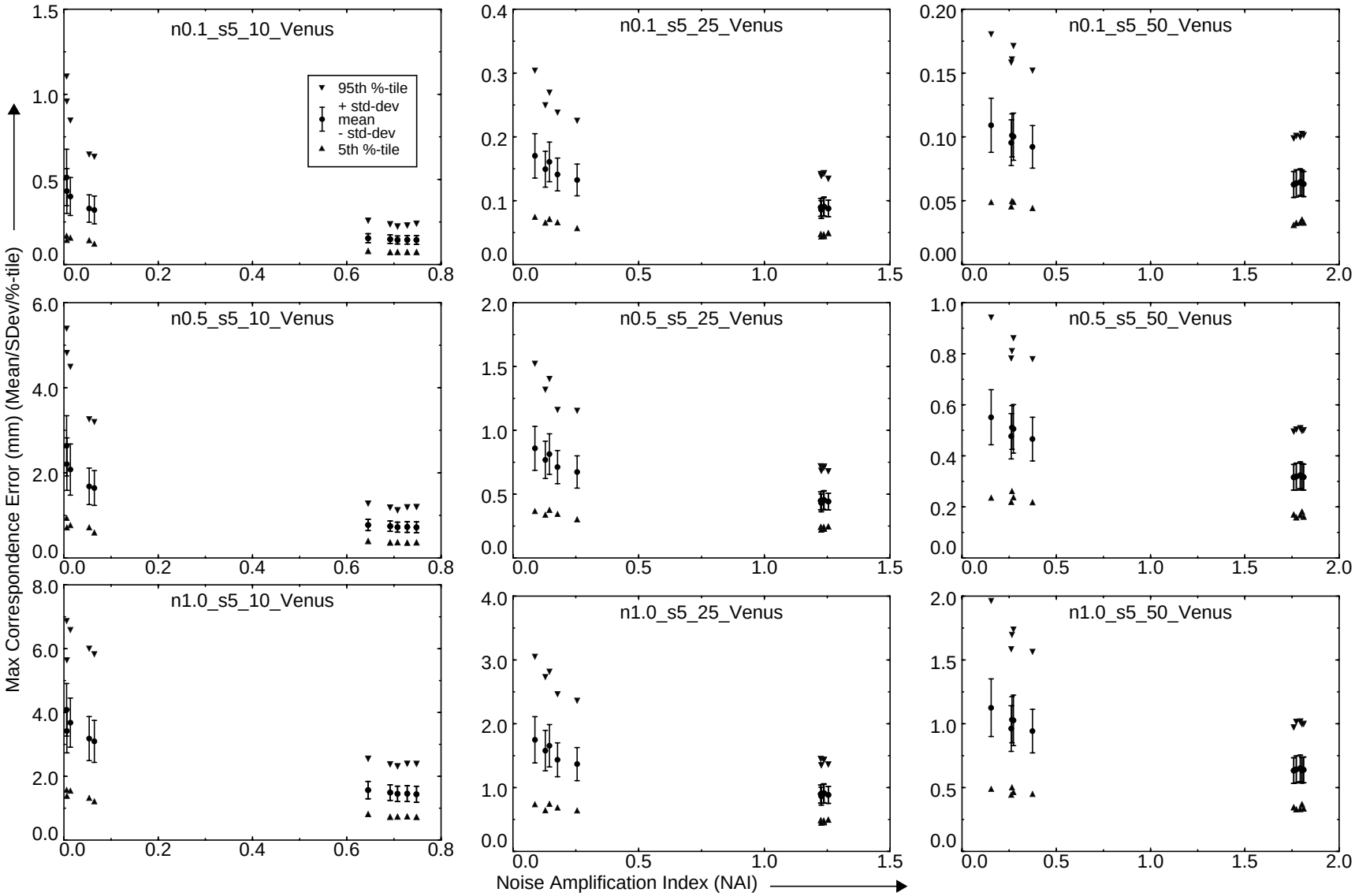


Figure 4-17: MCE vs. NAI for the Venus Model for random and synthesized configurations. Data collection uncertainty of 5.0 mm.

Figure 4-18 presents experimental results based upon the Pelvis Model which are qualitatively similar to the results for the Venus Model. The same value of the Data collection uncertainty radius, 5.0 mm, was used in both sets of experiments. Figure 4-19 presents results for the Pelvis using a larger value of the uncertainty radius, 10.0 mm. Comparing the results of Figures 4-18 and 4-19, it can be seen that the relative utility of the synthesized configurations decreases with increasing Data collection uncertainty, as expected. The most apparent differences between these figures are seen in the leftmost columns of the graphs corresponding to the configurations with the smallest number of points (i.e., 10). The results of the other two columns are quantitatively similar in both the figures. The fact that increasing Data collection uncertainty has the strongest effect upon the smallest configurations is expected from the observations of the Section 4.4.3.

As observed in Section 4.4.3, eliminating regions of high curvature from certain Models can result in improved NAI stability. This is illustrated in the results of the registration experiments presented in Figure 4-20. The graphs in this figure contain three groupings of results corresponding to random (leftmost), synthesized (rightmost), and synthesized/filtered (middle) configurations. The configurations which were synthesized from the filtered Model exhibit smaller MCE values than either the random or unfiltered configurations due to the improvement in NAI stability resulting from the filtering process (as described in Section 4.4.3.).

The final results of this section are for the Clinical Pelvis Model of Figure 4-10 in which only regions which are clinically accessible during total hip replacement surgery are included. Experimental registration results are shown in Figure 4-21, and the trends evident in the graphs are similar to those reported earlier in this section. This result is important because it demonstrates that constraint synthesis performs well compared to random Data selection, even when the size of the Model is relatively small. This particular application (i.e., registration of the pelvis during total hip replacement surgery) will likely be the first clinical demonstration of the Intelligent Data Selection method.

4.5 Discussion

The improvement in registration accuracy resulting from the application of Intelligent Data Selection (IDS) is a function of many factors including:

- The magnitude of noise in the Data collection sensor.
- The magnitude of uncertainty in the Data collection process.
- The incremental cost of acquiring Data, and the number of Data points which can be “afforded”.

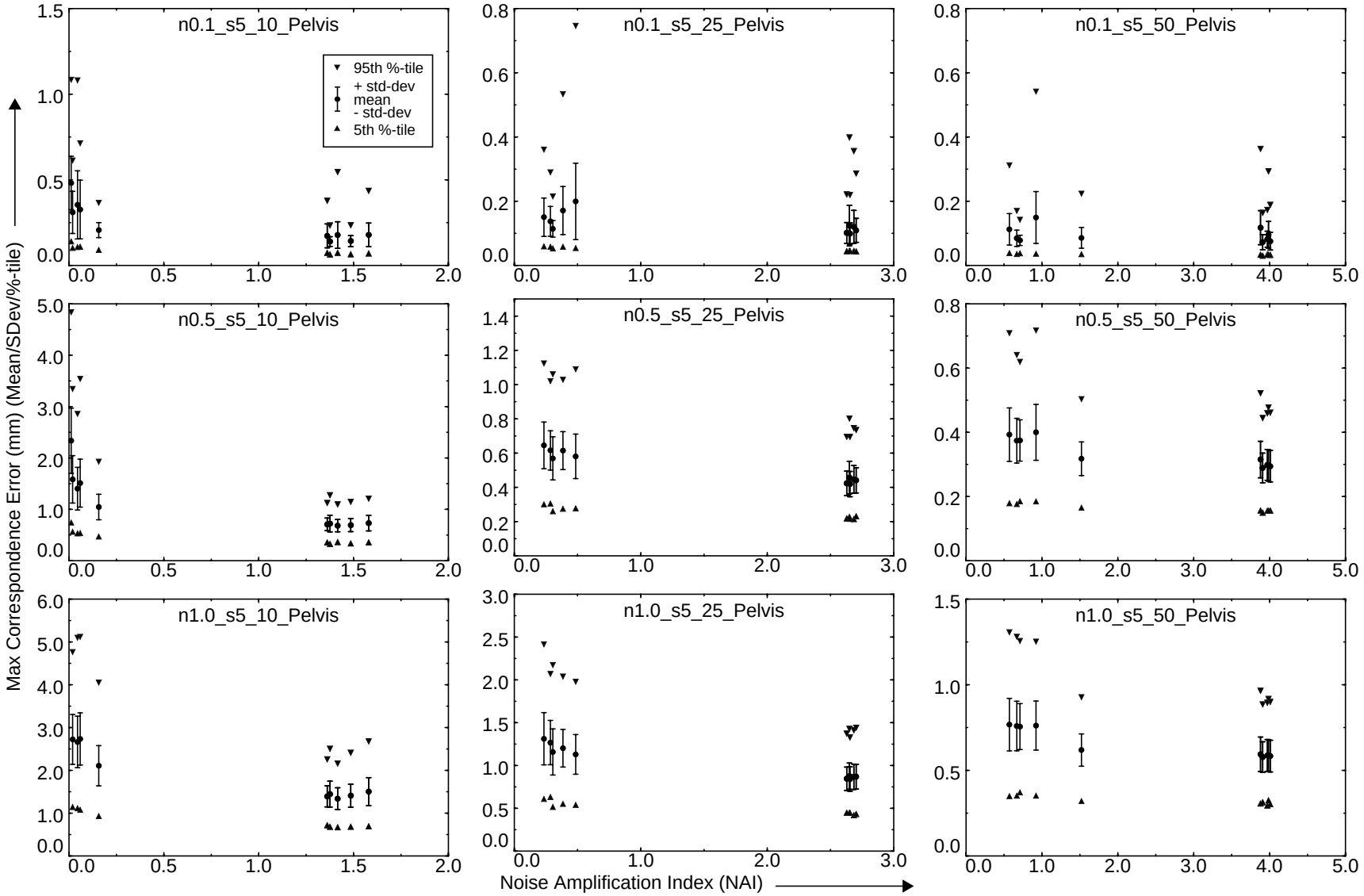


Figure 4-18: MCE vs. NAI for the Pelvis Model for random and synthesized configurations. Data collection uncertainty of 5.0 mm

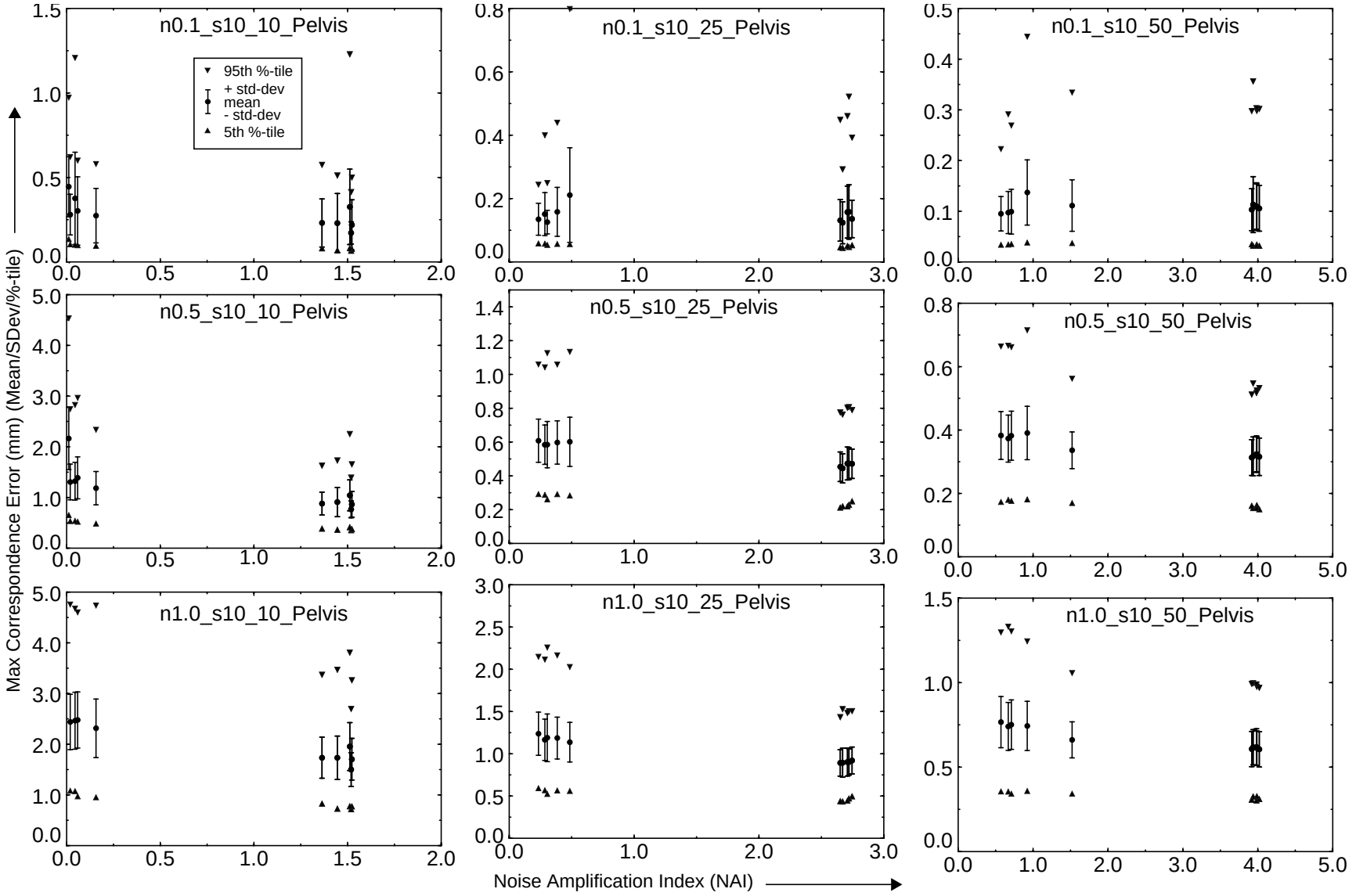


Figure 4-19: MCE vs. NAI for the Pelvis Model for random and synthesized configurations. Data collection uncertainty of 10.0 mm.

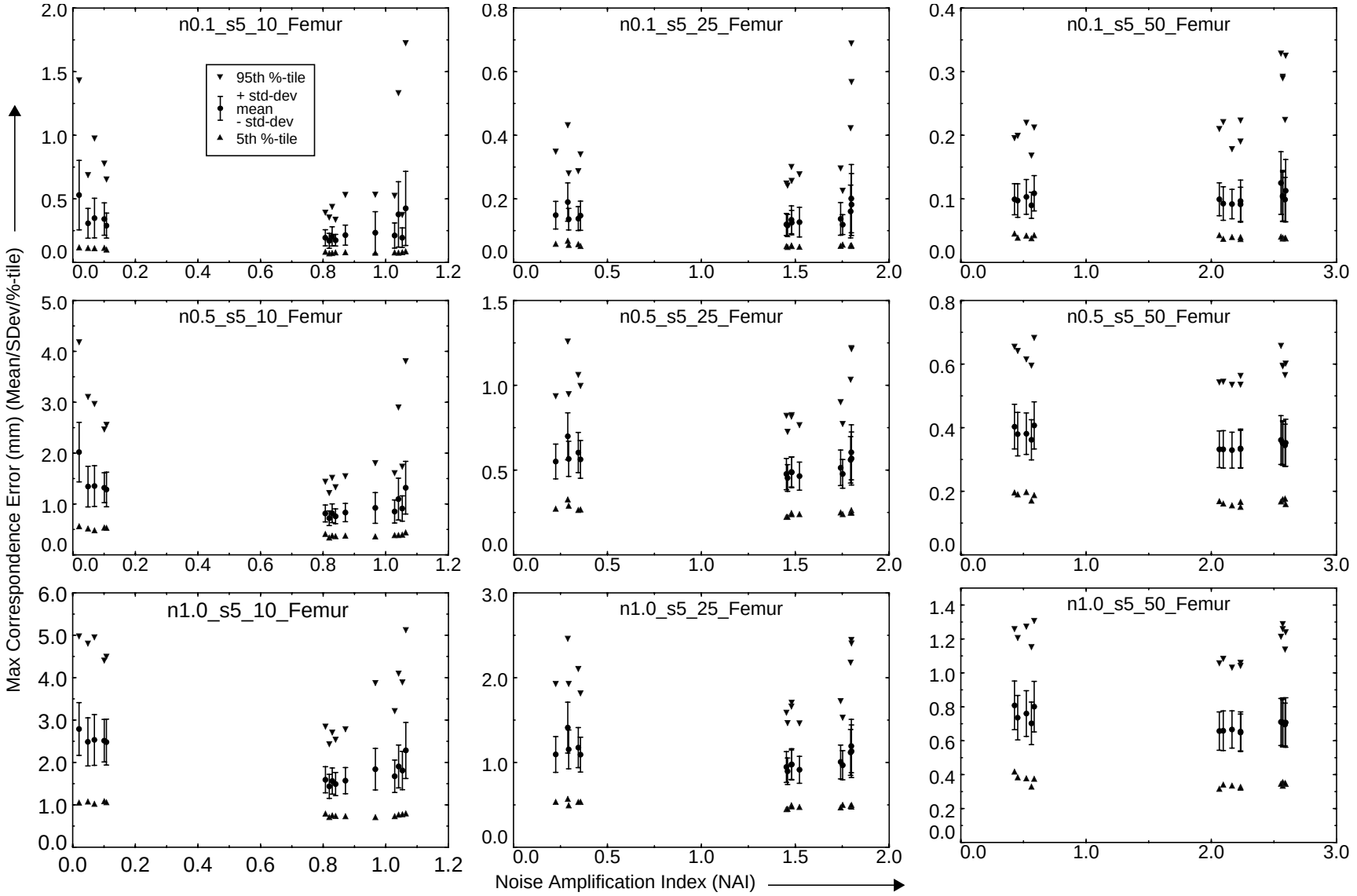


Figure 4-20: MCE vs. NAI for the Femur Model for random and synthesized configurations. Data collection uncertainty of 5.0 mm.

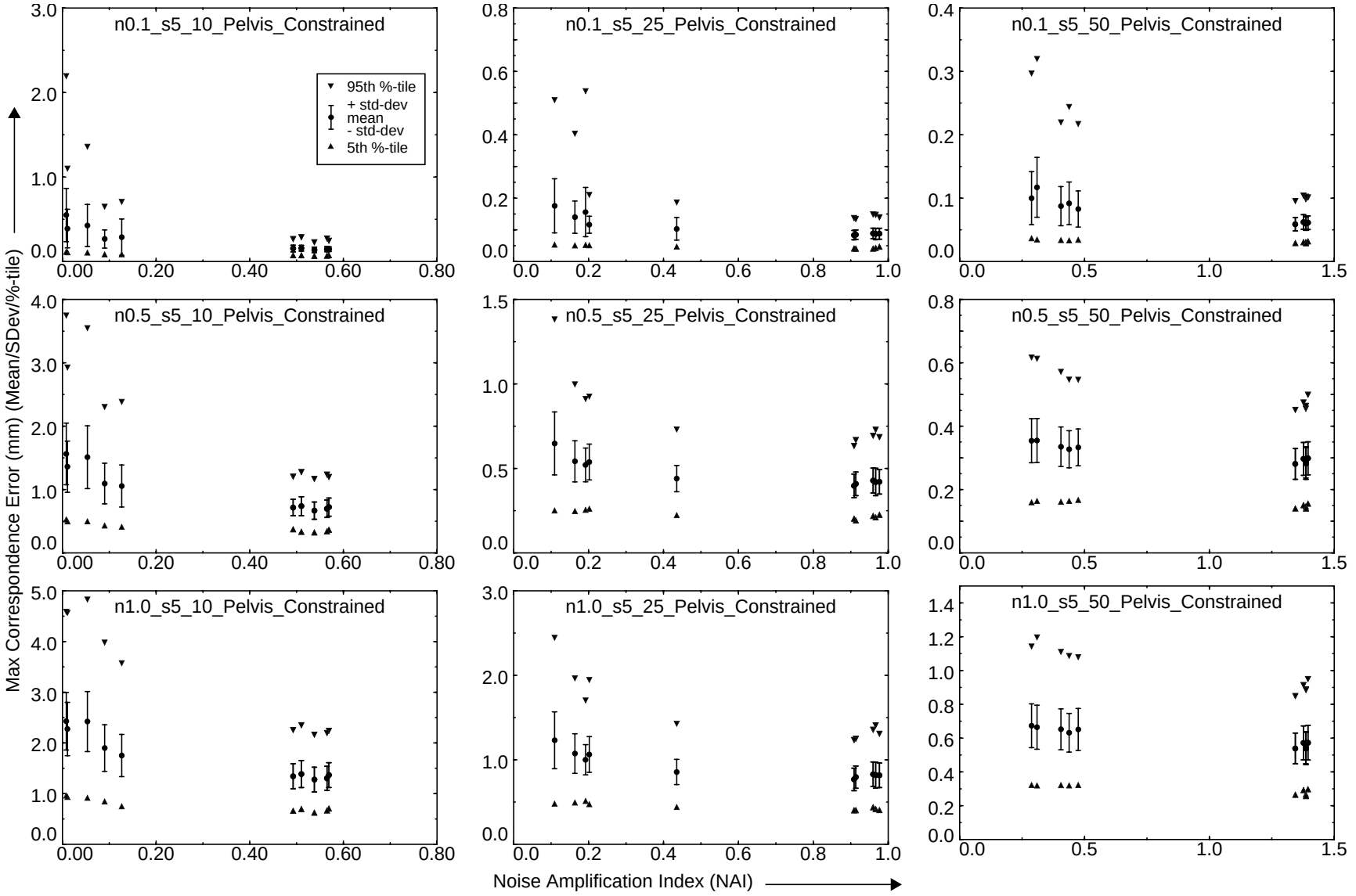


Figure 4-21: Clinical Pelvis Registration Results

As discussed in this chapter, increasing uncertainty in the Data collection process will have an adverse effect upon the utility of IDS. The amount of uncertainty which can be tolerated is a function of the shape of the object being registered and the number of points which can be collected. Data collection uncertainty has the largest effect upon the NAI when the size of the Data configuration is small and there are many regions of high curvature on the object.

Noise in the Data collection sensor will affect the accuracy which can be achieved by shape-based registration, but does not seem to have a strong effect on the utility of the IDS method (as measured by the relative performance of random and synthesized configurations as a function of increasing sensor noise). Of course, if sensor noise becomes too large, it will also increase Data collection uncertainty, and thus the utility of IDS will decrease.

While the experiments in this chapter have not dealt with the magnitude of noise in the Model (i.e., how well the Model represents the true shape), it is likely that this will also be an important factor in assessing the benefit of IDS. One reason that Model noise was not investigated is the difficulty in developing realistic mathematical models of noise in the Model generation process. This process includes not only image formation (i.e, from CT or MRI), but also the process of extracting geometric surface descriptions from the 3-D image data. It is unlikely that simple Gaussian noise models would realistically represent the errors introduced by these complex and highly non-linear processes. In addition, such a simple noise model is very similar to the one used to generate corrupted Data points in this chapter. Therefore, it is likely that corrupting the Model with Gaussian noise would result in similar experimental outcomes to those reported.

There are several other factors which will affect the decision of whether to apply IDS in a given situation, including:

- The registration accuracy required for the given task.
- The incremental costs of acquiring Data.

When registration accuracy requirements are low, manually-selected Data configurations may result in sufficient registration accuracy. When the costs of acquiring registration Data are low, the need for IDS is reduced. Increasing the quantity of registration Data collected will typically result in improved registration accuracy (assuming that the noise characteristics are stationary over the collected Data). Therefore, when Data are cheap, collecting large quantities will always result in improved registration accuracy, thus reducing the benefit of IDS.

In the ongoing assessment of the IDS method, each new situation (i.e., shape of the registration object, Data collection device, Model generation method, etc.) is being evaluated on a case-by-case basis to determine whether application of IDS is appropriate. By performing

stability and registration experiments such as those outlined in this chapter, it is possible to systematically evaluate the utility of IDS, and to better understand the influence of noise upon registration accuracy for the given situation.

The next chapter applies the IDS methods to several registration problems in a laboratory environment. In addition, it is shown that the constraint analysis method can be useful for the purpose of estimating registration accuracy during the registration process. In addition to laboratory studies, the IDS methods are currently being incorporated into a computer-assisted surgical system for use in total hip replacement surgery [DiGioia et al., 1995]. As demonstrated in the next chapter, constraint analysis and constraint synthesis appear to be useful in real-world problems.

Chapter 5

Applying Constraint Analysis and Synthesis

The previous chapters have outlined methods for performing registration, for assessing registration accuracy via geometric constraint, and for automatically synthesizing Data configurations which maximize geometric constraint. The current chapter brings all of these components together to demonstrate how accuracy can be improved on a real-world registration problem. The resulting framework encompasses all of the components outlined in Figure 5-1. The most important results presented in this chapter demonstrate that synthesized Data sets collected by a human from a physical object result in significantly better registration accuracy compared to manually-selected Data sets.

Another problem addressed in this chapter is minimizing the quantity of Data needed to satisfy a given accuracy requirement. It is shown that solving this problem requires the ability to estimate registration accuracy online, during the registration process. A method for online accuracy estimation is proposed which combines a residual error measure defined in Chapter 2, with the NAI criterion measure from constraint analysis. A second method is proposed which uses this online accuracy estimate together with Intelligent Data Selection to minimize the quantity of Data required to satisfy accuracy requirements.

The remainder of this chapter is organized as follows. Section 5.1 examines the problem of generating minimally-sized Data configurations. A method for solving this problem is presented, and variants are suggested for reducing the effects of Data collection uncertainty and sensor noise. Section 5.2 proposes a method for generating reliable online estimates of registration accuracy based upon a residual error measure and the NAI criterion measure. Section 5.3 describes the experimental setup used to perform the registration experiments reported in this chapter, and Section 5.4 presents the results of these experiments.

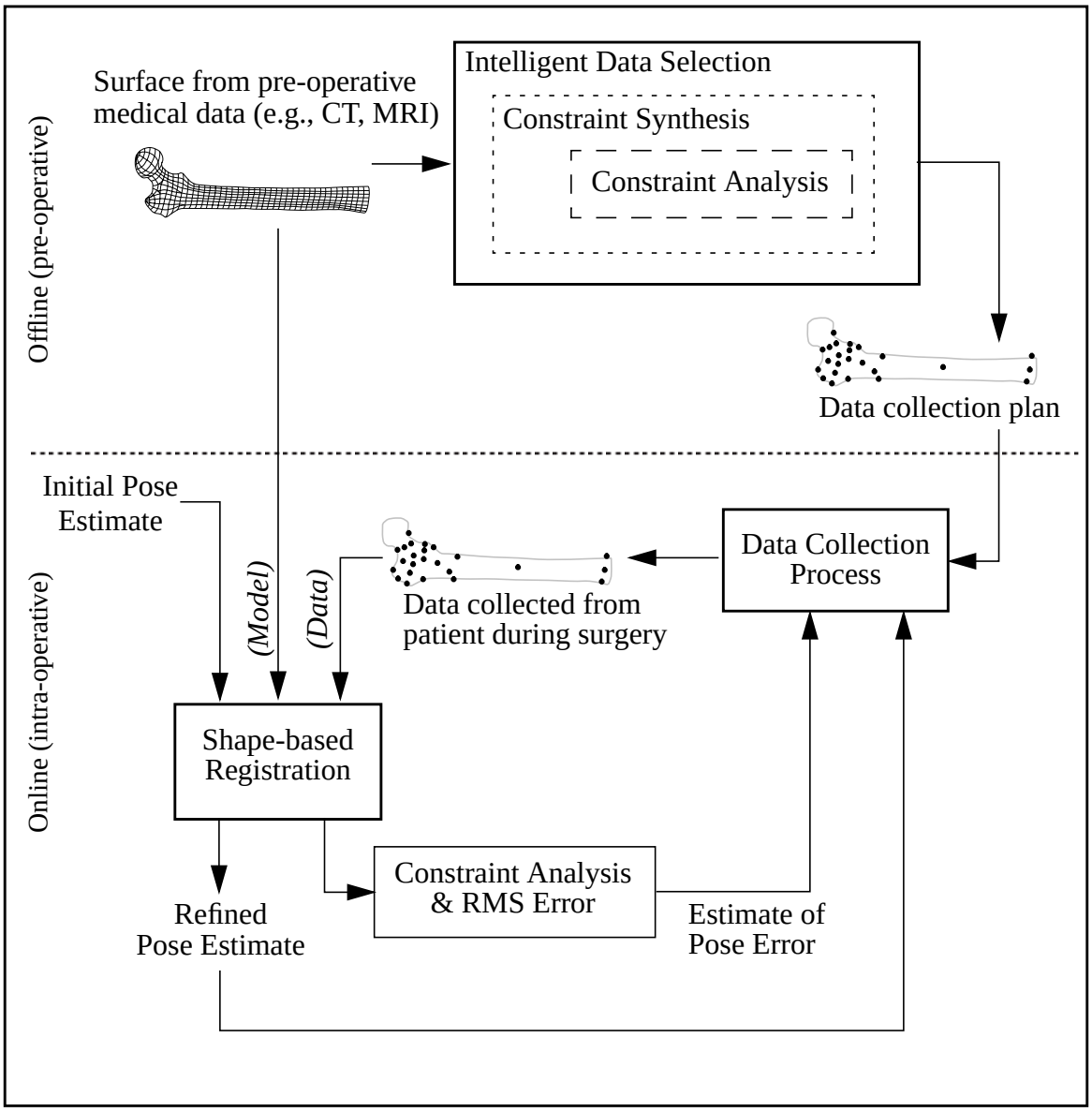


Figure 5-1: All of the components of the object localization framework are combined in this chapter to demonstrate accuracy improvements in a real-world registration problem.

5.1 Minimally-Sized Data Configurations

One motivation for using the framework outlined in Figure 5-1 is to minimize the quantity of Data required to satisfy registration accuracy requirements. As demonstrated in Sections 3.4 and 4.4, there are many factors which affect registration accuracy including: the geometry of the registration object; the quantity of registration Data; the geometric constraint provided by the Data (i.e., the NAI value); the Data collection uncertainty; and the

magnitude of noise in the Data and Model. In the experimental results reported in previous chapters, nominal values for the noise parameters were selected for each experiment (e.g., expected sensor noise magnitude, μ_Δ, Data collection uncertainty radius, r_u). Given these parameters and a particular Data configuration, it was possible to estimate the resulting registration accuracy using the experimental method outlined in Figure 3-15. Unfortunately, in practice it may be difficult or impossible to reliably estimate sensor noise and Data collection uncertainty parameters *a priori*.

For example, in the experimental results presented in this chapter, an optical digitizing probe was used for Data collection (see Section 5.3 for details). With such a probe, sensor noise is a function of several parameters including the quality of probe calibration, the probe's location within the sensor's workspace, the characteristics of the particular probe which is used, and the skill of the Data collector. Similarly, Data collection uncertainty is a function of factors such as the geometry of the registration object and the skill of the Data collector. In general, it is impossible to reliably estimate sensor noise and Data collection uncertainty *a priori*. Therefore, it is difficult to estimate registration accuracy (and thus the number of Data points required to satisfy accuracy requirements) without additional information.

Fortunately, additional information is available online, during the registration process. Recall from the discussion of Section 2.5 that there are several registration accuracy measures which do not require the availability of the known pose transformation. Based upon one of these measures, a method is presented in Section 5.2 for online estimation of the true registration accuracy. Given this online accuracy measure, it is possible to minimize the quantity of Data needed to satisfy accuracy requirements using the following approach.

Method for Achieving Minimum-Sized Data Sets ("MSDS Method")

1. Offline, before the registration process, use Intelligent Data Selection to synthesize multiple Data configurations of increasing size, each of which is a superset of the previous configuration. For example, synthesize 5 configurations P_{10}, P_{20}, P_{30}, P_{40} and P_{50}, containing 10, 20, 30, 40, and 50 points, respectively, such that the following relation holds: $P_{10} \subset P_{20} \subset P_{30} \subset P_{40} \subset P_{50}$.
2. Online, during the registration process, collect all of the points in the first configuration, P_{10}. Compute the registration transformation using these points.
3. Compute the online registration accuracy estimate and determine if accuracy requirements are satisfied. If so, EXIT.
4. Collect the points within the next configuration set (e.g., P_{i+10}) which have not already been collected. If all Data configurations have been collected, EXIT.
5. Compute the registration transformation using the new, larger set of Data. Goto Step 3.

The basis for this approach is demonstrated in the experimental results of Section 4.4. For example, Figure 4-20 shows that both the NAI and MCE increase as a function of increasing configuration size for synthesized configurations on the Femur object. Therefore, by applying the MSDS method, registration accuracy should increase monotonically with increasing configuration size. In terms of implementation, only minor modifications of constraint synthesis are necessary in order to synthesize configurations which are supersets of pre-existing configurations.

An interesting question regarding this method is the effect of the superset condition upon the NAI-optimality of a configuration. For example, assume that a configuration of 30 points, P_{30}, is synthesized by first synthesizing a configuration of 10 points, P_{10}, and then a configuration of 20 points, P_{20}, such that $P_{10} \subset P_{20} \subset P_{30}$. Now assume that a second configuration of 30 points, P_{30}^{*}, is synthesized without any conditions. How do the NAI values of the two 30 point configurations compare? Experiments to answer this class of questions have not yet been conducted. If synthesizing many intermediate configurations degrade the eventual NAI value, then the choice of the intermediate set sizes will be crucial. A particularly interesting question is how the MSDS method behaves when the point-size increment between subsequent sets is one (e.g., $P_{10} \subset P_{11} \subset P_{12} \ldots$). If the behavior of the method is reasonable in this case, it may be possible to synthesize configurations online, during the registration process in real-time.

Recall that the registration solution method described in Section 2.3 contains a step which removes outlier Data points based upon a residual threshold, ε_o (see Section 2.3.3). In some situations, points removed as outliers may be valuable with respect to geometric constraint, and removing them may significantly reduce the effective NAI value. Therefore, in Step 4 of the MSDS method, it may be desirable to re-collect points which have been removed by outlier elimination on the previous iteration.

In the MSDS method, it is possible to terminate without satisfying accuracy requirements (see Step 4). Recall that one of the most significant factors influencing registration accuracy is the magnitude of sensor noise. If the effects of sensor noise can be reduced, registration accuracy should improve. This observation suggests the following procedure for handling termination without satisfying accuracy requirements.

MSDS Method (continued)

(replace “EXIT” in Step 4 above with “goto Step 6”).

6. Reduce the outlier elimination residual threshold, ε_o, by a fixed amount, and recompute the registration transformation using all previously collected Data.

7. Compute the online measure of registration accuracy. If accuracy requirements are satisfied, EXIT.
8. Re-collect all of the Data points which were eliminated as outliers during the last registration trial (i.e., during Step 6 or Step 9).
9. Compute the registration transformation and the online accuracy estimate.
10. If accuracy requirements are satisfied, EXIT. If there were any points removed by outlier elimination in Step 9, then goto Step 8. If the residual error threshold, ε_o, is below a minimum value, $\varepsilon_{o\text{-}min}$, then EXIT. Else goto Step 6.

This procedure is based upon the assumption that points eliminated as outliers during the registration process are “noisy”, and tend to increase the average sensor noise magnitude. By eliminating and re-collecting these outliers, the newly collected replacement Data may have improved noise characteristics, thus resulting in smaller registration errors. This procedure also assumes that sensor noise results from a random process, and not a systematic (and thus repeatable) sensor error.

A final variant of the MSDS method addresses the problem of Data collection uncertainty. Recall that Data collection uncertainty results in deviations between the locations of *desired* and *collected* Data points. As explained in Section 4.3, this deviation may result in a reduction in the NAI, especially when there is a large discrepancy between the surface normals corresponding to the desired and collected points. As outlined in Figure 5-1 and demonstrated in Section 5.3, it is possible to use current estimates of the registration transformation to guide the Data collection process. Using this guidance mechanism, better transformation estimates result in reduced Data collection uncertainty. The above observations suggest a modification to the MSDS method in which points from relatively flat regions of the surface are collected first, and those in regions of high curvature are collected later. By collecting points within flat regions when the transformation estimate is poor (and Data collection uncertainty is large), it may be possible to reduce the effects of uncertainty. After Data in the flat regions are collected and used in registration, the transformation estimate should be more accurate, and thus Data collection uncertainty will be reduced. With reduced uncertainty, points from regions of higher curvature which are relatively sensitive to uncertainty can be collected more precisely.

A method for implementing the above multi-resolution Data collection is outlined in Figure 5-2. This method requires the use of the high-curvature point elimination filter described in Section 4.3. This filter eliminates regions from a Model based upon surface normal variations within the region. As a first step, high curvature regions of a Model are eliminated, and a configuration, P_1, is synthesized from the resulting filtered Model.[1] Next,

1. In this example, the subscript on a configuration P_i is an index, not the configuration size.

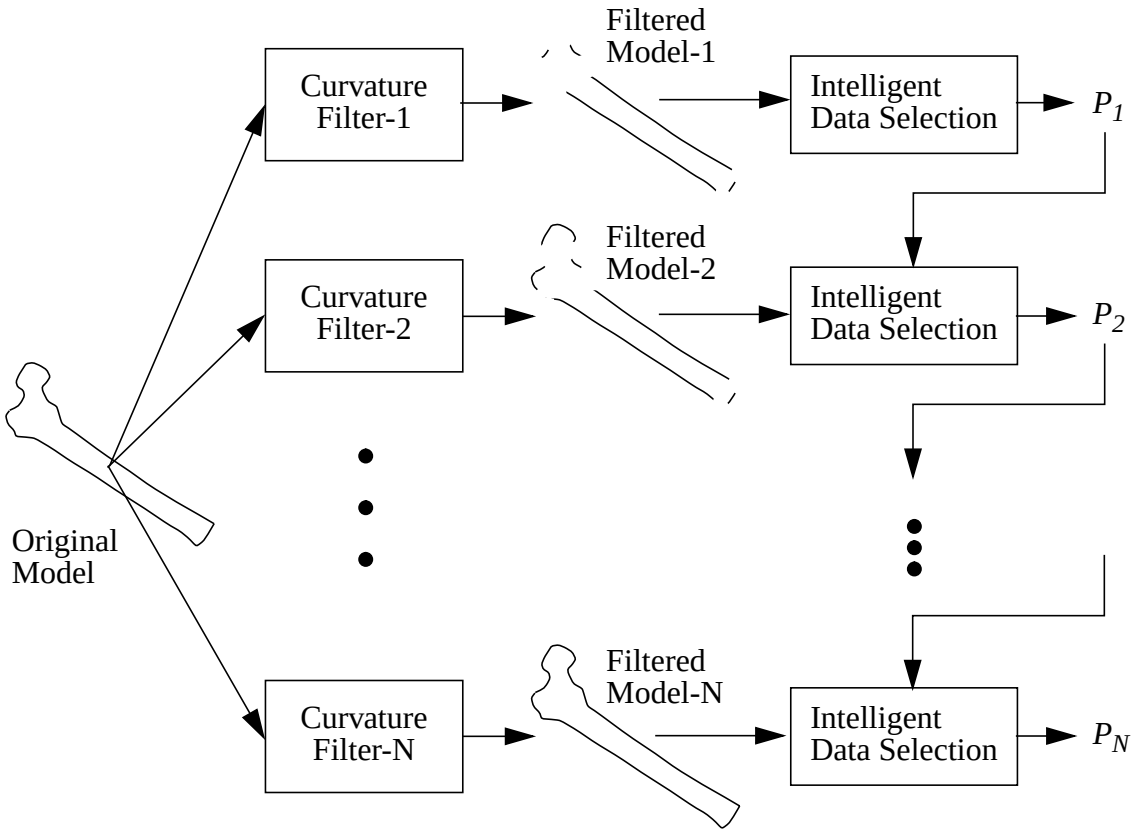


Figure 5-2: A method for minimizing Data set size and reducing the effects of Data collection uncertainty. The sets are generated such that $P_1 \subset P_2 \subset \ldots \subset P_N$.

the filter is run on the surface again with relaxed parameters, thus including higher curvature regions in the second filtered Model. A second configuration, P_2, is then synthesized such that $P_1 \subset P_2$. This process is repeated several times, increasing the included curvature and point size each time. Once these configurations have been synthesized, the original MSDS method can be applied as before.

5.2 Online Accuracy Estimation

To implement any of the methods for selecting minimally-sized Data sets described in the previous section, the ability to generate online estimates of registration accuracy is required. To be useful, an online accuracy estimate must relate a quantity which can be measured during the registration process, to a second quantity which has physical meaning to the task for which registration is being performed. The estimate of this second quantity can then be used to determine whether accuracy requirements have been satisfied. Throughout most of this dissertation, the maximum correspondence error (MCE) has been used as a generic measure

of true registration accuracy; however, other measures which are more specific to a given task may be desired, as discussed in Section 2.5.

This section outlines a method for estimating an upper bound on the true registration accuracy as a function of a residual error which can be measured online. The measure of registration accuracy used in this discussion is MCE; however, it is likely that the method can be extended to other measures of registration accuracy. It is shown that online computation of the NAI is required in the process, since the proposed method breaks down when the effective NAI value is too small. The feasibility of the proposed method is demonstrated via experimental results presented in Section 5.4.2.

Recall from the discussion of Section 2.5 that there are several error measures based upon the residuals computed during the registration process. For example, the root-mean-square error (RMS) is defined in Equation 2.11 as:

$$\text{RMS} = \sqrt{\frac{1}{N} \cdot \sum_{i}^{N} \left\| M_i - (\boldsymbol{R} D_i + \boldsymbol{T}) \right\|^2} \tag{5.1}$$

where N is the number of points in a Data configuration; each D_i is a Data point; $\boldsymbol{R}$ and $\boldsymbol{T}$ are a rotation and translation which collectively comprise the registration transformation; and each M_i is the Model point which is closest (in terms of Euclidean distance) to the corresponding transformed Data point, $\boldsymbol{R} D_i + \boldsymbol{T}$. The RMS error is a measure of the distance between the Data points and the Model. One advantage of this measure is that it can be computed using only information available during registration. The method proposed in this section is based upon the observation that there is a strong relation between RMS and MCE when both are computed at the termination of the registration process.

In Section 3.4.1, the results of several registration experiments are presented which relate MCE to effective NAI for the Femur, Pelvis and Venus Models (see Figure 3-24). The parameters used in these registration experiments are reported in Table 3-14. Using a subset of the data from the same registration experiments, the plot of Figure 5-3 was generated and shows the relationship between MCE and RMS for the Femur Model. Each datum represents a registration experiment performed using a different random configuration, sensor noise value and initial pose transformation. An important feature of this plot is that a line passing through the origin provides a good upper bound estimate of MCE as a function of RMS. In Section 5.4.2, it is shown that the slope of this upper bound line does not change as a function of sensor noise or configuration size; however, the MCE-RMS relationship does break down when the effective NAI value is near zero. This is expected since NAI values of zero reflect a singularity condition in which the MCE can increase indefinitely with no change in RMS error. Based upon these observations, the following method is proposed for online estimation of registration accuracy.

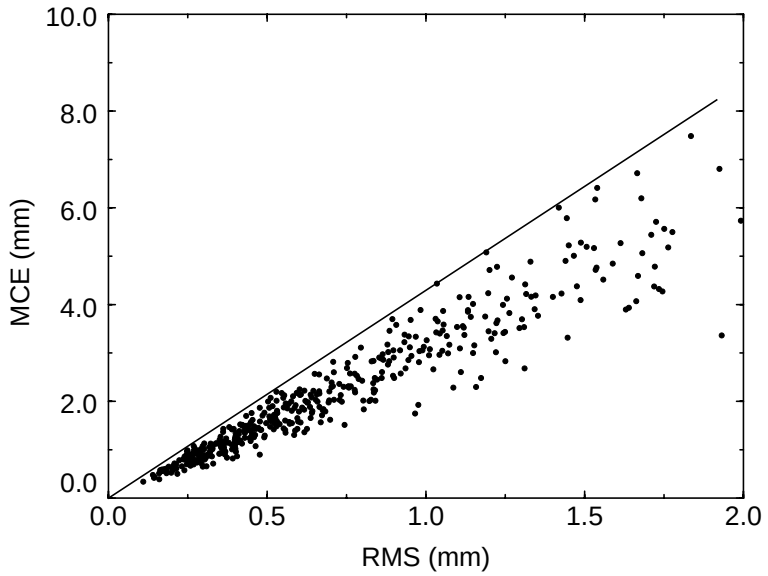


Figure 5-3: Relation between the maximum correspondence error (MCE) and root-mean-square error (RMS) for a series of registration trials using the Femur Model.

Online Accuracy Estimation Method

Offline Steps:

1. Before registration, perform a series of registration experiments such as those used to generate the plot of Figure 5-3 (see Figure 3-15 for a description of the experimental method).
2. For each registration result of Step 1, compute the effective NAI value at the termination of registration. Eliminate MCE-RMS pairs which have corresponding NAI values below a threshold, NAI_{min}.
3. Estimate the slope of the MCE-RMS upper bound line based upon the data remaining after the elimination procedure of Step 2.

Online Steps:

4. After a registration trial has terminated, compute the RMS and effective NAI values from the registration results.
5. If the effective NAI value is below NAI_{min}, it is necessary to collect additional registration Data since the relation between MCE and RMS will be unpredictable. Collect the additional Data and goto Step 4.
6. Using the online RMS measurement and the pre-computed slope of the upper bound line, compute an upper bound estimate for MCE.

The feasibility of this method is demonstrated in the experimental results of Section 5.4.2. In particular, it is shown that the slope of the upper bound line is relatively independent of sensor noise, data collection uncertainty, configuration size and NAI, assuming that effective NAI values are above the specified threshold. In addition, it is shown that estimates of the slope of the upper bound line computed via both simulated and actual registration experiments are very similar.

In Section 5.4.2, it is also shown that the slope of the upper bound line is Model dependent. Therefore, each new registration object may require offline computation of this slope. An interesting question is whether the slope of the upper-bound line must be computed for each new registration object for which online accuracy estimates are required, or whether slope estimates based on a single prototype Model can be used for each class of objects (i.e., all femurs). This question remains to be addressed in future investigation.

5.3 Experimental Setup

To evaluate the registration accuracy achieved in physical experiments, a precise estimate of the actual registration transformation must be available to use as ground-truth. The framework in Figure 5-4 uses fiducial-based registration to provide the high-accuracy ground-truth required to evaluate shape-based registration. The fiducial markers, 0.5 inch diameter

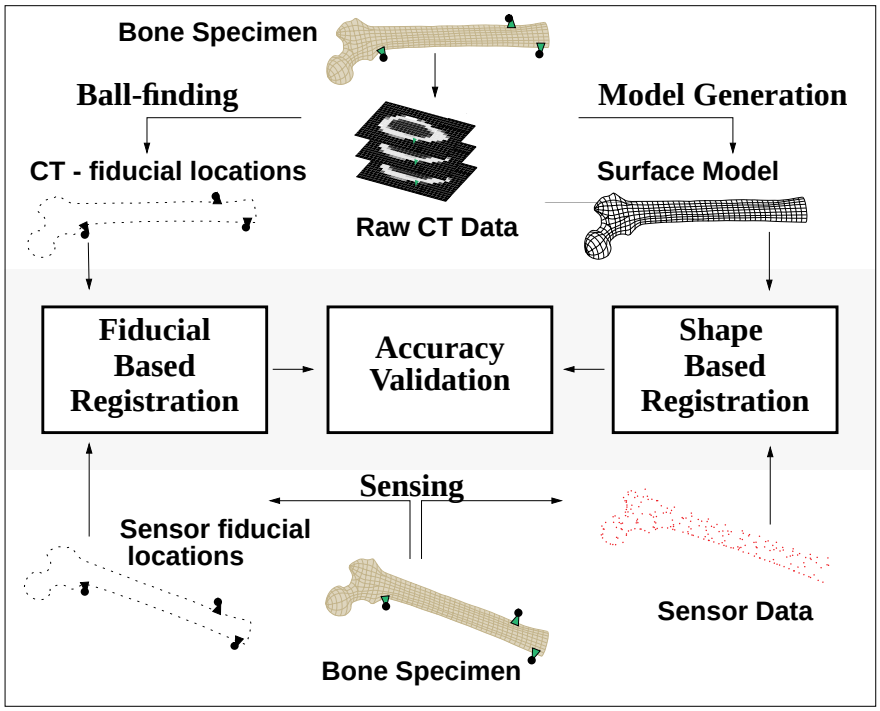


Figure 5-4: Validating shape-based registration accuracy using the result of fiducial-based registration as ground-truth.

aluminum spheres mounted on plastic stand-offs, were carefully designed to maximize the accuracy with which they could be localized in CT images and in the surgical coordinate system (e.g., using a digitizing probe). After determining the locations of the markers in the CT and surgical coordinate systems, a corresponding point registration method (see Figure 2-3) was used to establish the ground-truth registration transformation [Horn, 1987]. Validating the accuracy of the fiducial-based method is a difficult problem since a more accurate standard is required for this purpose. In the experiments described below, the average residual errors for fiducial-based registration are on the order of 0.1 mm, with similar values for the maximum residual errors. Relating these numbers to true registration error depends upon the relative positions of the fiducial markers on the registration object, and requires that certain assumptions be made regarding the source of the residual errors. Assuming a Gaussian noise model for the residual errors, it can be argued that maximum correspondence errors resulting from fiducial-based registration of the Femur object are roughly equal to the maximum residual errors (i.e., about 0.1 mm).

The registration object used in the current experiments was a human femur with attached fiducial markers as seen in Figure 5-5. The femur was CT scanned in air using a General Electric “High-Speed Advantage” clinical CT imager at a resolution of 0.29 mm x 0.29 mm x 1.0 mm with 1.0 mm slice thickness. A CT cross section of the femur which includes one of the fiducial markers is seen in Figure 5-6. The location of the fiducial markers can be automatically extracted from the CT data by intensity-thresholding the images, computing the volume and eccentricity of the resulting connected regions, identifying the regions likely to be spheres, and computing the resulting 3-D centroids. A surface Model of the femur such as the one shown in Figure 3-21 was generated from the CT images by extracting bounding contours of the femur from each CT image, and then linking these contours into a triangle mesh surface using the method described in [Geiger, 1993].

To collect the registration Data points and to estimate the centroids of the fiducial markers in surgical coordinates, the optical digitizing system shown in Figure 5-7 was used. This system is capable of determining the location of a digitizing probe’s tip to within 0.1 mm at rates of 100 Hz and faster. The two digitizing probes shown in Figure 5-8 were used to collect the Data points for the experiments. As demonstrated in Section 5.4, these two probes have very different noise characteristics. A similar probe was used to estimate fiducial marker centroids, as seen in Figure 5-9.

The Data collection process of Figure 5-1 includes a human Data collector who uses a sensor (i.e., a digitizing probe) to collect Data on the surface of the registration object based upon the Data collection plan. In order to guide the Data collector, a display such as the one shown in Figure 5-10 was used. The display contains a graphical rendering of the registration object with a semi-transparent, colored sphere centered at the location of the current desired Data point. In the absence of additional information, this display is useful for visu-

Figure 5-5: A human femur with attached spherical fiducial markers.

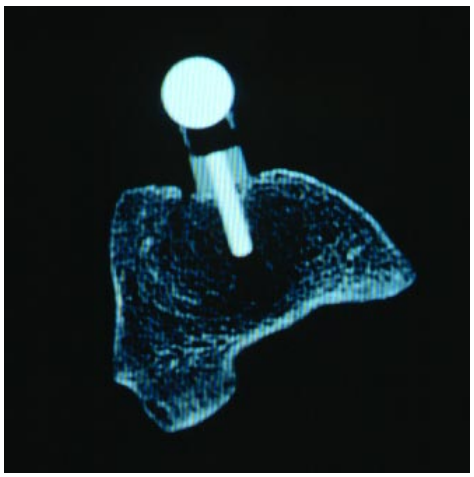


Figure 5-6: CT cross-section of the human femur and a spherical fiducial marker mounted on a plastic stand-off. Determining the centroid of the marker within the CT data set is done with a simple thresholding operation and 3-D centroid calculation.

ally communicating the location of the desired point to the Data collector. The radius of the sphere can be used to indicate the expected region of collection.

Incremental registration transformation estimates are available from the registration process after the collection of each new Data point, as indicated in Figure 5-1. Using these estimates, the location of the digitizer's probe-tip can be transformed into the Model's coordinate system and superimposed on the display of Figure 5-10. When the collector moves the

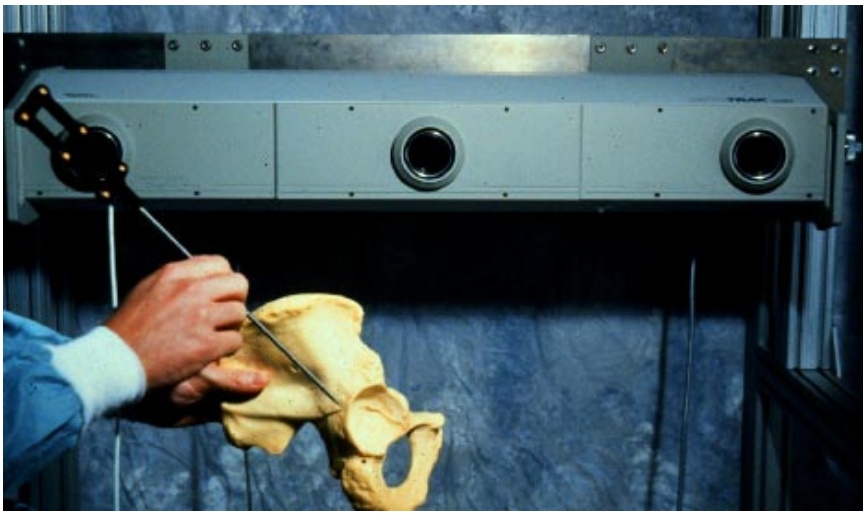
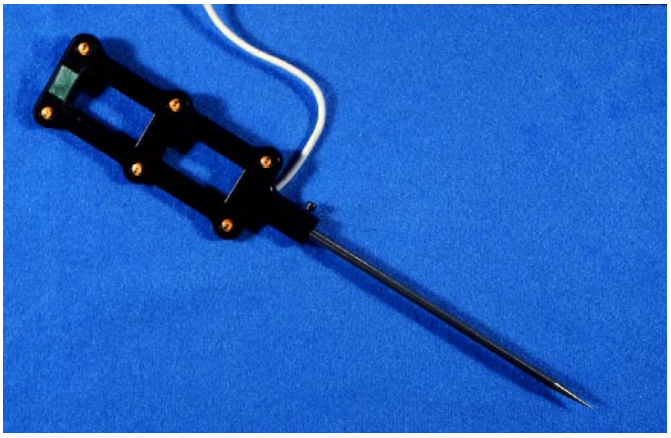


Figure 5-7: An optical digitizing system. The 3-lens camera system seen in the background determines the locations of the light-emitting-diodes (LEDs) mounted on the digitizing probe seen in the foreground. The location of the digitizing probe-tip can be determined to within 0.1 mm at rates upwards of 100 Hz.

6-Marker Probe



24-Marker Probe

Figure 5-8: The 6- and 24-marker digitizing probes.

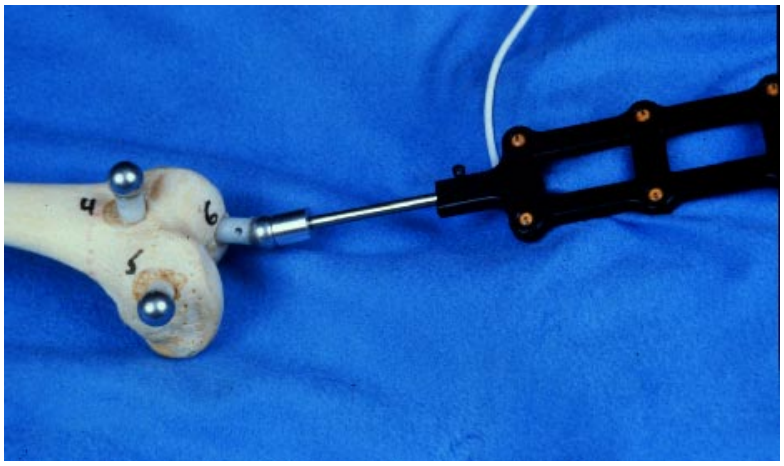


Figure 5-9: Determining the location of a fiducial marker's centroid in surgical coordinates. The above probe is calibrated such that the probe-tip's origin is coincident with the centroid of a spherical fiducial marker when the probe-tip (a hollow cylinder) is mated with a fiducial. This allows estimation of a fiducial's centroid using a single measurement.

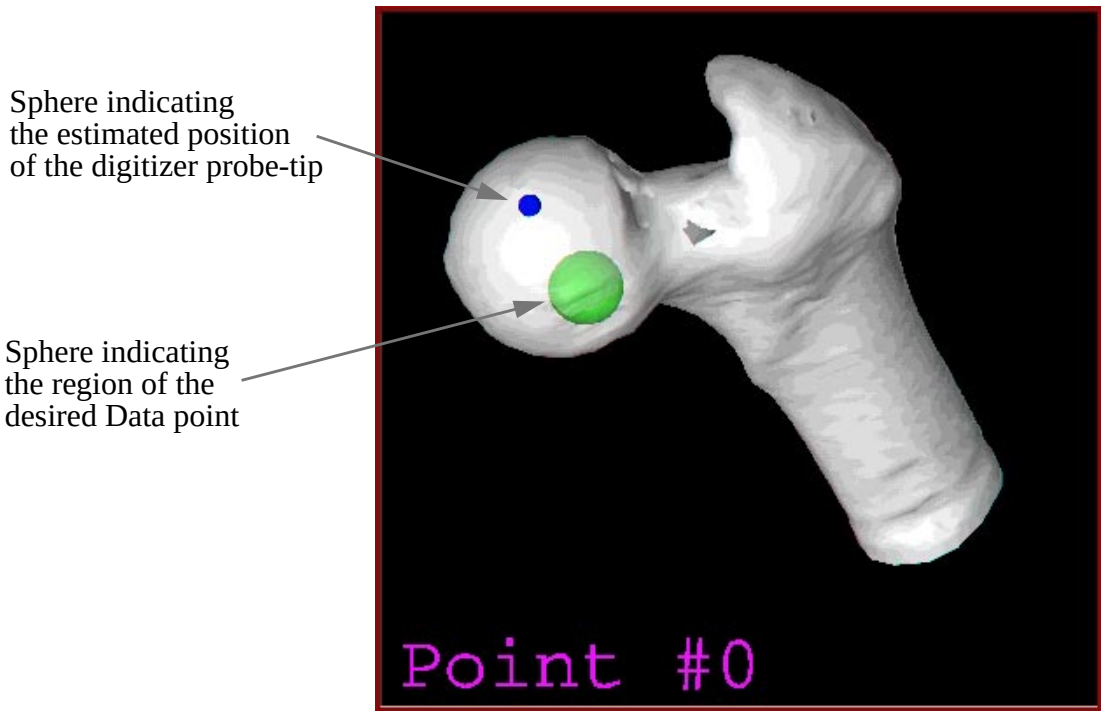


Figure 5-10: Display provided to the human Data collector which provides navigational guidance during the Data collection process. The large, semi-transparent sphere indicates the location of the desired point. The small dark sphere indicates the estimated location of the digitizer's probe-tip.

probe-tip, its location on the display is updated in real-time. The goal of the Data collector is to align the sphere representing the probe-tip position and the sphere representing the desired Data point. Each time a Data point is collected, the display automatically updates to show the next desired point. The viewing direction is also updated so that the desired point is always facing towards the viewer.

When the registration transformation estimate is poor, the sphere representing the probe-tip location can appear to be floating in space, even when the probe-tip is positioned on the object's surface. Since this effect can be disconcerting, it is possible to project the sphere representing the probe-tip location onto the closest point of the rendered surface. While this approach often leads to good results, care must be taken since it can be misleading in certain regions of the Model.

5.4 Experimental Results

The first set of results in this section presents the very encouraging outcome of applying constraint analysis and synthesis to a practical registration problem. The second set of results demonstrates the feasibility of performing accuracy estimation online, during the registration process.

5.4.1 Comparing Synthesized and Manually-selected Data Configurations in a Practical Registration Problem

The results presented in this section are perhaps the most important of the entire dissertation. They demonstrate that the Intelligent Data Selection method can significantly improve registration accuracy and reduce the need for large quantities of registration Data in real registration problems. The registration experiments used to generate these results are similar to those performed earlier in the dissertation (e.g., see Figure 3-15) with several important differences.

1. All registration Data points were manually collected by a human using the optical digitizing probe on the Femur registration object.
2. Instead of randomly generated Data configurations, manually-selected Data configurations were used.
3. No sensor or Data collection uncertainty noise was added to the Data since the Data already contain these noise components.

The experimental parameters used in these experiments are summarized in Table 5-1. Synthesized and manually-selected configurations of 6, 10, 25 and 50 points were collected using both the 6- and 24-marker digitizing probes shown in Figure 5-8. For each configuration size / digitizing probe combination, 50 manually-selected and 25 synthesized configura-

tions were collected by a human Data collector (i.e., the author). The navigational guidance display outlined in Figure 5-10 was used for collection of all synthesized configurations. The particular synthesized configurations used were the same ones reported in the “PBIL-NAH” row of Table 4-2, with a new set of configurations synthesized for the 6-point configurations. Each of the 5 synthesized configurations reported in the table was independently collected 5 times, for a total of 25 collected synthesized configurations.

Table 5-1: Registration parameters for the experiments reported in Section 5.4.1.

Parameter	Value
Data sources	Manually selected & Synthesized
Data configuration size	6, 10, 25, 50 points
Noise state	Disabled
Uncertainty noise	Disabled
pose_count	1
set_count	50 manually selected, 25 synthesized
Minima suppression	Enabled
$\tau_{p\text{-}max}$	5.0 mm
$\theta_{p\text{-}max}$	3.5 degrees
I_p	8 iterations
Outlier elimination	Enabled
ε_o (threshold)	1.5 mm
ρ_o (ratio)	0.1

Initial pose estimates for registration were derived using anatomical landmark-based corresponding point registration. Three landmark points were identified on the Femur Model (i.e., in CT coordinates), and the Data collector attempted to acquire these same three points on the Femur object using the digitizing probe (i.e., in surgical coordinates). New landmark points were collected in surgical coordinates for each registration trial; however, the same CT landmark points were used for all trials. Minima suppression and outlier elimination were both enabled during the registration trials.

The results of the femur registration experiments are shown in Figures 5-11 and 5-12 as plots of maximum correspondence error (MCE) versus the effective noise amplification

index (NAI). There are 8 graphs in the two figures, one for each configuration size / digitizing probe combination. Graphs in the same row represent experiments using the same number of Data points, while graphs in the same column represent experiments using the same digitizing probe. Within each graph, the open circles represent manually-selected configurations, and the closed circles represent synthesized configurations. The y-axes of the two graphs within each row have the same scale; however, the x-axes of the graphs within each column have different scales. It is important to recognize that the effective NAI values plotted along the x-axes are computed online, after the registration process has terminated, and can thus be used for online accuracy estimation.

Consider the two graphs in the top row of Figure 5-11 corresponding to the 6-point Data configurations. As shown at the end of Section 3.3.1, six is the theoretical minimum number of Data points required to unambiguously localize an arbitrarily-shaped object using shape-based registration. Both of these graphs demonstrate that the synthesized 6-point configurations have significantly smaller MCE values (and larger NAI values) than manually-selected configurations. In fact, the majority of manually-selected 6-point configurations resulted in very poor performance despite fairly good initial pose estimates. This is a significant result because it suggests that it is difficult for humans to select six Data points which completely constrain an arbitrarily-shaped object.[1]

In the graphs generated using the 6-point Data configurations, there are several synthesized configurations which result in relatively poor registration accuracy. This is most likely caused by the effects of large Data collection uncertainty resulting in reduction in the effective NAI compared to the ideal NAI, as explained in Section 4.3. Fortunately, using the methods of online accuracy estimation discussed in Section 5.2, configurations resulting in large MCE values can be identified during the registration process based upon the RMS error and effective NAI value. Once identified, the accuracy for such configurations can be improved via additional Data collection.

Due to physical differences between the two digitizing probes shown in Figure 5-8, the probes have very different noise characteristics. These differences are apparent in the results of Figures 5-11 and 5-12 by comparing MCE values between the two graphs of each row. Both average and maximum MCE values are smaller for each 6-marker configuration compared to the same-sized 24-marker configuration. This suggests that the 6-marker probe has smaller noise magnitudes (i.e., higher accuracy) than the 24-marker probe, an observation which is supported by the digitizer's manufacturer [Northern Digital Inc., 1992]. There are two reasons for these differences in noise magnitudes. First, the 6-marker probe has a larger separation between LEDs perpendicular to the probe's central axis than the 24-marker probe. This separation results in greater probe-tip localization sensitivity. Second, the LEDs

1. For the results presented in this section, the author performed all Data collection. A subset of these experiments was repeated using other human Data collectors with similar results.

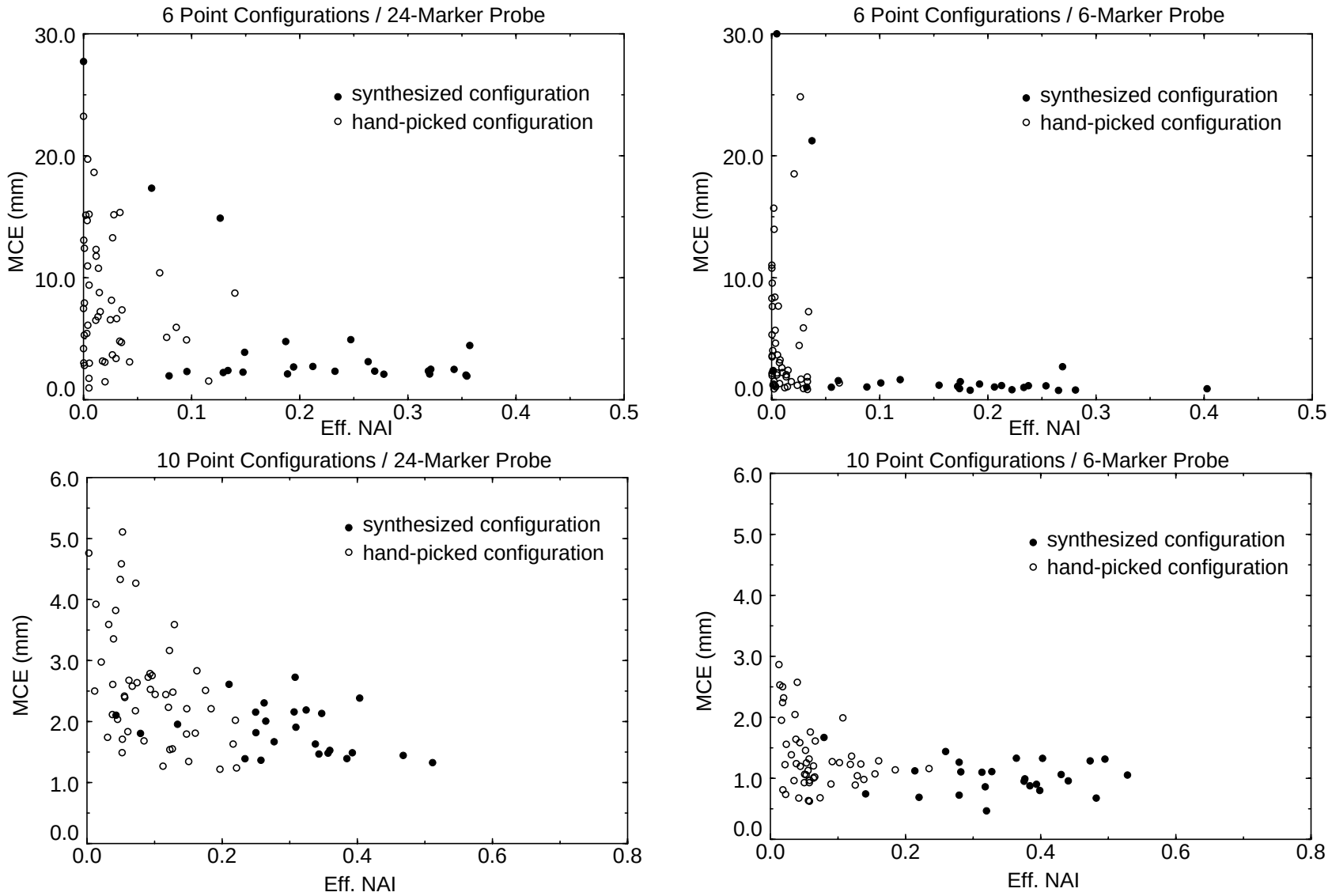


Figure 5-11: MCE versus Effective NAI for 6- and 10-point Data configurations using the 6- and 24-Marker Probes

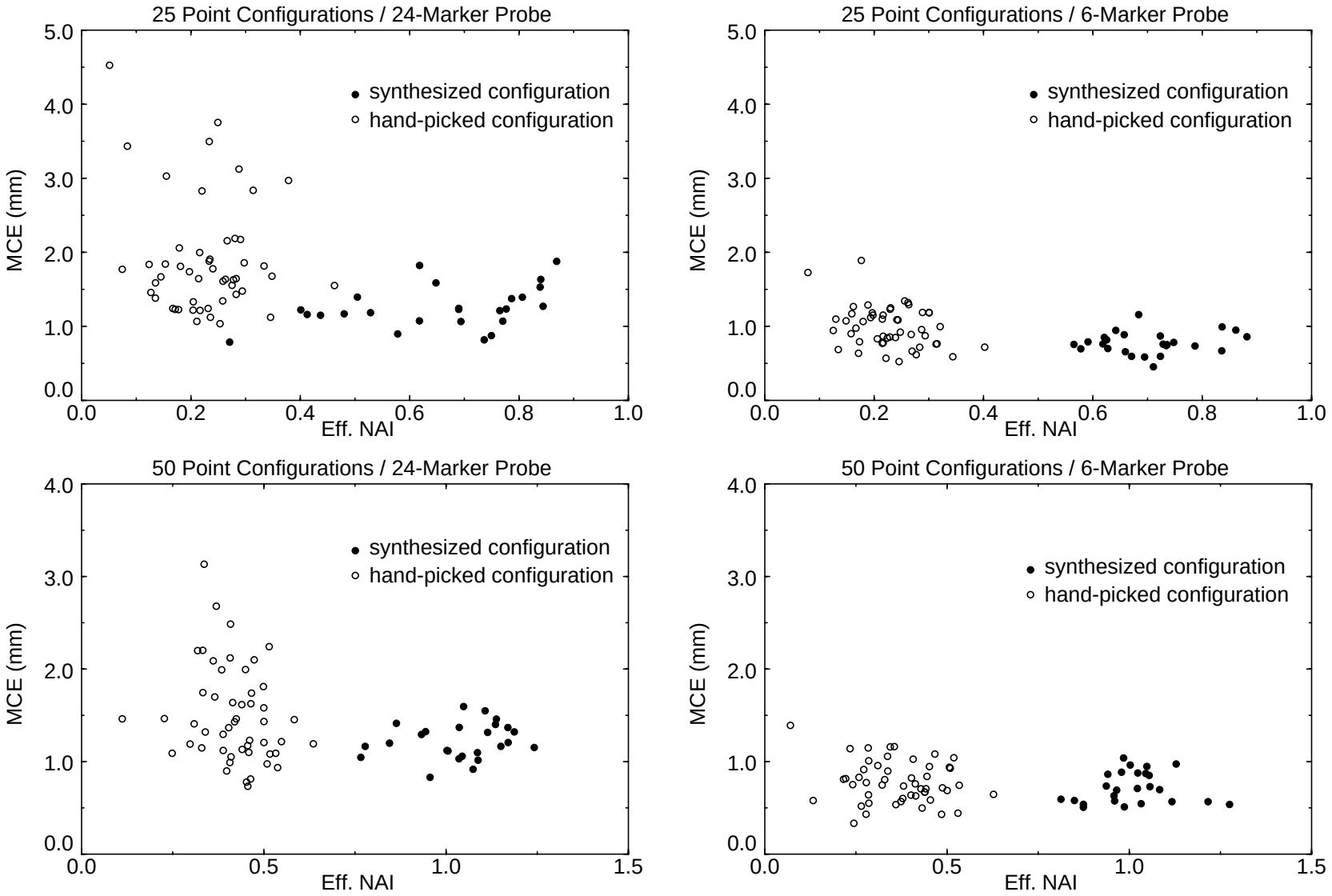


Figure 5-12: MCE versus Effective NAI for 25- and 50-point Data configurations using the 6- and 24-Marker Probes

in the 6-marker probe all point in the same direction, while the LEDs on the 24-marker probe are all oriented outward on the surface of a cylinder. This variation in LED orientation in the 24-marker probe results in apparent shifts of the LED centroids, thus decreasing probe-tip localization accuracy.

The benefit of using synthesized as opposed to manually-selected configurations is more apparent with the less accurate (i.e., 24-marker) probe, independent of configuration size. For example, using the 6-marker probe, for both the 25- and 50-point configurations, there is little apparent improvement in MCE values between the manually-selected and synthesized configurations. However, using the 24-marker probe, the advantage of the synthesized configurations is significant. This suggests that constraint synthesis is more beneficial for smaller configuration sizes and larger sensor noise magnitudes, a trend which is apparent in some of the experimental results of Chapter 4.

The results of Figures 5-11 and 5-12 also demonstrate the importance of minimizing inaccuracies in a Data collection device if high accuracy registration results are desired. This is true independent of the method used for generating Data configurations. For example, in the graphs of the 25- and 50-point configurations, random configurations using the more accurate probe result in smaller values of MCE than synthesized configurations using the less accurate probe. Whether to use a less accurate probe with constraint synthesis or a more accurate probe without constraint synthesis is a decision which will depend upon several factors including the relative costs involved.

The results of Figures 5-11 and 5-12 clearly demonstrate the benefits of Intelligent Data Selection on a practical problem. It should be noted that collecting the Data for these figures was a long, tedious process, requiring manual collection of over 15,000 Data points. Nevertheless, similar experiments must be repeated in the future using several registration objects to fully evaluate the benefits of this approach in practice.

5.4.2 Online accuracy estimation

This section demonstrates the feasibility of the online accuracy estimation method proposed in Section 5.2. The experimental results presented in this section demonstrate three key points.

1. That the upper bound lines resulting from plots of MCE versus RMS are relatively independent of sensor noise, configuration size and NAI, assuming that results with effective NAI values close to zero have been eliminated.
2. That upper bound lines estimated based upon offline registration experiments are sufficiently similar to corresponding online estimates.

3. That the slope of the upper bound line is dependent upon the shape of the registration object.

In order to demonstrate point 1 from the above list, several graphs of MCE versus RMS were plotted using registration results from the same online experiments reported in the previous section. These graphs are shown in Figures 5-13 and 5-14. Each graph shows the relationship between MCE and RMS for a different subset of the experimental results of Section 5.4.1. The four graphs of Figure 5-13 correspond to the experiments performed with Data configuration sizes of 6, 10, 25 and 50 points. The four graphs in Figure 5-14 correspond to experiments performed using the 6- and 24-Marker digitizing probes, and to trials which resulted in small effective NAI values ($0.1 < NAI < 0.5$) and large effective NAI values ($0.5 < NAI$). Each datum plotted within the 8 graphs represents a single registration experiment with the indicated values of MCE and RMS. The results of registration trials with effective NAI values less than 0.1 were not included in any of the graphs. The upper bound lines are drawn with a solid diagonal line, and the slopes of these lines are indicated. The significance of the dotted diagonal lines will be explained below. The important feature of these graphs is that the slope of the upper bound lines are very similar, ranging from a low value of 3.7 to a high value of 4.1. This suggests that the MCE-RMS upper bound relationship is relatively independent of configuration size, sensor noise and NAI value (for effective NAI values > 0.1).

In order to apply the upper bound relation demonstrated above to the problem of online accuracy estimation, it is necessary to determine the slope of the upper bound offline, before registration is performed. Once the slope of this line is known, it can be used online to estimate an upper bound on MCE given a measured value of RMS. The next set of results demonstrates that it is possible to estimate the slope of the upper bound line based upon offline registration experiments. The parameters used in these offline experiments are presented in Table 5-2. The registration parameters are identical to those used in the online experiments of Section 5.4.1. Experiments were performed using synthesized and random configurations containing 6, 10, 25 and 50 Data points. The synthesized configurations are identical to those which were physically collected in the experiments of the previous section. Sensor noise with expected magnitudes of 0.5 mm, 1.0 mm and 2.0 mm was added to the Data, and collection uncertainty noise with a bound of 5.0 mm was also added. There were a total of 250 random configurations evaluated at 2 initial poses each, and 5 synthesized poses evaluated at 100 initial poses each.

Figures 5-15 and 5-16 contain 8 graphs of MCE versus RMS for this set of offline experiments. These 8 graphs are similar to those presented in Figures 5-13 and 5-14. The slopes of the upper bound lines in each of these graphs are similar to each other, with slopes ranging from 4.3 to 4.5. Once again, this result suggests the independence of the MCE-RMS relation as a function of configuration size, NAI value, and noise magnitude.

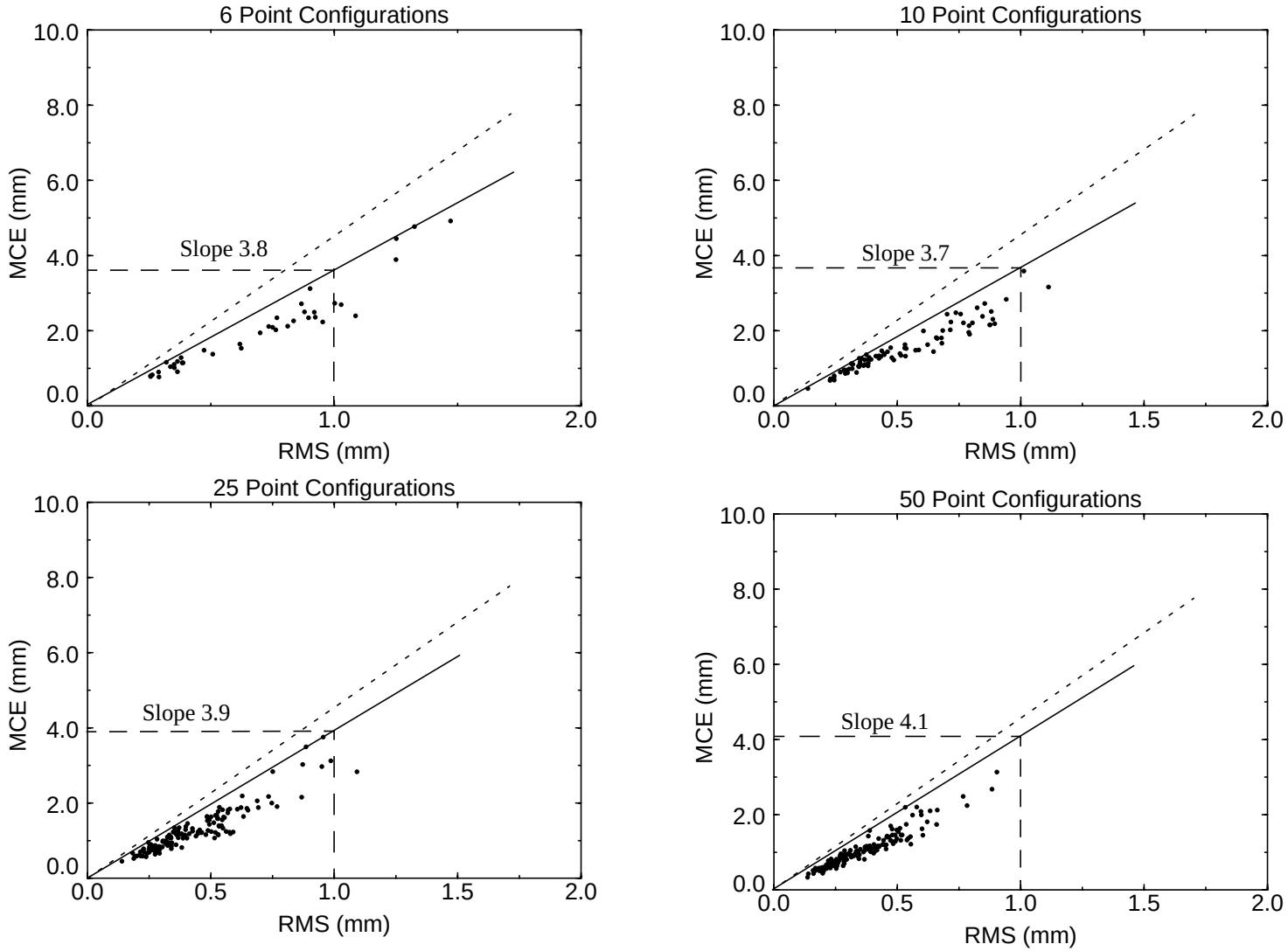


Figure 5-13: MCE versus RMS error for the online registration experiments of Section 5.4.1 using the Femur object. The four graphs contain subsets of the results based on configuration size. Slopes of the upper bound lines are indicated.

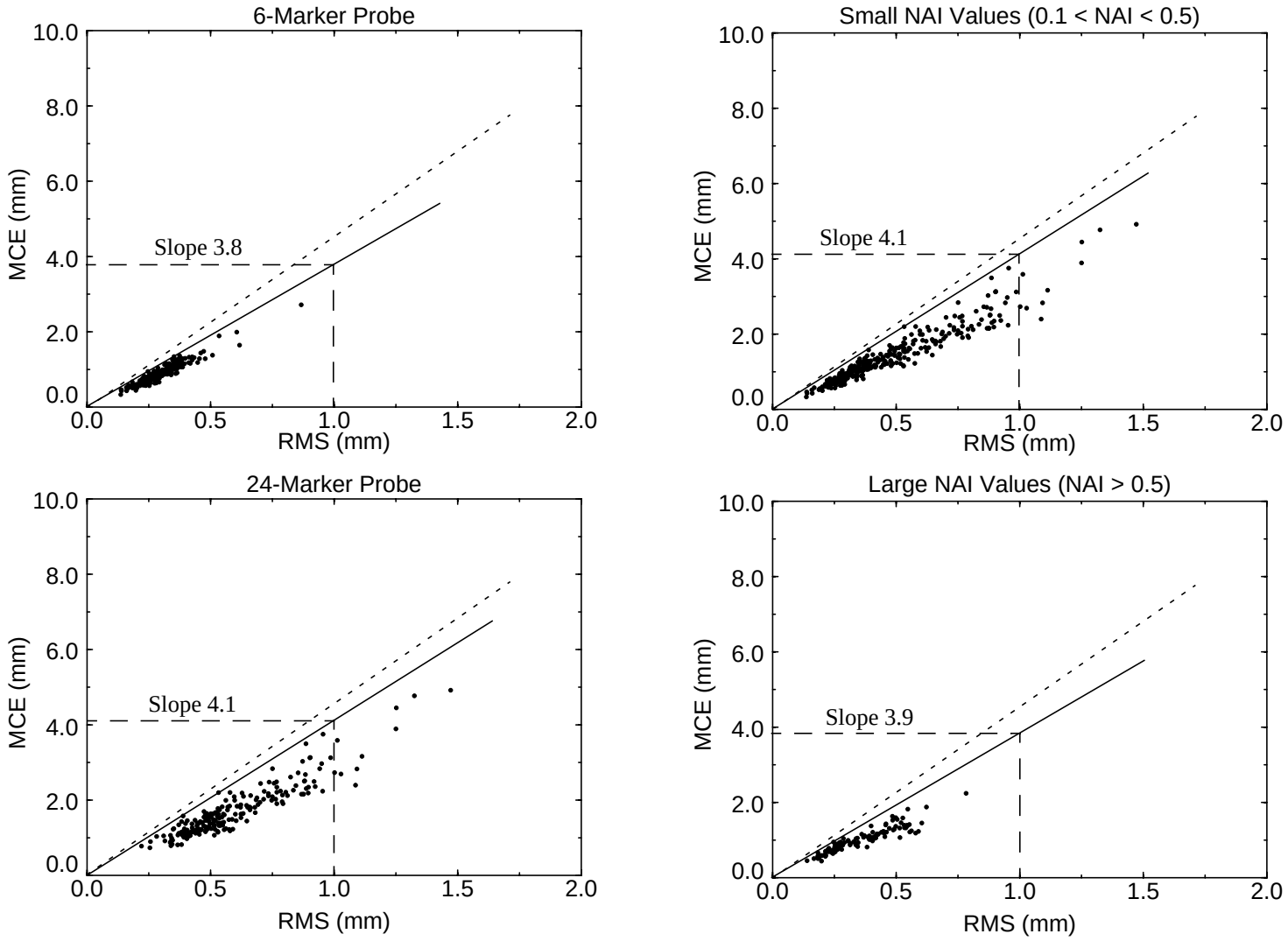


Figure 5-14: MCE versus RMS error for the online registration experiments of Section 5.4.1 using the Femur object. The four graphs contain subsets of the results based on digitizing probe and NAI value. Slopes of the upper bound lines are indicated.

Table 5-2: Registration parameters for experiments reported in Section 5.4.2.

Parameter	Value
Data sources	Random & Synthesized
Data configuration size	6, 10, 25, 50 points
Noise state	Enabled
μ_Δ - expected noise magnitude	0.5 mm, 1.0 mm, 2.0 mm
Uncertainty noise	Enabled
r_u - uncertainty radius	5.0
pose_count	2 for random, 100 for synthesized
set_count	250 random, 5 synthesized
Minima suppression	Enabled
$\tau_{p\text{-}max}$	5.0 mm
$\theta_{p\text{-}max}$	3.5 degrees
I_p	8 iterations
Outlier elimination	Enabled
ε_o (threshold)	1.5 mm
ρ_o (ratio)	0.1

Comparing the slopes of the upper bound lines from the offline (low: 4.3, high: 4.5) and online (low: 3.7, high: 4.1) experiments, it appears that the offline experiments have consistently higher slopes. This is probably the result of the much larger number of experiments performed in the offline experiments. The slope of the upper bound line is determined by the maximum value of the MCE/RMS ratio over all experimental trials. As the number of experiments increases, it is expected that this ratio will also increase as the “tails” of the distribution fill out. Therefore, it is likely that the slopes computed in the offline experiments are more realistic estimates of the true upper bound slopes than those computed in the online experiments. It is expected that if the number of online experiments were significantly increased, the corresponding slope of the upper bound line would approach the high value of 4.5 estimated by the offline experiments. For reference purposes, lines with a slope of 4.5 were added to all of the graphs in the online results of Figures 5-13 and 5-14 (dotted diagonal lines in the graphs). In all of these graphs, and especially for small values of MCE, the

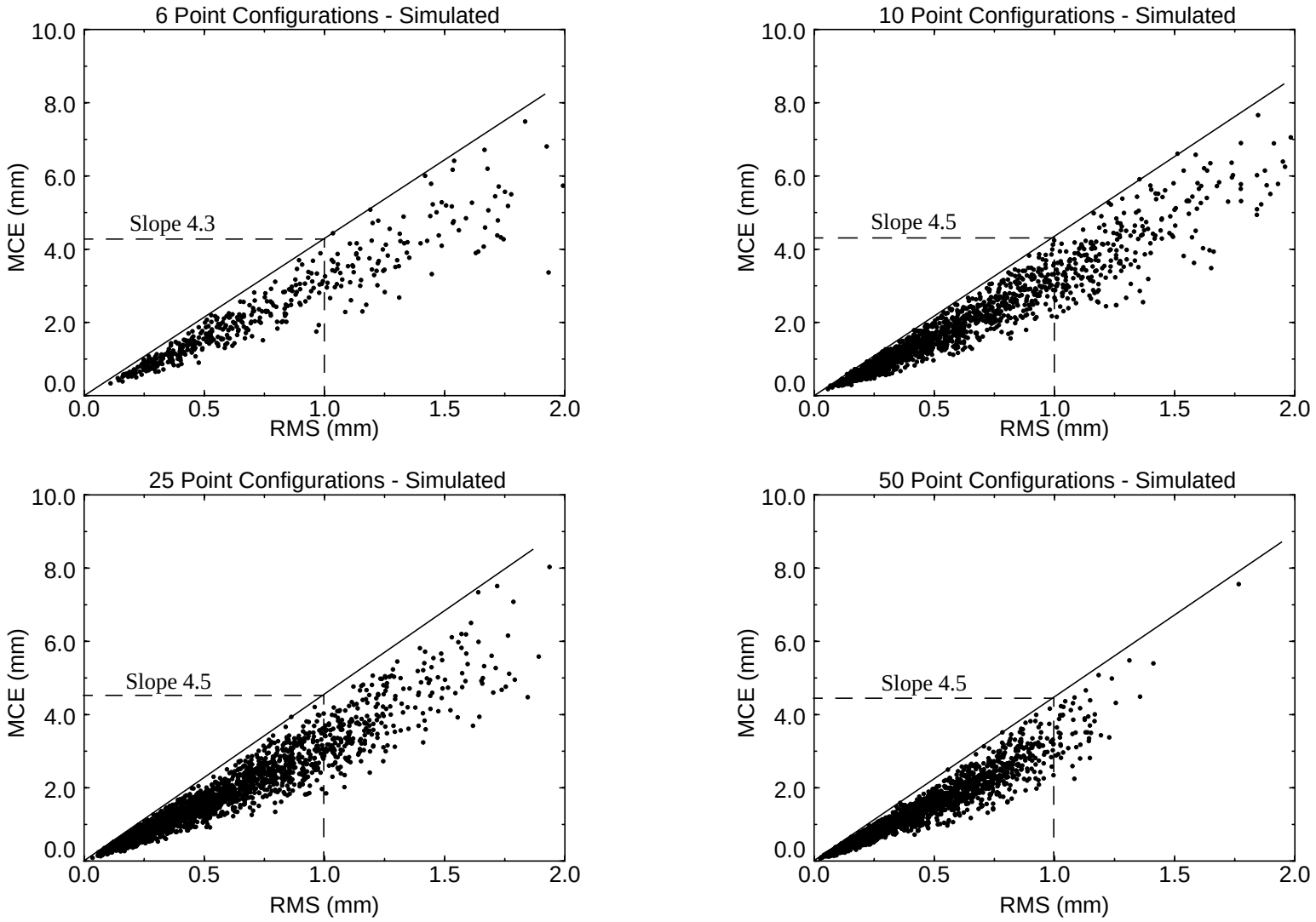


Figure 5-15: MCE versus RMS error for offline registration experiments using parameters similar to those of Section 5.4.1. The four graphs contain subsets of the results based on configuration size. Slopes of the upper bound lines are indicated.

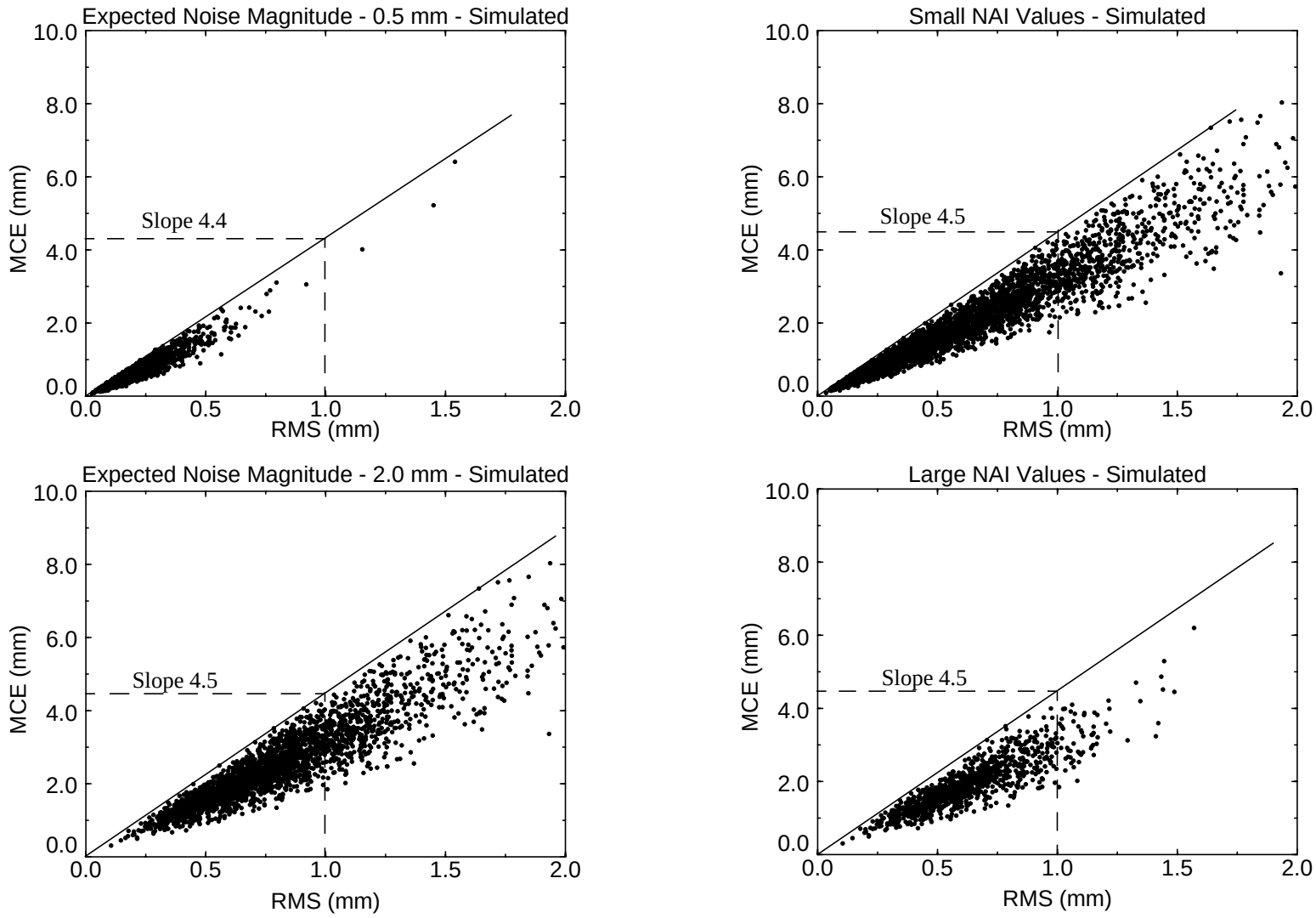


Figure 5-16: MCE versus RMS error for offline registration experiments using parameters similar to those of Section 5.4.1. The four graphs contain subsets of the results based on expected noise and NAI values. Slopes of the upper bound lines are shown.

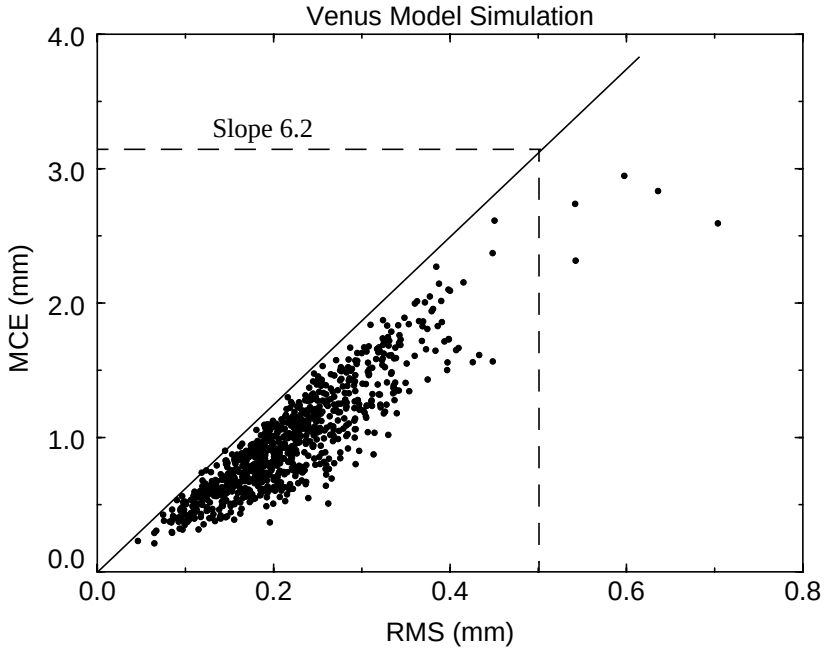


Figure 5-17: The MCE versus RMS error for the Venus Model exhibits a different upper bound line slope than the previous results for the Femur Model.

effect of the differences between online and offline slope estimates for the purposes of online accuracy estimation is minimal.

As mentioned, the upper bound relation between MCE and RMS breaks down for very small values of the NAI. This is expected since NAI values of zero reflect a singularity condition in which the MCE can increase indefinitely with no change in RMS error. Initially, it was expected that the slope of the upper bound line would be a function of NAI for large values of NAI as well; however, experimental results have not demonstrated this relation. In the experimental results presented in this section, all registration trials with effective NAI values below 0.1 were eliminated. This threshold was determined empirically. Further study is required to determine how this threshold should be selected on an object specific basis.

The final experimental result of this section demonstrates that the slope of the upper bound line depends upon the registration object which is used. In Figure 5-17, MCE is plotted versus RMS error for the Venus object based upon offline registration experiments. As seen in the graph, the slope of the upper bound line is 6.2, which is significantly larger than the slope estimates of roughly 4.5 for the Femur object. Additional study of the factors which affect this slope is warranted. In particular, it will be interesting to determine the sensitivity of the upper bound line slope to the shape of the object. If the sensitivity is small, it may be possible to estimate the upper bound line slopes for entire classes of objects (i.e., all Faces or all Femurs) so that offline slope estimation would not be required for each new Model.

5.5 Discussion

This chapter has presented important results which suggest that the object localization framework of Figure 5-1 can be used in practice to improve the accuracy of shape-based registration. Despite these encouraging results, there are several questions which remain to be answered regarding the practicality of the proposed methods. These questions are currently being studied by applying the object localization framework to a problem in computer-assisted orthopaedic surgery [DiGioia et al., 1995]. Several of the proposed registration methods have already been used in human cadaver studies, and clinical patient trials are scheduled to begin within the next several months.

The experimental results of Section 5.4.1 demonstrate that it is possible to use Intelligent Data Selection to improve registration accuracy in a real registration problem. Whether or not this method will be useful in practice depends upon several factors including the noise characteristics of the Data collection sensor, the registration accuracy requirements, the costs associated with Data collection (i.e., time, anatomical exposure), and the difficulty of acquiring Data based upon a specified Data collection plan (i.e., using the interface shown in Figure 5-10). The need for the Intelligent Data Selection framework is more important when accuracy requirements are high, expected sensor noise magnitudes are large, or the physical act of collecting Data is time consuming. If large quantities of high quality Data can be acquired in minimal time, then the benefits of Intelligent Data Selection may be reduced. In such situations, it may be possible to manually select Data points, and use online accuracy estimation to determine if requirements are satisfied. If requirements are not satisfied, the Data collector can be prompted to manually select additional Data.

The proposed method for estimating registration accuracy online appears to be quite useful. Several questions remain to be answered before this method can be widely used in practice.

1. Is the slope of the upper bound line dependent upon the particular Model used? Will a change in the resolution of the Model affect the slope? Will small changes in the shape of the Model affect the slope?
2. Is the slope of the upper bound line sensitive to registration parameters such as the outlier elimination threshold, ε_o?
3. The results of Figures 5-13 through 5-16 show that offline estimates of the slope of the upper bound line agree with online estimates. Will this similarity between online and offline estimates hold for other object shapes?

The proposed method for achieving minimally-sized Data sets (MSDS method) may also be useful in practice. Data collection efficiency will play an important role in determining when this method is practical. If it is difficult and time consuming to collect specified Data points from a Data collection plan, it is less likely that the proposed MSDS method will be practi-

cal. However, for certain applications in which automated Data collection is possible (e.g., using a robot or a numerically controlled machine), collecting a specific Data point may be no more difficult than collecting a random one. In such applications, the MSDS method is likely to be quite beneficial.

Chapter 6

Conclusion

This dissertation presents techniques for improving the speed and accuracy of shape-based registration. The goal of fast registration was achieved via a high-speed registration solver which increases registration speed by a factor of nearly two orders of magnitude compared to conventional methods. The capabilities of this solver were demonstrated via a system for tracking the position and orientation of arbitrarily-shaped objects at rates of nearly 10 Hz using range data from a high-speed sensor developed at Carnegie Mellon University.

The goal of accurate shape-based registration was achieved via two mechanisms: Intelligent Data Selection and online accuracy estimation. Intelligent Data Selection is based on the observation that all Data are not equally beneficial for performing registration accurately. Judicious selection and careful collection of a limited amount of Data can result in better registration accuracy than random use of larger amounts of Data. Benefits of the Intelligent Data Selection method were demonstrated on several practical registration problems.

Estimating registration accuracy online, during the registration process, is a difficult problem which has not received much attention in the literature. This dissertation proposes and demonstrates a method which can be used to establish an upper bound between a conventional root-mean-squared error, and a correspondence error which is a measure of true registration accuracy. Given this upper bound relation, it is possible to estimate worst case registration errors based on online measurements of the root-mean-squared error. These worst case estimates can then be used to determine whether accuracy requirements have been satisfied.

Most of the results of this dissertation have value to a wide class of problems despite the specialized nature of the application area used in the examples (i.e., surgical registration). Many of the techniques, analyses and experimental results contribute to the general knowledge of shape-based registration, and to the role of shape in the registration process. The remainder of this chapter summarizes the contributions of this dissertation, and proposes directions for future research.

6.1 Contributions

The most significant contributions of this dissertation are the development and demonstration of a new set of tools for analyzing, understanding and reasoning about accuracy in shape-based registration. These tools include constraint analysis and synthesis, a method for achieving minimum-sized Data sets, and a method for estimating registration accuracy online. These tools should be useful to other researchers or system developers for exploring novel methods of shape-based registration, or for studying fundamental issues involving object shape and the registration process.

Constraint Analysis

While previous researchers have suggested methods similar to the proposed constraint analysis method, the presentation in this dissertation is significant for several reasons.

1. The important problem of scale dependence in constraint analysis is identified, the mechanism of this dependence is illustrated, and a solution is proposed. It is shown that variations in object scale affect the relative contributions of rotational and translational motions in the resulting constraint analysis. The proposed solution is to normalize an object's scale so that the average radius vector has a magnitude of unity when measured about the centroid. It is shown that this normalization will cause translations and rotations to be treated equivalently in the analysis on average, and will provide a method for ensuring repeatable results despite changes in object scale.
2. The equally important problem of coordinate system dependence in constraint analysis is identified and studied, and a solution for this dependence is proposed. It is shown that varying the location of the origin of the constraint analysis coordinate system affects the sensitivity of constraint analysis. It is argued that an origin location which results in the smallest rate of change in least-squared energy for a rotation of the Data relative to the Model is preferred for maximizing constraint analysis sensitivity. It is demonstrated that this condition can be achieved by placing the origin at a location which is a function of the Model's surface normals and surface points.
3. A strong connection is demonstrated between the criterion measure provided by constraint analysis (i.e., the noise amplification index, or NAI) and the accuracy resulting from experimental registration trials. The effect of registration Data noise upon the resulting accuracy is clearly illustrated, and it is shown that the NAI is a good predictor of the sensitivity to Data noise which a given Data configuration will exhibit (as measured by registration accuracy).

Constraint Synthesis

Based upon the criterion measure from constraint analysis, the constraint synthesis problem is defined and several solution methods are explored. It is shown that constraint synthesis requires the solution of a high-dimensionality optimization problem in which the goal is to select a Data configuration which maximizes the NAI criterion measure. Several optimization methods including two hillclimbing algorithms and a variant of genetic algorithms (Population-Based Incremental Learning, or PBIL) were applied to the constraint synthesis problem. It is shown that a hybrid PBIL-hillclimbing approach results in superior performance in terms of maximizing the NAI; however, one of the hillclimbing algorithms performed almost as well, and had significantly faster execution times.

The effect of uncertainty in the Data collection process and the resulting implications on Intelligent Data Selection are described and illustrated via experimental results. An important observation is that constraint analysis can be performed using either the output of constraint synthesis (i.e., a Data collection plan or DCP) resulting in the *ideal* NAI, or using the registration Data which is actually collected resulting in the *effective* NAI. It is demonstrated that Data collection uncertainty tends to reduce the value of the effective NAI relative to the ideal NAI. The ratio of effective to ideal NAI provides a measure of the NAI stability of a given Data configuration with respect to Data collection uncertainty. Ideally, configurations generated by constraint synthesis should maximize both the ideal NAI value as well as the expected value of NAI stability. Unfortunately, it is shown that incorporating the NAI stability criterion into the constraint synthesis problem results in exponential complexity. An alternative method is proposed for maximizing expected NAI stability based upon the elimination of high curvature regions from a Model before constraint synthesis is applied. It is experimentally demonstrated that this method improves the NAI stability for certain objects.

Results are presented based upon both simulated and actual registration experiments which demonstrate that the Intelligent Data Selection framework can be useful for improving registration accuracy. It is shown that several factors influence the benefit of Intelligent Data Selection compared to manual or random Data selection including: the magnitude of noise in the Data collection process; the magnitude of uncertainty in the Data collection process; the incremental cost of acquiring registration Data; and the number of Data points which can be “afforded”. In the registration experiments performed using real Data, the accuracy improvements resulting from Intelligent Data Selection varied between a factor of roughly five (for 6-point configurations with low noise) to almost no improvement (for 50-point configurations with low noise).

Measures of Registration Accuracy

Several registration accuracy measures are defined and their strengths and limitations are noted. Accuracy measures are divided into two categories based upon whether a ground-

truth transformation (i.e., the true registration transformation) is available or not. Since ground-truth transformations are typically not available during the registration process, ground-truth-based measures are most useful in the evaluation and development phase of a registration method, and as a means for specifying registration accuracy requirements.

It is shown that there is a fundamental ambiguity which arises from certain common accuracy measures that represent registration error in terms of separate translation and rotation components. This ambiguity arises due to a dependence between the magnitude of the translation components and the location of the coordinate system about which the error is expressed. An example is presented in which a commonly used translation error measure is shown to vary by a factor of 10 depending upon the choice of coordinate system, despite the fact that the physical transformation is the same in all cases. This example emphasizes the importance of careful definition of accuracy measures and related coordinate systems.

A class of ground-truth-based accuracy measures referred to as *correspondence error measures* are proposed which do not suffer from the above dependence upon coordinate system. A proof is presented which shows that the maximum correspondence error (MCE) when computed over the surface of an object provides an upper bound on maximal point displacement both on the object's surface and within its entire volume. The MCE is used as a generic measure of registration accuracy throughout most of the dissertation.

In the context of registration accuracy requirements, it is important that *task-specific* measures of registration accuracy be used. Task-specific measures express the effect of registration inaccuracies in terms which can be physically related to a particular task for which registration is being performed. An example is presented in the context of orthopaedic surgery in which the task is to place a prosthetic implant within a bone. It is argued that registration accuracy requirements should be expressed in terms of the implant misalignment which would result as a function of registration error.

Selection of Minimum-Sized Data Sets and Online Accuracy Estimation

The Intelligent Data Selection process, comprised of constraint analysis and synthesis, requires that the number of Data points be specified as an input. Minimizing the quantity of registration Data needed to satisfy a given accuracy requirement is an important problem, especially when the incremental cost of Data collection is high. A method is presented for achieving this goal which requires computation of multiple, near-optimal Data configurations via Intelligent Data Selection, each of which is a superset of its predecessors. An integral part of this method is the ability to estimate registration accuracy online, during the registration process. Each time additional Data are collected, the expected registration accuracy is computed, and the resulting accuracy estimate is used to determine whether accuracy requirements are satisfied. If not, additional Data can be collected. Experimental results are presented which demonstrate the feasibility of the minimum-sized Data set method.

To be useful, an online estimate of registration accuracy must relate a quantity which can be measured during the registration process, to a second quantity which has physical meaning to the task for which registration is being performed. A method is proposed which relates a conventional root-mean-squared (RMS) registration error, to a second ground-truth-based measure of registration accuracy. For the purposes of demonstrating this method, the maximum correspondence error (MCE) is used; however, other task-specific measures of registration accuracy could be substituted. It is shown that a linear relation exists between the RMS error and an upper bound on the MCE. Furthermore, it is demonstrated that the parameters of this upper bound relation can be computed offline, before registration. Given an RMS error value computed at the termination of the registration process, the pre-computed upper bound relation can be used to estimate a worst-case MCE value. It is shown that this upper bound relation breaks down when the NAI from constraint analysis is very close to zero. This is expected since NAI values of zero reflect a singularity condition in which the MCE can increase indefinitely with no change in RMS error. To account for this, the effective NAI value is computed at the termination of the registration process as well. If the resulting NAI value is sufficiently close to zero, additional Data must be collected since it becomes impossible to estimate any measure of true registration accuracy based upon the RMS error.

Registration Solution Methods

In the area of registration solution methods, the most significant contribution is the demonstration that shape-based registration can be performed at much higher speeds than other methods previously reported in the literature. Results are presented which demonstrate that the speed-enhanced iterative closest point (EICP) algorithm developed for this work is roughly two orders of magnitude faster than the un-enhanced version. The capabilities of this algorithm are demonstrated in a system for tracking the position and orientation of arbitrarily-shaped objects at speeds approaching 10 Hz using a high-speed range sensor developed at Carnegie Mellon University.

Miscellaneous

In the process of conducting the experiments for this dissertation, a variety of software and hardware tools were constructed which stand as contributions on their own. Among these are the tools for acquiring highly accurate registration ground-truth transformations via fiducial-based registration. Design of a fiducial-based registration system capable of achieving the error tolerances required for use as a ground-truth is a non-trivial engineering problem. Special care was taken in the design of the fiducial markers themselves, as well as the methods for locating these markers within the computed tomographic (CT) and surgical coordinate systems.

Development of the infrastructure required to perform the hundreds of thousands of registration trials required to generate the results reported in this dissertation was a non-trivial endeavor. Software was written to automate much of the experimental process. Experiments were often distributed over 30 or more computer workstations, and software systems were developed to automatically invoke the experiments and record the results. Analysis of the data was also semi-automated, and software was written to automatically convert the voluminous output of a single registration experiment to the various graphs presented in the dissertation. Performing the reported experiments would have been impossible without distributing the computation over multiple workstations, and without the automated infrastructure for collecting and analyzing the results.

A navigational display mechanism was implemented to guide a human Data collector in the process of collecting the particular Data points specified by a Data collection plan. This mechanism uses the best estimate of the registration transformation to map the current location of the Data collection sensor into the Model's coordinate system. As additional Data are collected, the registration transformation estimate should improve. This mechanism will be evaluated shortly as the Intelligent Data Selection method is used in upcoming clinical trials of a computer-assisted orthopaedic surgical system.

6.2 Future Work

The registration and accuracy methods described in this dissertation have assumed that the Model is a triangle-mesh surface, while the Data are composed of discrete points. Many of the methods presented in this dissertation can be generalized to other representations. For example, a recently popular method of shape-based registration in computer-assisted surgery matches surface models from pre-operative CT or MRI images to bounding contours extracted from intra-operative X-rays. In this framework, it is desirable to minimize the number of X-rays needed to satisfy accuracy requirements, and thus reduce the patient's exposure to radiation. The Intelligent Data Selection method can be applied to this problem via modifications to constraint synthesis. The goal of constraint synthesis in this case would be to select two or more bounding contours from the surface Model which maximize the NAI. Rather than evaluating the NAI over sets of discrete points, the NAI would be evaluated over closed contours on the Model (which could be approximated via sets of discrete points which lie along closed contours). The resulting Data collection plans would specify approximate view directions of the X-ray imager with respect to the registration object with the goal of maximizing geometric constraint in the subsequent registration process.

In this dissertation, it is assumed that accuracy requirements are uniform in all directions (i.e., isotropic). In general, this may not be the case. For example, consider the problem of localizing a pelvis for the task of placing an implant into the pelvic cup (i.e., the acetabulum). Many acetabular implants are symmetric about the central axis, and therefore localiza-

tion of the pelvis about this axis may not be critical. In this example, Intelligent Data Selection would select points in a manner which completely localizes the registration object. Potentially valuable Data points would be "spent" localizing the pelvis about the acetabulum's central axis, an unnecessary task. It may be possible to reformulate the constraint analysis criterion measure to consider transformations only in certain pre-specified directions. Similarly, it might be desirable to weight certain localization directions more heavily than others. For example, for a given task it may be desirable to localize a human femur translationally along its central axis less accurately than in other directions.

Despite the encouraging experimental results which demonstrate the potential benefits of the Intelligent Data Selection and online accuracy estimation methods, additional study is required to assess the practicality of these methods for use in particular application areas (e.g., computer-assisted surgery, factory automation). Data collection efficiency will play an important role in determining when the proposed methods are practical. If it is difficult and time consuming to collect specified Data points from a Data collection plan compared to unconstrained Data collection, it is less likely that the Intelligent Data Selection method will be practical. The efficiency of Data collection is likely to be application dependent. For example, using a robot or numerically controlled machine, the cost of collecting a specified Data point may be the same as the cost of collecting a random one; however, when Data collection is performed manually by a human, this may not be the case. Additional study is necessary to evaluate these issues.

In the current constraint synthesis process, there is no way to incorporate the fact that certain Data points may be more expensive than others to collect. For example, in computer-assisted orthopaedic surgery, certain Data points can be collected directly from exposed bony surfaces, while other Data points may require percutaneous (i.e., through the skin) collection, with associated higher collection costs. In this example, it may be desirable to limit the quantity of percutaneous Data collection, if possible. Modifications could be made to the constraint analysis measure to incorporate Data collection costs which vary between regions.

In the framework presented in this dissertation, surface Models are built from CT image data. In general, it is desirable to minimize the number of CT images required to build sufficiently accurate surface Models. CT acquisition parameters such as inter-slice spacing, slice thickness, and in-slice resolution, should be adjusted to minimize radiation exposure to the patient while ensuring that the resulting surface Models are accurate enough to satisfy registration accuracy requirements. Studying this problem requires an understanding of how surface Model accuracy varies as a function of CT imaging parameters. By applying a signal theoretic approach to this problem, it may be possible to vary CT resolution parameters during the scanning process in a manner which reflects the spatial frequency of the underlying anatomical structure.

In the area of high-speed tracking and registration via the speed-enhanced ICP algorithm, it should be possible to parallelize the registration process by dividing the closest point computations over multiple processors. This should result in speed improvement roughly proportional to the number of processors. It would also be interesting to examine the Intelligent Data Selection method in the context of high-speed tracking. In this application, there are high costs associated with the collection of large amounts of registration data (i.e., reduction in tracking speed). Therefore, Intelligent Data Selection may be useful for identifying regions of the registration object in which Data collection should be concentrated with the goal of minimizing the quantity of Data, and thus the maximizing tracking speed, required for accurate tracking.

Appendix A

Application of the Speed-Enhanced ICP Algorithm to High-Speed Tracking

This appendix describes a system for tracking the position and orientation of arbitrarily-shaped objects from range data at speeds of roughly 10 Hz. The Carnegie Mellon University (CMU) high-speed range sensor, which is capable of acquiring 32x32 cell range images at 500 Hz, provides the data used for tracking. The primary software component of the system is the speed enhanced iterative closest point (EICP) algorithm described in Section 2.4. While other researchers have addressed the problem of pose tracking using range data [Grimson et al., 1994] [Yamamoto, 1993], to this author's knowledge none have demonstrated sub-second performance without the use of fiducial markers.

A.1 The Tracking Algorithm

An outline of the tracking algorithm is shown in Figure A-1. Each box in the diagram represents a processing step, and the processing sequence is indicated by the large-headed arrows. Inputs to a processing step are indicated by the quantities to the left of each box, while outputs are indicated by the quantities to the right.

During initialization, a pre-computed triangle mesh Model, *M*, is loaded into memory, and a k-d tree is built from *M*. In the tracking experiments, *M* is constructed offline using a technique based on deformable surfaces [Delingette et al., 1992]. This technique can fuse range data collected from multiple views into a single triangle mesh surface model. The surface model was created from range data acquired using three identical, commercially available light-stripe range finders [Sato and Inokuchi, 1987]. These sensors were calibrated so that all data were expressed in a single, world-centered coordinate frame.

To initialize the tracking algorithm, the transformation between the Model, *M*, and the initial object Data, *D*[0], must be calculated. This transformation, ${}^{M}\boldsymbol{T}_{D}[0]$, can be found in several seconds using the EICP algorithm with a starting transformation provided by the user. Fully

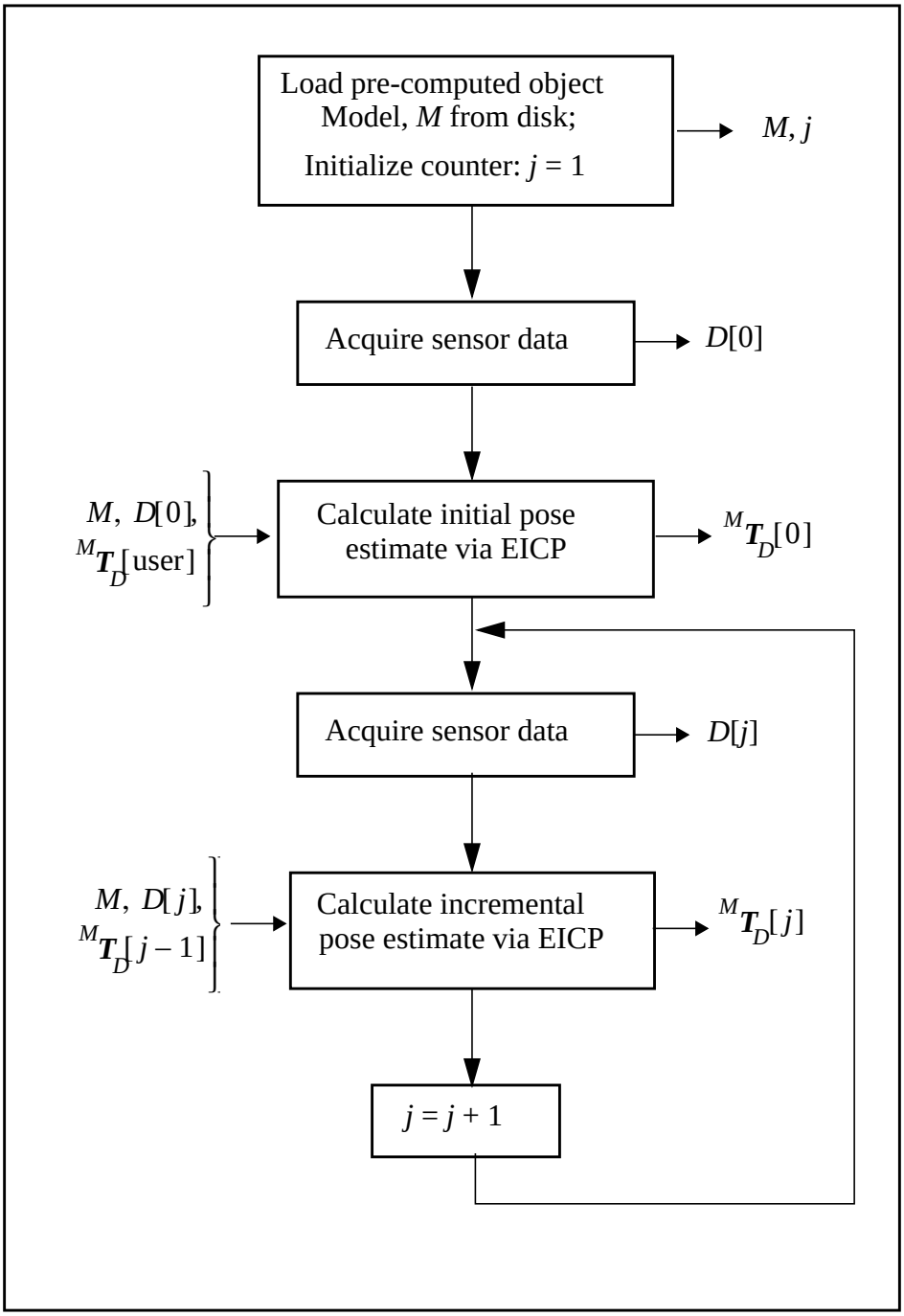


Figure A-1: Framework of the high-speed tracking algorithm.

automatic initialization would be possible using techniques similar to those described in [Johnson and Hebert, 1996]. Initial pose errors as large as 15 degrees of rotation about each axis, and 50% of the object size in translation along each axis will typically converge to the global minimum. Once ${}^{M}\boldsymbol{T}_{D}[0]$ has been calculated, it is used to transform the Model, *M* to

the initial object pose. All future pose estimates are measured with respect to this initial starting pose.

After initialization, the algorithm enters a tracking loop. Within the loop, Data are acquired by the high speed range sensor, and the object pose is estimated via the speed enhanced iterative closest point (EICP) algorithm in roughly 0.1 - 0.3 sec. These high speeds are possible for two reasons. First, the difference in object pose at time *j* and time *j-1* is typically small. For example, translational velocities of 10 cm per second and rotational velocities of 20 degrees per second lead to incremental object pose discrepancies of roughly 2 cm and 4 degrees. Thus, since the EICP algorithm uses ${}^{M}\boldsymbol{T}_{D}[j-1]$ as the starting point when finding ${}^{M}\boldsymbol{T}_{D}[j]$, the algorithm can perform registration in a small number of iterations, typically 3-10. Second, the resolution of the range Data used in the tracking loop, usually 16x16, is less than the full sensor resolution of 32x32. The reduced number of Data points in the set *D*[*j*] results in a faster calculation of the pose estimate.

During each data acquisition cycle, two simple preprocessing steps are performed on the range Data: noise reduction and background-foreground separation. For the CMU high speed range sensor, noisy data is associated with poor reflection of projected light from the object. Such data can be reduced by thresholding the reflected intensity values which can be measured by each cell in the sensor. Background-foreground separation determines which data points lie on the surface of the object to be tracked. Since the tracking experiments were performed in an uncluttered environment, range data on the object surface can be distinguished by thresholding the Z component of the range data. While this simple operation has worked well in this setup, a more sophisticated approach would be required if the object were in a cluttered environment.

Using ${}^{M}\boldsymbol{T}_{D}[j-1]$ as the starting point for incremental pose estimation works well when object motion is erratic and unpredictable. In some situations, object motion may be smooth, continuous and thus easier to predict using an extrapolation method such as Kalman filtering. A Kalman filter has not been implemented for this purpose; however, both first and second order extrapolation methods have been applied and tested. Since the extrapolated pose is often closer to the true pose than ${}^{M}\boldsymbol{T}_{D}[j-1]$, the time required to compute the pose via EICP is reduced.

A.2 Experimental Setup

The experimental setup is shown in Figure A-2. The CMU high speed VLSI range sensor [Gruss et al., 1992] consists of two primary components: the sensor head and the light stripe generator. The tracked object, in this case a small bust of the goddess Venus, is mounted on the end effector of a Microbot robot. The CCD imager is not a primary component of the system, but is used for display purposes only. Not shown is a Sparc-10 workstation used for

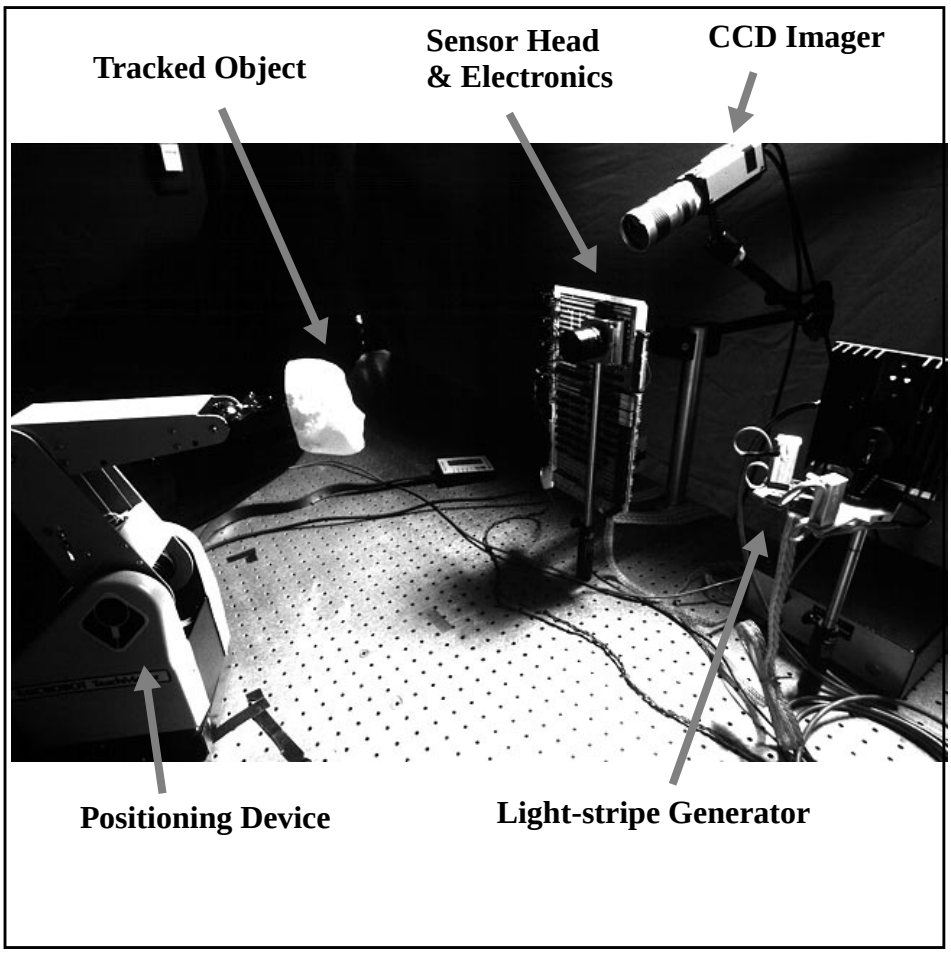


Figure A-2: Experimental apparatus for the high-speed tracking system.

computing the pose estimate and for graphically displaying a 3-D model of the tracked object. The pose of the graphical 3-D model is updated at high speed to reflect the current object pose estimate.

The CMU high speed range sensor is based on a modified version of the traditional light-stripe range imaging technique known as the cell-parallel light-stripe method. The primary advantage of the cell-parallel method is that range image acquisition time is independent of the number of data points in each frame.

The current version of the CMU range sensor can acquire a complete 32x32 cell range image in as little as one millisecond. The range data is acquired at 10 bits of resolution, and is accurate to 0.1% or better (e.g., 0.5 mm accuracy at a distance 500 mm) [Gruss et al., 1992]. The sensor workspace is shaped like a four sided pyramid. As configured for the reported experiments, a cross section of the workspace is an 11.5 cm square at a distance

of 55 cm from the sensor along the optical axis. Thus, the lateral sensor resolution at this distance is about 2.8 range measurements per centimeter in each direction.

The results presented in the next section were collected using the face object shown in Figure A-3. This object was manufactured directly from a triangle mesh CAD model using a stereolithographic process [Marcus and Bourell, 1993]. The advantage of this approach is that the physical object is very accurately represented by the corresponding CAD model. Thus, for purposes of characterizing system accuracy, errors caused by differences between the physical object and the Model, *M*, are reduced.

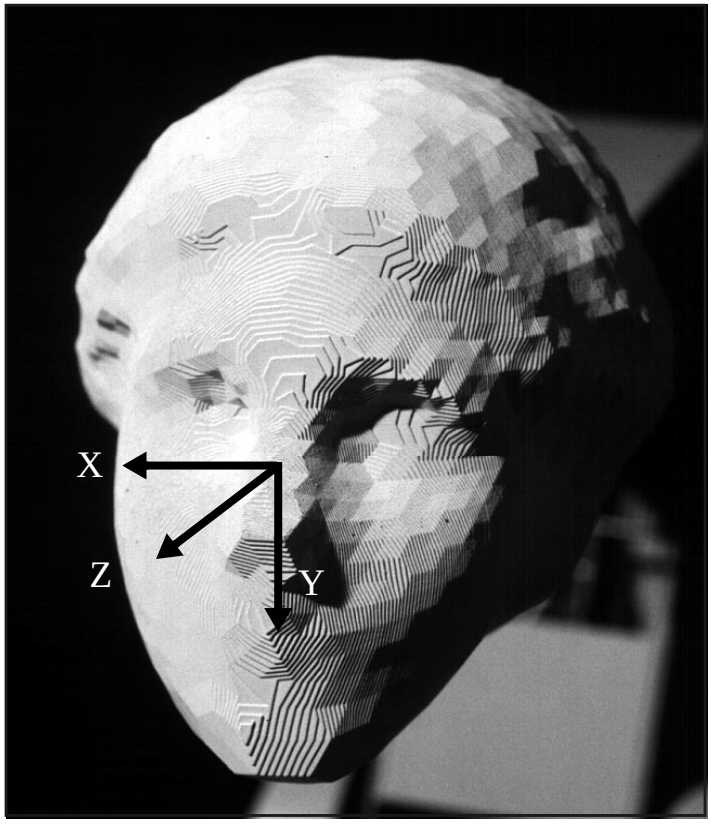


Figure A-3: Face object used in the tracking experiments.

All pose estimates presented below are specified in an object centered coordinate system as shown in Figure A-3. The object itself is roughly 8 cm x 10 cm x 6 cm in the X, Y, and Z directions, respectively.

A.3 Pose Estimation Results

There are two results presented in this section. The first demonstrates the ability of the tracking system to *accurately* estimate the pose of stationary, or slowly moving objects. The sec-

ond demonstrates the ability to track complex motions in a highly *repeatable* manner. At the time these experiments were performed, the equipment required to generate complex and accurately calibrated dynamic trajectories which were precisely known at each point along the trajectory was not available. Therefore, it remains to be shown that the system can *accurately* track *high-speed* motions.

A.3.1 Static Accuracy Results

The graphs in Figure A-4 demonstrate the absolute accuracy of the system when the object is assumed to be stationary. To collect this data, the object was manually positioned to selected points along a trajectory using a high precision positioning device. At each point, 100 pose estimates were computed, and corresponding mean and standard deviation values were calculated. Each data point in the graphs compares the object's ground truth position to the mean of the corresponding estimated position. The solid line represents the zero error case, and vertical deviations from this line can be interpreted as error.

The object trajectory for these experiments consisted of coupled translations along each axis, and rotations about the y-axis. It was not possible to generate rotations about the x and z-axes due to limitations in the experimental apparatus. Multiple data points at a given actual position arise since certain translations or rotations may be visited multiple times as the object is moved through the trajectory. The measurement errors visible in the graphs should therefore be interpreted as pose-dependent inaccuracies. For example, the errors visible in X rotation represent the range of X rotation estimates over all points along the trajectory (recall that X rotation of the physical object remains constant). The average error between ground truth and estimated positions is 0.93 mm in the translation components and 1.4 degrees in the rotation components. The standard deviation of each pose estimate is less than 0.06 mm in translation and 0.1 degree in rotation. Part of the measured error may be caused by the fact that the physical axis of rotation (i.e., the y-axis) may not correspond exactly to the y-axis in the Model.

The results of Figure A-4 demonstrate that the system can generate accurate pose estimates for stationary or slowly moving objects. In these experiments, the full resolution of the sensor was used, and the EICP termination thresholds, ε_{Rr}, ε_{Tr}, were small. In the current implementation, the system is only capable of tracking very slowly moving objects using these parameter settings. When tracking faster motions, such as those described in Section A.3.2, the sensor resolution is typically decreased by a factor of 2, while the EICP termination thresholds are increased.

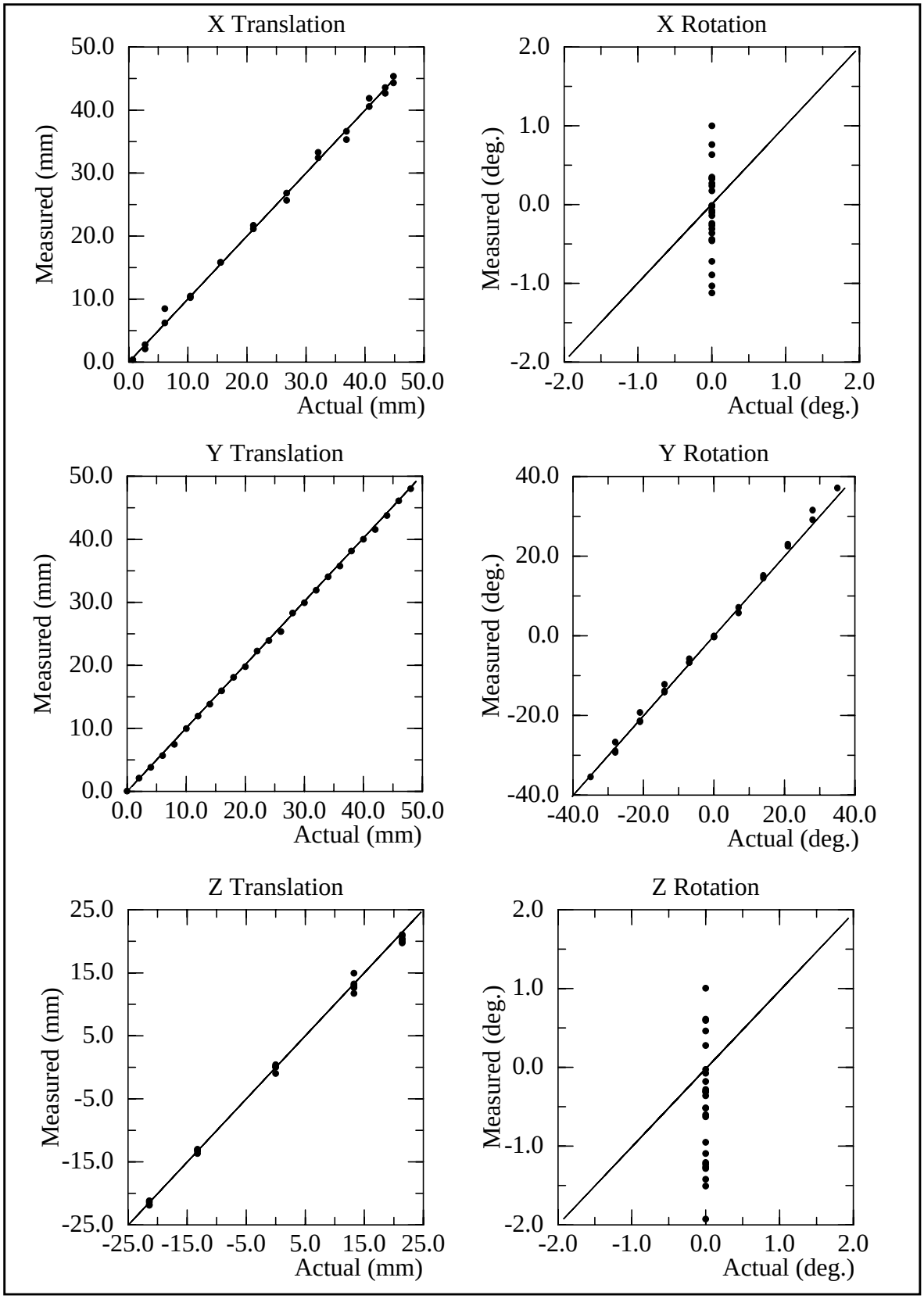


Figure A-4: Experimental static accuracy results for the high-speed pose tracker.

A.3.2 Dynamic Tracking Results

Figure A-5 contains plots of estimated pose as the object is moved through a complex trajectory by the Microbot. Pose estimates are specified with respect to the object's initial pose at time 0. Maximum object velocities are roughly 100 mm/sec in translation and 22 degrees/sec in rotation. Each graph in this figure actually contains 2 overlaid data sets corresponding to 2 different executions of the trajectory. Furthermore, each single execution of the trajectory is periodic with two cycles shown. It is evident from these graphs that the *repeatability* of the pose estimation system is quite good. These results also demonstrate that the system can perform pose estimation fast enough to track object motion at the velocities specified above. The average cycle time in these experiments was about 0.3 seconds (3.3 Hz), with variation between about 0.1 seconds (10 Hz) and 0.5 seconds (2 Hz). This variation in cycle time reflects variation in the initial pose estimate, ${}^{M}\boldsymbol{T}_{D}[j-1]$, relative to the final pose estimate, ${}^{M}\boldsymbol{T}_{D}[j]$. Large transformations between initial and actual pose result in an increased number of iterations required by the EICP algorithm, and thus a longer overall cycle times. Therefore, faster object velocities typically lead to longer cycle times, while slower velocities lead to shorter cycle times.

A.3.3 Human Face Tracking

The high-speed pose tracker has also been used to estimate the pose of real human faces. A polygonal mesh surface Model of one of these faces is seen in Figure A-6. The data in this figure was acquired using a commercially available light-stripe range finder [Sato and Inokuchi, 1987].

While no quantitative results were obtained for the human face tracking experiments, several results which qualitatively demonstrated the speed, accuracy and viability of the approach were performed. Figure A-7 shows a sequence of video images of a human face, together with images of a 3-D graphical icon which represent pose estimates of the corresponding face image. Use of the icon allowed much faster graphical rendering than if the full surface mesh were displayed. In the icon, the pupils and the tip of the nose lie in a raised plane relative to the other elements of the icon, thus allowing visualization of out of plane rotations. The motion observed in the sequence is primarily within a single plane. A small amount of out-of-plane rotation is evidenced by a shifting of the pupils within the icon's eyes.

A.3.4 Discussion

High-speed, 3-D pose tracking has potential uses in a variety of applications. In manufacturing, 3-D tracking could allow a mechanism (e.g., a robot) to perform an operation (e.g., grasping) on arbitrarily-shaped moving parts. In human computer interaction, high-

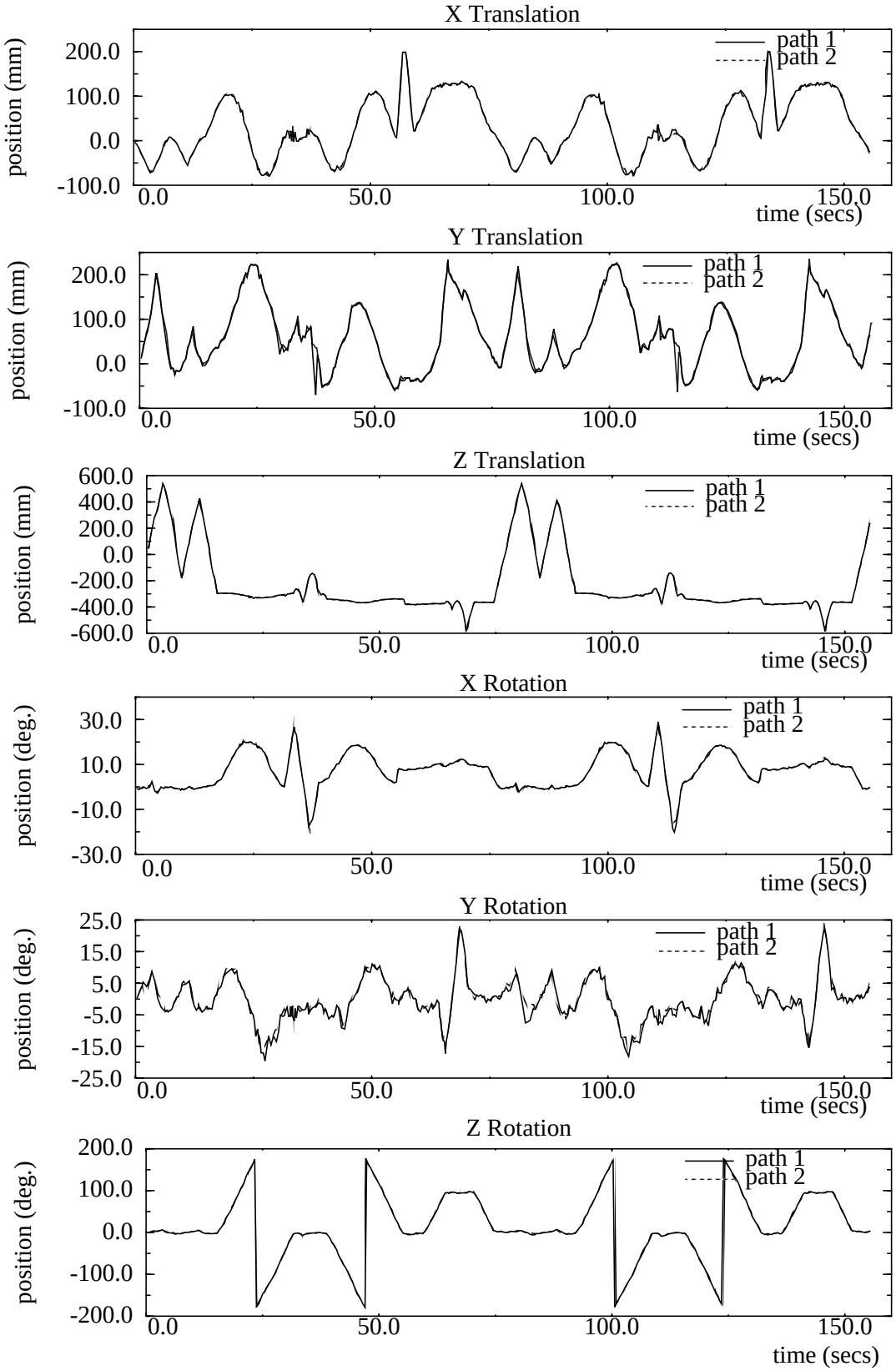


Figure A-5: Experimental dynamic repeatability results for the high-speed pose tracker.

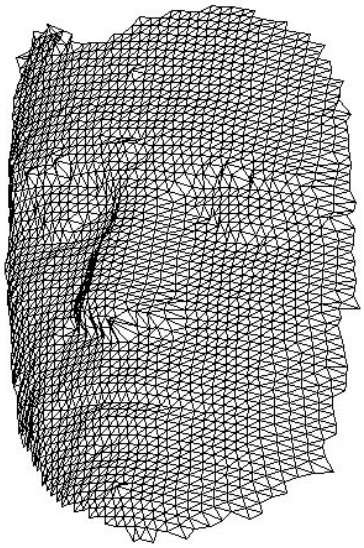


Figure A-6: Surface mesh of the author's face.

speed pose estimation could be used to track body movements for subsequent interpretation as input to a computer. In medical registration, patient tracking could help eliminate invasive and bulky fixation devices.

This appendix describes an approach for estimating the full 3-D pose of arbitrarily-shaped, rigid objects at speeds up to 10 Hz. The approach uses a high speed VLSI range sensor capable of acquiring 32x32 cell range images in 1 millisecond or less; however, the technique is independent of the particular sensor used. The primary software component of the tracker is the speed enhanced iterative closest point (EICP) algorithm described in Section 2.4.

As demonstrated in this appendix and Chapter 2, the EICP algorithm is capable of very high-speed registration. To this author's knowledge, the implementation of EICP described in this work is capable of shape-based registration at speeds which are significantly faster than systems developed by other researchers (based upon a qualitative assessment of other systems described in the literature). It is difficult to know precisely how much faster EICP is due to a lack of standardized test data for making such evaluations.

There is an interesting phenomenon observed in this work which is common to many systems for tracking moving objects. As noted above, there is a direct relationship between the magnitude of the true registration transformation and the computation time required to estimate this transformation. In general, slower object velocities (or higher computational power) result in the ability to estimate the registration transformation in shorter periods of time, possibly using simpler algorithmic approaches. There is a positive feedback effect

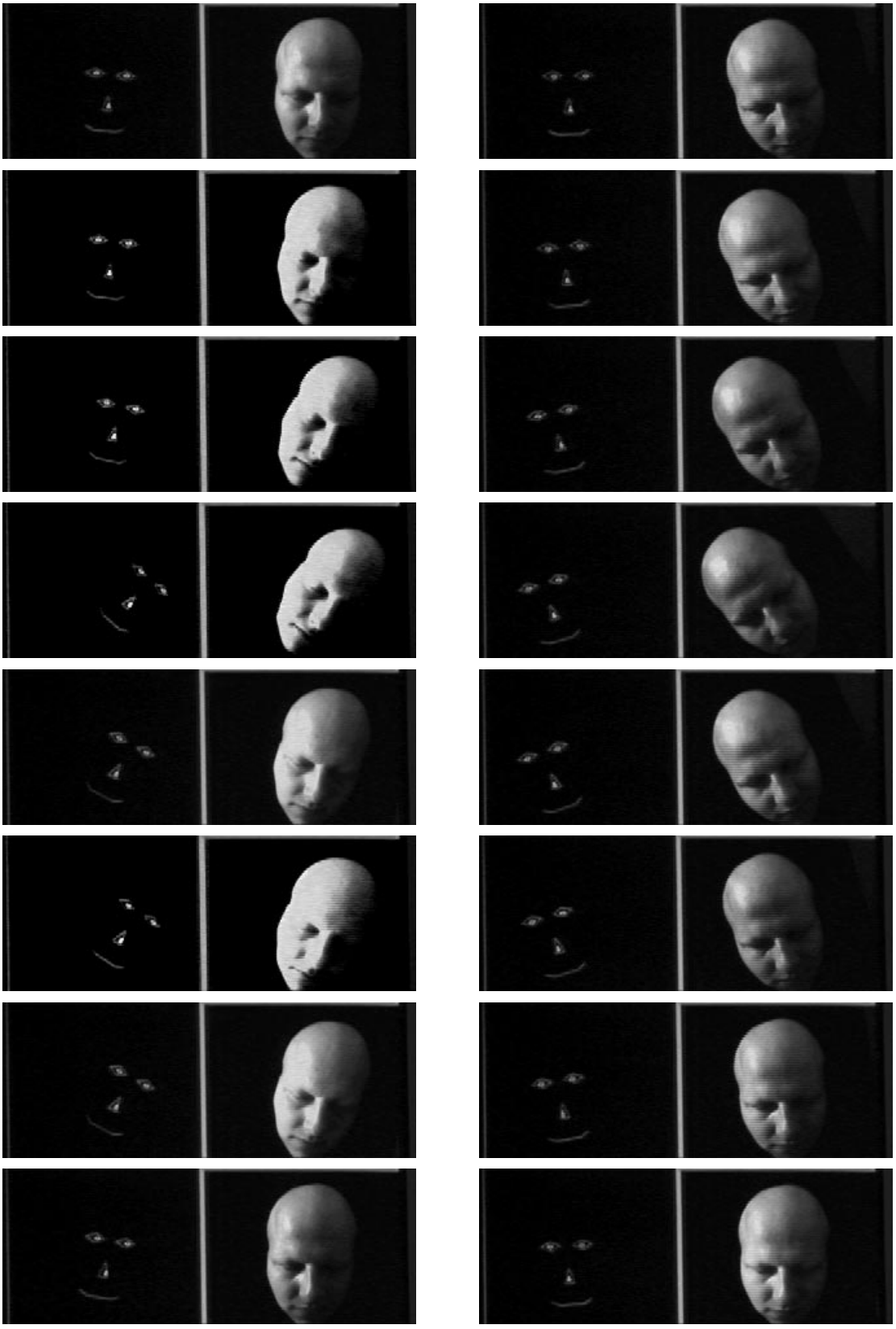


Figure A-7: Demonstration of high-speed human face pose tracking.

implied by this phenomenon. The ability to solve for the registration transformation at higher speeds will result in smaller changes in the object pose transformation between tracking cycles. Smaller changes in pose transformation will, in turn, result in the ability to solve for the transformation faster, thus reducing the pose transformation. An interesting question arises from this observation. Assuming that the object pose transformation between cycles of the tracking loop are small, what is the simplest possible registration algorithm which can be used to solve for the pose transformation? Identifying such a simplified registration algorithm may result in the ability to perform pose estimation at significantly higher speeds than those reported in this work.

Appendix B

Derivation of the Sensitivity Vector

As suggested in Equation 3.2, the following expression is an approximation of the distance between a point and a surface:

$$D(\boldsymbol{x}) = \frac{F(\boldsymbol{x})}{\|\nabla F(\boldsymbol{x})\|}$$

where $F(\boldsymbol{x}) = 0$ is the implicit equation of the surface, $\|\nabla F(\boldsymbol{x})\|$ is the magnitude of the gradient to the surface, and $\boldsymbol{x}$ is a point which may or may not lie on the surface. The choice of an implicit surface as a representation is relaxed below so that the final result is independent of surface representation.

Assume that there exists a point $\boldsymbol{x}_s$, which lies on the surface such that $D(\boldsymbol{x}_s) = 0$. This point can be perturbed with respect to the surface by applying a differential transformation, *T* to the point. *T* can be represented by a homogeneous transformation, and is a function of 6 parameters $(t_x, t_y, t_z, \omega_x, \omega_y, \omega_z)$ where $(\omega_x, \omega_y, \omega_z)$ are rotations about the *X, Y,* and *Z* axes, respectively, and (t_x, t_y, t_z) are translations along the newly rotated *X, Y,* and *Z* axes. Define:

$$\boldsymbol{t} = \left[t_x \; t_y \; t_z \; \omega_x \; \omega_y \; \omega_z\right]^{\mathbf{T}} \tag{B.1}$$

as the 6-vector of parameters. If rotations are applied in the order: $Rot(X, \omega_x), Rot(Y, \omega_y), Rot(Z, \omega_z)$, followed by translations along the newly rotated axes: $Trans(X^*, t_x), Trans(Y^*, t_y), Trans(Z^*, t_z)$, then the corresponding homogeneous transformation matrix, *T* is:

$$T = \begin{bmatrix} cz\ cy & (cz\ sy\ sx - sz\ cx) & (cz\ sy\ cx + sx\ sz) & t_x \\ sz\ cy & (sz\ sy\ sx + cx\ cz) & (sz\ sy\ cx - cz\ sx) & t_y \\ -sy & cy\ sx & cy\ cx & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{B.2}$$

where $sx = \sin(\omega_x)$, $cx = \cos(\omega_x)$, and likewise for the *y* and *z* components. In order to determine the variation of the distance, *D*, as the surface point is perturbed, the gradient of *D* about $\boldsymbol{t} = 0$ is calculated:

$$\frac{\partial}{\partial \boldsymbol{t}} D(T(\boldsymbol{x}_s)) = \left[\frac{\partial D}{\partial t_x} \frac{\partial D}{\partial t_y} \frac{\partial D}{\partial t_z} \frac{\partial D}{\partial \omega_x} \frac{\partial D}{\partial \omega_y} \frac{\partial D}{\partial \omega_z} \right]^{\mathrm{T}} \Bigg|_{\boldsymbol{t} = 0} \tag{B.3}$$

Thus, it is necessary to calculate the partials of:

$$D(T(\boldsymbol{x}_s)) = \frac{F(T(\boldsymbol{x}_s))}{\left\| \nabla F(T(\boldsymbol{x}_s)) \right\|}$$

with respect to each of the parameters of $\boldsymbol{t}$. By applying the quotient rule for derivatives:

$$\frac{\partial}{\partial \boldsymbol{t}} D(T(\boldsymbol{x}_s)) = \frac{1}{\left\| \nabla F(T(\boldsymbol{x}_s)) \right\|} \frac{\partial}{\partial \boldsymbol{t}} F(T(\boldsymbol{x}_s)) \Bigg|_{\boldsymbol{t} = 0} \tag{B.4}$$

where the second term has dropped out, since by the definition of an implicit surface:

$$F(T(\boldsymbol{x}_s))\big|_{\boldsymbol{t} = 0} = 0$$

Applying the chain rule to find the partials with respect to each element of $\boldsymbol{t}$:

$$\frac{\partial}{\partial t_i} D(T(\boldsymbol{x}_s)) = \frac{1}{\left\| \nabla F(T(\boldsymbol{x}_s)) \right\|} \cdot \frac{\partial T}{\partial t_i} \boldsymbol{x}_s \cdot \nabla F(T(\boldsymbol{x}_s)) \Bigg|_{\boldsymbol{t} = 0} \tag{B.5}$$

where t_i is a shorthand used to represent the *i'th* element of the parameter vector, $\boldsymbol{t}$. Note that the unit gradient evaluated at a point on the surface $F(\boldsymbol{x}_s) = 0$, is simply the surface normal evaluated at that point. Thus:

$$\frac{\nabla F(T(\boldsymbol{x}_s))}{\left\| \nabla F(T(\mathrm{x}_s)) \right\|} \Bigg|_{\boldsymbol{t} = 0} = \boldsymbol{n}_{\boldsymbol{x}_s} \tag{B.6}$$

where $\boldsymbol{n}_{\boldsymbol{x}_s}$ is the unit surface normal evaluated at the point $\boldsymbol{x}_s$. Equation B.6 allows the final result to be expressed independently of surface representation.

Expanding each of the elements of $\left.\frac{\partial T}{\partial \boldsymbol{t}}\right|_{\boldsymbol{t}=0}$, and using the fact that the rotations are infinitesimal:

$$\frac{\partial T}{\partial t_x} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad \frac{\partial T}{\partial t_y} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad \frac{\partial T}{\partial t_z} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\frac{\partial T}{\partial \omega_x} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad \frac{\partial T}{\partial \omega_y} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad \frac{\partial T}{\partial \omega_z} = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{B.7}$$

Multiplying each of the above by $\boldsymbol{x}_s$, grouping into a matrix, and eliminating the last column (which results from the homogenous coordinates), results in the Jacobian, *J*, of the transformation *T*.

$$J = \begin{bmatrix} \left(\frac{\partial T}{\partial t_x}\boldsymbol{x}\right)^{\mathrm{T}} \\ \circ \\ \circ \\ \circ \\ \left(\frac{\partial T}{\partial \omega_z}\boldsymbol{x}\right)^{\mathrm{T}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix} \tag{B.8}$$

Combining Equations B.5, B.6 and B.8, results in:

$$\frac{\partial}{\partial \boldsymbol{t}} D(T(\boldsymbol{x}_s)) = J\,\boldsymbol{n}_{\boldsymbol{x}_s} \tag{B.9}$$

This expression can be simplified further by expanding the right hand side:

$$\begin{aligned} \frac{\partial}{\partial \boldsymbol{t}} D(T(\boldsymbol{x}_s)) &= \left[n_x \; n_y \; n_z \; (yn_z - zn_y) \; (zn_x - xn_z) \; (xn_y - yn_x)\right]^{\mathrm{T}} \\ &= \begin{bmatrix} \boldsymbol{n}_{\boldsymbol{x}_s} \\ \boldsymbol{x}_s \times \boldsymbol{n}_{\boldsymbol{x}_s} \end{bmatrix} \end{aligned} \tag{B.10}$$

This is the final result which relates a small change in the position of a point originally on the surface to a change in distance between the point and the surface. The result does not depend upon the implicit surface representation, but only upon the location of the surface point, and the surface normal at that location. Throughout the thesis, the subscript from the normal vector is dropped, so that $\boldsymbol{n}_{\boldsymbol{x}_s} = \boldsymbol{n}$. A geometric interpretation of Equation B.10 is presented in Section 3.3.

Appendix C

Effect of Scale upon Constraint Analysis

As discussed in Section 3.3.3, the size of a rigid body will change the relative contribution of rotations and translations in constraint analysis. The effect of object size is demonstrated in Figure C-1 for the case of the cube using Data configuration C2. Recall from Table 3-7 that the constraint analysis eigenvalues for this configuration can be separated into rotational and translational components. All rotational eigenvalues in the table are the same, as are all translational eigenvalues. Figure C-1 plots the magnitude of the translational and rotational eigenvalues, and the NAI as a function of cube size. A scale of 1 is equivalent to the default constraint analysis scaling of Equation 3.28. The magnitude of the three translational eigenvalues remains constant as a function of scale. In contrast, the magnitude of the rotational eigenvalues (and thus the rotational contribution to constraint analysis) increases as a function of size. The NAI increases until the rotational eigenvalues become larger than the constant translational eigenvalues, and then it begins to decrease. The result of this experiment demonstrate that the mechanism for pre-scaling an object described in Section 3.3.3 has the desired behavior, at least for the case of a cube. For more complex objects, similar results

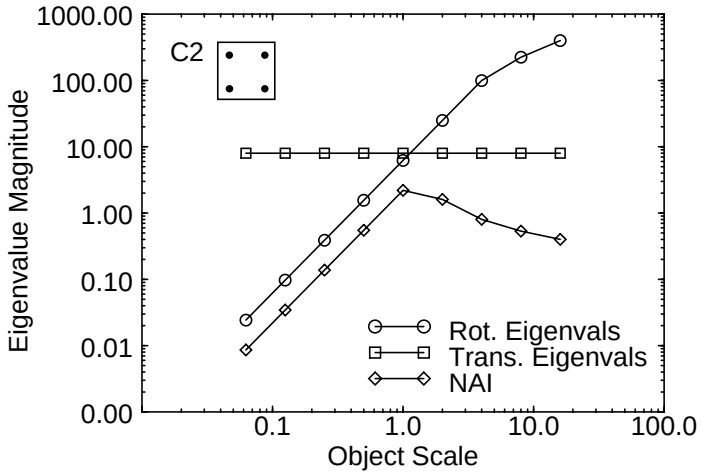


Figure C-1: Magnitude of the translation and rotation eigenvalues, and the noise amplification index as a function of scale for the cube Data configuration, C2.

can be obtained; however, near an object scale of 1, the rotational and translational components of constraint analysis are usually coupled, and can therefore not be represented using the simple graphical representation of Figure C-1.

Bibliography

Arun, K. S., Huang, T. S., and Blostein, S. D. (1987). Least-squares fitting of two 3-d point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):698–700.

Asada, H. and By, A. B. (1985). Kinematic analysis of workpart fixturing for flexible assembly with automatically reconfigurable fixtures. *IEEE Transactions on Robotics and Automation*, RA-1(2):86–94.

Asada, H. and Kitagawa, M. (1989). Kinematic analysis and planning for form closure grasps by robotic hands. *Robotics and Computer-Integrated Manufacturing*, 5(4):293–299.

Ayache, N., editor (1995). *Proceedings of the First International Conference on Computer Vision, Virtual Reality and Robotics in Medicine*, Nice, France. Springer-Verlag.

Ayache, N. and Duncan, J., editors (1995). *Medical Image Analysis*. Oxford University Press.

Baluja, S. (1995). An empirical comparison of seven iterative and evolutionary function optimization heuristics. School of Computer Science CMU-CS-TR-95-193, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213.

Baluja, S. and Caruana, R. (1995). Removing the genetics from the standard genetic algorithm. In Prieditis, A. and Russell, S., editors, *Proceedings of the International Conference on Machine Learning (ML-95)*, pages 38–46, San Mateo, CA. Morgan Kaufmann Publishers.

Baluja, S. and Simon, D. A. (1996). Evolutionary methods for selecting point data for object localization: Applications to computer-assisted surgery. School of Computer Science CMU-CS-TR-96-183, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213.

Barrow, H. G., Tenenbaum, J. M., Bolles, R., and Wolf, H. C. (1977). Parametric correspondence and chamfer matching: Two new techniques for image matching. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 659–663.

Bentley, J. L. (1975). Multidimensional binary search trees used for associative searching. *Communications of the Association for Computing Machinery*, 18(9):509–517.

Besl, P. J. and McKay, N. D. (1992). A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256.

Betting, F., Feldmar, J., Ayache, N., and Devernay, F. (1995). A new framework for fusing stereo images with volumetric medical images. In Ayache, N., editor, *Proceedings of the First International Conference on Computer Vision, Virtual Reality and Robotics in Medicine*, pages 30–39, Nice, France. Springer-Verlag.

Borm, J. and Menq, C. (1991). Determination of optimal measurement configurations for robot calibration based on observability measure. *The International Journal of Robotics Research*, 10(1):51–63.

Brown, L. G. (1992). A survey of image registration techniques. *ACM Computing Surveys*, 24(4):325–376.

Bucholz, R. D., editor (1995). *Journal of Image Guided Surgery*. John Wiley & Sons, Inc.

Champleboux, G., Lavallee, S., Szeliski, R., and Brunie, L. (1992). From accurate range imaging sensor calibration to accurate model-based 3-d object localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 83–89.

Cox, I. J., Kriskal, J. B., and Wallach, D. A. (1990). Predicting and estimating the accuracy of a subpixel registration algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(8):721–734.

Craig, J. J. (1986). *Introduction to Robotics Mechanics & Control*. Addison-Wesley Publishing Company, Inc., Reading, Massachusetts, 1st edition.

Cuchet, E., Knoplioch, J., Dormont, D., and Marsault, C. (1995). Registration in neurosurgery and neuroradiotherapy applications. In *Proceedings of the Second International Symposium on Medical Robotics and Computer Assisted Surgery*, pages 31–38, Baltimore.

DeJong, K. (1975). *An Analysis of the Behavior of a Class of Genetic Adaptive Systems*. PhD thesis, University of Michigan.

Delingette, H., Hebert, M., and Ikeuchi, K. (1992). Shape representation and image segmentation using deformable surfaces. *Image and Vision Computing*, 10(3):132–144.

DiGioia, A., Simon, D., Jaramaz, B., Blackwell, M., Morgan, F., O'Toole, R., Colgan, B., and Kischell, E. (1995). HipNav: pre-operative planning and intra-operative navigational guidance for acetabular implant placement in total hip replacement surgery. In *Proceedings of the Computer Assisted Orthopaedic Surgery Symposium*, Bern, Switzerland.

DiGioia, A. M., Kanade, T., and Taylor, R., editors (1994). *Proceedings of the First International Symposium on Medical Robotics and Computer Assisted Surgery*, Pittsburgh, PA.

DiGioia, A. M. and Taylor, R., editors (1995). *Proceedings of the Second International Symposium on Medical Robotics and Computer Assisted Surgery*, Baltimore, MD.

Dohi, T., editor (1995). *Journal of Computer Aided Surgery*. Japan Society of Computer Aided Surgery.

Doty, K. L., Melchiorri, C., Schwartz, E. M., and Bonivento, C. (1995). Robot manipulability. *IEEE Transactions on Robotics and Automation*, 11(3):462–468.

Ettinger et al., G. (1994). Automatic registration for multiple sclerosis change detection. In *Applications of Computer Vision in Medical Image Processing*, pages 297–306. AAAI.

Faugeras, O. D. and Hebert, M. (1986). The representation, recognition, and locating of 3-d objects. *The International Journal of Robotics Research*, 5(3):27–52.

Feldmar, J., Ayache, N., and Betting, F. (1995). 3d-2d projective registration of free-form curves and surfaces. In *International Conference on Computer Vision*, pages 549–556, Boston, MA.

Friedman, J. H., Bentley, J. L., and Finkel, R. A. (1977). An algorithm for finding best matches in logarithmic expected time. *ACM Transactions on Mathematical Software*, 3(3):209–226.

Fua, P. and Leclerc, Y. G. (1994). Registration without correspondences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 121–128, Seattle.

Galloway, R. L. and Maciunas, R. J. (1990). Stereotactic neurosurgery. *Critical Reviews in Biomedical Engineering*, 18(3):181–205.

Geiger, B. (1993). *Three-dimensional modeling of human organs and its application to diagnosis and surgical planning*. PhD thesis, Ecole des Mines de Paris.

Goldberg, D. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Publishing Company, Inc., Reading, Massachusetts.

Grimson, W. E. L., Ettinger, G. J., White, S. J., Gleason, P. L., Lozano-Perez, T., Wells, W. M., and Kikinis, R. (1995). Evaluating and validating an automated registration system for enhanced reality visualization in surgery. In Ayache, N., editor, *Proceedings of the First International Conference on Computer Vision, Virtual Reality and Robotics in Medicine*, pages 3–12, Nice, France. Springer-Verlag.

Grimson, W. E. L., Lozano-Perez, T., Wells, W. M., Ettinger, G. J., White, S. J., and Kikinis, R. (1994). Automated registration for enhanced reality visualization in surgery. In *Proceedings of the First International Symposium on Medical Robotics and Computer Assisted Surgery*, pages 82–89, Pittsburgh, PA.

Gruss, A., Tada, S., and Kanade, T. (1992). A VLSI smart sensor for fast range imaging. In *International Conference on Intelligent Robots and Systems (IROS '92)*, pages 349–58, Raleigh, NC. IEEE.

Hamadeh, A., Sautot, P., Lavallee, S., and Cinquin, P. (1995). Towards automatic registration between CT and X-ray images: Cooperation between 3d/2d registration and 2d edge detection. In *Proceedings of the Second International Symposium on Medical Robotics and Computer Assisted Surgery*, pages 39–46, Baltimore.

Haralick, R. M. and Joo, H. (1988). 2d-3d pose estimation. In *Proceedings of the 9th International Conference on Pattern Recognition*, pages 385–391, Rome, Italy. IAPR, IEEE.

Haralick, R. M., Joo, H., Lee, C. N., Zhuang, X., Vaidya, V. G., and Kim, M. B. (1989). Pose estimation from corresponding point data. *IEEE Transactions on Systems, Man and Cybernetics*, 19(6):1426–1446.

Hauser, D. L. and Taylor, D. L. (1990). Optimal alignment of geometric models for comparison. In *Proceedings of the First Conference on Visualization in Biomedical Computing*, pages 426–431, Atlanta. IEEE.

Henri, C. J., Colchester, A. C. F., Zhao, J., Hawkes, D. J., Hill, D. L. G., and Evans, R. L. (1995). Registration of 3-d surface data for intra-operative guidance and visualization in frameless stereotactic neurosurgery. In Ayache, N., editor, *Proceedings of the First International Conference on Computer Vision, Virtual Reality and Robotics in Medicine*, pages 47–56, Nice, France. Springer-Verlag.

Hertz, J., Krogh, A., and Palmer, R. (1991). *Introduction to the Theory of Neural Computation*. Addison-Wesley Publishing Company, Inc., Reading, Massachusetts.

Hollerbach, J. (1993). Advances in robot calibration. In *Robotics Research - The 6'th International Symposium*, pages 319–326.

Horn, B. K. P. (1987). Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A*, 4(4):629–642.

Jiang, H., Robb, R. A., and Holton, K. S. (1992). A new approach to 3-d registration of multimodality medical images by surface matching. In *Visualization in Biomedical Computing - SPIE Vol 1808*, pages 196–213. SPIE.

Johnson, A. E. and Hebert, M. (1996). Recognizing objects by matching oriented points. RI tech report CMU-RI-TR-96-04, Carnegie Mellon University, Robotics Institute, Pittsburgh, Pennsylvania 15213.

Kendall, M. G. and Stuart, A. (1977). *Canonical Variables*, chapter 43, pages 320–369. Griffin, London, 4th edition.

Kim, J. and Khosla, P. K. (1991). Dexterity measures for design and control of manipulators. In *Proceedings of the International Workshop on Intelligent Robots and Systems*, pages 758–763, Osaka, Japan. IEEE/RSJ.

Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220:671–680.

Kwoh, Y. S., Hou, J., Jonckheere, E. A., and Hayati, S. (1988). A robot with improved absolute positioning accuracy for CT guided stereotactic brain surgery. *IEEE Transactions on Biomedical Engineering*, 35(2):153–160.

Lavallee, S. (1995). Registration for computer-integrated surgery: Methodology, state of the art. In Taylor, R. H., Lavallee, S., Burdea, G. G., and Mosges, R., editors, *Computer-Integrated Surgery*, chapter 5, pages 77–97. The MIT Press, Cambridge, Massachusetts.

Lavallee, S., Sautot, P., Troccas, J., Cinquin, P., and Merloz, P. (1994). Computer assisted spine surgery: a technique for accurate transpedicular screw fixation using ct data and a 3-d optical localizer. In *Proceedings of the First International Symposium on Medical Robotics and Computer Assisted Surgery*, pages 315–322, Pittsburgh, PA.

Lavallee, S., Sautot, P., Troccaz, J., Cinquin, P., and Merloz, P. (1995). Computer-assisted spine surgery: A technique for accurate transpedicular screw fixation using CT data and a 3-d optical localizer. *Journal of Image Guided Surgery*, 1(1):65–73.

Lavallee, S. and Szeliski, R. (1995). Recovering the position and orientation of free-form objects from image contours using 3d distance maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(4):378–390.

Lavallee, S., Szeliski, R., and Brunie, L. (1991). Matching 3-d smooth surfaces with their 2-d projections using 3-d distance maps. In *SPIE Proceedings: Geometric Methods in Computer Vision*, pages 322–336, San Diego.

Lea, J. T., Santos-Munne, J. J., and Peshkin, M. A. (1995a). Diagramming registration connectivity and structure. *IEEE Engineering in Medicine and Biology Magazine*, 14(3):271–278.

Lea, J. T., Watkins, D., Mills, A., Peshkin, M. A., III, T. C. K., and Stulberg, S. D. (1995b). Registration and immobilization in robot-assisted surgery. *Journal of Image Guided Surgery*, 1(2):80–87.

Lowe, D. G. (1991). Fitting parameterized three-dimensional models to images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(5):441–50.

Maciunas, R. J., editor (1993). *Interactive Image-Guided Neurosurgery*. American Association of Neurological Surgeons.

Marcus, H. and Bourell, D. (1993). Solid free form fabrication. *Advanced Materials and Processes*, 144(3):28–35.

Masuda, T. and Yokoya, N. (1994). A robust method for registration and segmentation of multiple range images. In *Proceedings of the 2nd CAD-Based Vision Workshop*, pages 106–113.

Maurer, Jr., C. R. and Fitzpatrick, J. M. (1993). A review of medical image registration. In *Interactive Image-Guided Neurosurgery*, chapter 3, pages 17–44. American Association of Neurological Surgeons.

McCarthy, J. M. (1990). *An Introduction to Theoretical Kinematics*. The MIT Press, Cambridge, Massachusetts.

Menq, C., Yau, H., and Lai, G. (1992). Automated precision measurement of surface profile in CAD-directed inspection. *IEEE Transactions on Robotics and Automation*, 8(2):268–278.

Mishra, B. and Silver, N. (1989). Some discussion of static gripping and its stability. *IEEE Transactions on Systems, Man and Cybernetics*, 19(4):783–796.

Nahvi, A. and Hollerbach, J. (1996). The noise amplification index for optimal pose selection in robot calibration. In *Proceedings of IEEE International Conference on Robotics and Automation*, Minneapolis.

Nolte, L. P., Zamorano, L. J., Zhaowei, J., Wang, Q., Langlotz, F., Arm, E., and Visarius, H. (1994). A novel approach to computer assisted spine surgery. In *Proceedings of the First International Symposium on Medical Robotics and Computer Assisted Surgery*, pages 323–328, Pittsburgh, PA.

Northern Digital Inc. (1992). Accuracy of digitizing probes. Technical Report 3, Northern Digital Inc., Ontario, CAN.

Ohwovoriole, M. S. and Roth, B. (1981). An extension of screw theory. *ASME Journal of Mechanical Design*, 103:725–735.

Papoulis, A. (1984). *Probablily, Random Variables, and Stochastic Processes*. McGraw-Hill Book Company, Inc., New York, 2nd edition.

Peria, O., Francois-Joubert, A., Lavallee, S., Champleboux, G., Cinquin, P., and Grand, S. (1994). Accurate registration of SPECT and MR brain images of patients suffering from epilepsy or tumor. In *Proceedings of the First International Symposium on Medical Robotics and Computer Assisted Surgery*, pages 58–62, Pittsburgh, PA.

Radermacher, K., Staudte, H. W., and Rau, G. (1994). Computer assisted orthopedic surgery by means of individual templates - aspects and analysis of potential applications. In *Proceedings of the First International Symposium on Medical Robotics and Computer Assisted Surgery*, pages 42–48, Pittsburgh, PA.

Rehg, J. and Witkin, A. (1991). Visual tracking with deformation models. In *Proceedings of IEEE International Conference on Robotics and Automation*, pages 844–850, Sacramento, CA.

Sahoo, K. C. and Menq, C. H. (1988). Localization of 3d objects based on surface representation and tactile sensing. Technical Report ERC/NSM-88-20, The Ohio State University, Columbus, OH.

Sahoo, K. C. and Menq, C. H. (1991). Localization of 3d objects having complex sculptured surfaces using tactile sensing and surface description. *Journal of Engineering for Industry*, 113:85–92.

Sampson, P. D. (1982). Fitting conic sections to very scattered data: An iterative refinement of the Bookstein algorithm. *Computer Graphics and Image Processing*, 18:97–108.

Satava, R., Morgan, K., Sieburg, H., Mattheus, R., and Christensen, J., editors (1995). *Medicine Meets Virtual Reality III Proceedings*, San Diego, CA. IOS Press.

Sato, K. and Inokuchi, S. (1987). Range-imaging system utilizing nematic liquid crystal mask. In *Proc. ICCV*, pages 657–661, London, UK.

Schweikard, A., Tombropoulos, R., Adler, J. R., and Latombe, J. (1994). Planning for image-guided radiosurgery. In *Applications of Computer Vision in Medical Image Processing*, pages 96–101. AAAI.

Simon, D. A., O'Toole, R. V., Blackwell, M. K., Morgan, F., DiGioia, A. M., and Kanade, T. (1995). Accuracy validation in image-guided orthopaedic surgery. In *Proceedings of the Second International Symposium on Medical Robotics and Computer Assisted Surgery*, Baltimore.

Spoor, C. W. and Veldpaus, F. E. (1980). Rigid body motion calculated from spatial co-ordinates of markers. *J. Biomechanics*, 13:391–393.

Strang, G. (1980). *Linear Algebra and its Applications*. Academic Press, Inc., New York, 2nd edition.

Sugimoto, K. (1989). On the manipulability and singularity of manipulators. In *Robotics Research - The 5'th International Symposium*, pages 301–308.

Taubin, G. (1991). Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(11):1115–1138.

Taylor, R. H., Lavallee, S., Burdea, G. G., and Mosges, R., editors (1995). *Computer-Integrated Surgery*. The MIT Press, Cambridge, Massachusetts.

Taylor, R. H., Mittelstadt, B. D., Paul, H. A., Hanson, W., Kazanzides, P., Zuhars, J. F., Williamson, B., Musits, B. L., Glassman, E., and Bargar, W. L. (1994). An image-directed robotic system for precise orthopaedic surgery. *IEEE Transactions on Robotics and Automation*, 10(3):261–275.

Troccaz, J., Laieb, N., Vassal, P., Menguy, Y., Cinquin, P., Bolla, M., and Giraud, J. Y. (1994). Patient set-up optimization for external conformal radiotherapy. In *Proceedings of the First International Symposium on Medical Robotics and Computer Assisted Surgery*, pages 306–312, Pittsburgh, PA.

Uenohara, M. and Kanade, T. (1995). Vision-based object registration for real-time image overlay. In Ayache, N., editor, *Proceedings of the First International Conference on Computer Vision, Virtual Reality and Robotics in Medicine*, pages 13–22, Nice, France. Springer-Verlag.

van den Elsen, P. A., Pol, E.-J. D., and Viergever, M. A. (1993). Medical image matching - a review with classification. *IEEE Engineering in Medicine and Biology Magazine*, 12(1):26–39.

Wells, W. M., Viola, P., and Kikinis, R. (1995). Multi-modal volume registration by maximization of mutual information. In *Proceedings of the Second International Symposium on Medical Robotics and Computer Assisted Surgery*, pages 55–62, Baltimore.

Wheeler, M. D. (1996). *Automatic Modeling and Localization for Object Recognition*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA. CMS-CS-96-188.

Woods, R. P., Mazziota, J. C., and Cherry, S. (1993). MRI-PET registration with automated algorithm. *Journal of Computer Assisted Tomography*, 17:536–46.

Yamamoto, M. (1993). Direct estimation of range flow on deformable shape from a video rate range camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(1):82–89.

Yoshikawa, T. (1985). Manipulability of robotic mechanisms. *International Journal of Robotics Research*, 4(2):3–9.

Yoshikawa, T. (1990). *Foundations of Robotics: Analysis and Control*. The MIT Press, Cambridge, Massachusetts.

Zhang, Z. (1994). Iterative point matching for registration of free-form curves and surfaces. *The International Journal of Computer Vision*, 13(1):119–152.

Zhuang, H., Wang, K., and Roth, Z. S. (1994). Optimal selection of measurement configurations for robot calibration using simulated annealing. In *Proceedings of IEEE International Conference on Robotics and Automation*, pages 393–398, San Diego, CA.

Zhuang, X. and Huang, Y. (1994). Robust 3-D - 3-D pose estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(8):818–824.