



3 1293 00899 2780

This is to certify that the
dissertation entitled
INDOOR ROBOT NAVIGATION
USING A SYMBOLIC LANDMARK MAP

presented by
STEPHEN JOHN WALSH

has been accepted towards fulfillment
of the requirements for

PhD degree in Computer Science

Major professor

Date 16 Nov 92



PLACE IN RETURN BOX to remove this checkout from your record.
TO AVOID FINES return on or before date due.

DATE DUE	DATE DUE	DATE DUE
JAN 09 2004		

MSU Is An Affirmative Action/Equal Opportunity Institution

c:\cic\date due.pm3-p.1

**INDOOR ROBOT NAVIGATION USING A
SYMBOLIC LANDMARK MAP**

By

Stephen John Walsh

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Computer Science Department

1992

ABSTRACT

INDOOR ROBOT NAVIGATION USING A SYMBOLIC LANDMARK MAP

By

Stephen John Walsh

This dissertation addresses the problem of a mobile robot autonomously estimating its location within an *a priori* map using sensed information. A 2D indoor robot world is sparsely represented as a collection of vertical landmarks encoded as an attributed edge graph. This map is matched to data obtained from a single image with depth to feature information estimated by ultrasonic sensors.

The goal of this work is the rapid recovery of an estimated position within a previously mapped domain from an initial state of complete uncertainty. The structure provided by indoor environments permit geometric assumptions that enable rapid vertical landmark detection and classification. Pairs of vertical edges are extracted from monocular images and candidate landmarks identified. Methods of fusing ultrasonic estimated range data with extracted vertical edge features from a single image are developed. Experiments have demonstrated that the proposed methods classified vertical landmarks with 84 percent accuracy over a large library of domain data. Further experiments lead to the definition of a footprint for accurate ultrasonic sensing.

Detailed composite error models for the ultrasonic and imaging systems are developed and used to examine the performance of the proposed sensing geometries. Analysis and experiments lead to the definition of a general navigation envelope that enhances sensing accuracy while only mildly restricting travel and locomotion.

An independent suite of test data combined with the developed composite error models are used to conduct a large simulation study on the performance of this methodology in real and synthetic domains. Several heuristic methods are proposed for recovering from sensing errors. Simulation results show that a deterministic error recovery approach outperforms the heuristic methods in measures of speed and accuracy. In one large domain, position recovery is obtained in less than 12 seconds *in the absence of error* and in less than 15 seconds *with realistic modeled error*; results show 98 percent correct pose decisions. Similar time and accuracy performances are observed in other domains. The time performance is limited not by the solution approach, but by the mechanical limitations of the robot base. Travel speeds up to 30 feet per second could be supported.

To my Lord and Savior Jesus Christ
by whose grace I do everything however great or small

ACKNOWLEDGMENTS

I wish to thank my thesis advisor, Professor George Stockman, for his guidance during the execution of this research. His ideas were always helpful and often precipitated welcome progress. I particularly appreciate his ability to coach more performance out of me than I thought possible. His investment of many weekend and evening hours toward the progress of this work was greatly appreciated. Dr. Stockman serves as a positive model for me in many ways. I also acknowledge the selfless efforts of my thesis committee members, Professors Richard Dubes, Lal Tummala, John Forsyth and Chuck MacCluer, who made valuable suggestions toward improving this research. Additional thanks to Dr. Anil Jain for equipping and maintaining the Pattern Recognition and Image Processing (PRIP) laboratory with advanced research tools, including the hardware necessary for constructing Rome, the mobile robot used to investigate this work. I also want to thank Dr. Abdol Esfahanian for his assistance in pursuing this research. Thanks also go to Dr. Carl Page for inviting me to return and pursue my doctoral work, and to Dr. Harry Hedges of the National Science Foundation and Dr. Mo Jamshidi of the University of New Mexico for their encouragement. I also wish to thank Dr. Stockman's secretary, Lora Mae Higbee, for her patience and maternal guidance.

I am also grateful for the friendship and technical dialogue shared with many former and current PRIPers. I'm especially thankful to Dr. Greg Lee for his patient advice on so many topics, including the details of defense and dissertation preparation as well as being a true friend. I also thank Jon Courtney, Tim Newman, Hans Dulimarta, Qian Huang, Jinlong Chen, Dr. Deb Trytten, Sushil Bhattacharjee, Sharathcha Pankanti, Marie-Pierre Dubuisson, Philippe Ohanian, and Sally Howden. Their support, both personal and professional, has enabled and motivated this research to bear fruit. The friendship and technical assistance of Jon Engelsma and Christian Trefftz have also been instrumental to the completion of this dissertation. It is difficult to imagine this work being enjoyable in the absence of these fine individuals. We shared many cheers and hardships together; thank you my friends. Thanks also to my colleagues Reid Baldwin, Edgar Kalns, and Barb Czerny for their friendship and support over the years. My sincere appreciation also goes to the PRIP

manager, John Lees, for establishing and maintaining a stable research environment. Roxanne Peacock and Brian Wright were instrumental in helping provide the tiny, but essential hardware bits that held Rome together.

My professional family at the U.S. Air Force Academy has also been pivotal to the progress of this work. Thanks to Colonel Dan Litwhiler for supporting my fellowship here at Michigan State, and thanks also to Lt. Colonel Steve Gordon for both his encouragement and his guidance. I also thank Lt. Colonels Bill Kiele, Bill Skeith, and Major Jerry Johnson for all their help, support, and understanding along the way. Thank you also to Major Laurie Wells and Captains Bob Lawconnell, Don Duckro and Jay Jacobson who covered a few of my classes so that I could conclude portions of this work. And thanks to Captains Jason Moore, Chris Warack and Joe Hewitt who helped me become familiar with our local computing environment.

Finally, I must thank the pillars of my life. Thanks to my wife, Lori, and our children, John, Niki, and Eric, who have supported and loved me through many challenging times. I believe they have worked harder and suffered greater than I did during times of stress and strain. Thanks also to my parents, Dr. Bill and Dorothy Walsh, and my fellow siblings Bill and Martha, who together have shared my academic peaks and valleys. I couldn't have began, nor completed, this work without my family's unfailing love and support.

TABLE OF CONTENTS

LIST OF TABLES	x
LIST OF FIGURES	xii
1 Introduction to the Problem	1
1.1 Problem Statement	2
1.2 Robotic Navigation	3
1.3 Organization of this Dissertation	6
2 Literature Background and Motivation	8
2.1 Sensor Overview	9
2.2 Navigation Survey	11
2.2.1 Indoor Applications	11
2.2.2 Outdoor Applications	20
2.3 Pose Estimation	22
2.4 Sparta and Rome	27
2.5 Summary	29
3 The Where-am-I Problem	31
3.1 Navigating with a Map	31
3.2 Representation	35
3.2.1 Geometric Paradigm	36
3.2.2 Computational Geometry	37
3.2.3 Occupancy Grid	38
3.3 A New Approach - StickRep	38
3.4 Summary	41
4 Sensing Landmarks	44
4.1 Visual Sensing of Vertical Ribbons	45
4.1.1 Feature Extraction	46
4.1.2 Landmark Classification	51

4.2	Theory of Ultrasonic Sensing	52
4.3	Ultrasound Studies	55
4.4	Summary	57
5	Sensing Geometries	59
5.1	Motivation	60
5.2	Two Side Geometry	61
5.3	Short Side Geometry	65
5.4	Error Analysis	67
5.4.1	Error Analysis of Geometric Approximation	69
5.4.2	Composite Range Error Model	71
5.4.3	Linear First Order Error Model	72
5.5	Monte Carlo Experiments	74
5.6	Summary	82
6	Inexact Graph Matching	83
6.1	Representation	83
6.2	Ambiguity	85
6.3	Simulations	90
6.4	Error	91
6.4.1	No Error	93
6.4.2	Empirical Based Error	95
6.5	Metric Relaxation Matching	96
6.6	Heuristic Error Recovery Experiments	98
6.6.1	First Frame Error	99
6.6.2	False Door Error	99
6.6.3	Mis-sized Door Error	100
6.6.4	Full Recovery Suite	101
6.7	Deterministic Error Recovery	102
6.8	Examining Additional Domains	104
6.9	Summary	109
7	Summary, Conclusions and Recommendations for Future Research	110
7.1	Summary and Conclusions	110
7.2	Contributions	112
7.3	Recommendations for Future Research	114
A	Ultrasound Histograms	116

B Partial Derivatives from Sensing Geometries	128
BIBLIOGRAPHY	143

LIST OF TABLES

2.1	Logic for label combination	15
4.1	Landmark classification boundaries showing ribbon width in inches .	52
4.2	Experiment 1: Ultrasonic percent range error as a function of surface type and angle	58
4.3	Experiment 2: Ultrasonic percent range error as a function of surface type and angle	58
5.1	Two Side Geometry acoustic angles	65
5.2	Composite ranging error summary	71
5.3	Worst case landmark error using linear first order estimate	73
5.4	Accuracy resulting from reducing $\Delta\theta$ to 3.75° using Short Side Geometry	73
5.5	Short Side θ tolerance for error within two inches	74
5.6	Probability of Short Side Geometry measuring within 2 inches	75
6.1	Domain ambiguity: Number of edge pairs with adjacency angle of 90°	85
6.2	Domain ambiguity: Number of edge pairs with adjacency angle of 180°	85
6.3	Domain ambiguity: Number of edge pairs with adjacency angle of 270°	86
6.4	Domain ambiguity: Three Edges: $\langle door \rangle \langle wall \rangle \langle door \rangle$, adjacency an- gles of 180°	86
6.5	Domain ambiguity: Three Edges: $\langle wall \rangle \langle door \rangle \langle wall \rangle$, adjacency angles of 180°	86
6.6	Domain ambiguity: Four Edges: $\langle door \rangle \langle wall \rangle \langle door \rangle \langle wall \rangle$, adjacency angles of 180°	87
6.7	Domain ambiguity: Four Edges: $\langle wall \rangle \langle door \rangle \langle wall \rangle \langle door \rangle$, adjacency angles of 90°	87
6.8	Domain ambiguity: Door type census	88
6.9	Domain ambiguity: Wall type census	88
6.10	Empirical Based Error Model	95
6.11	Performance without heuristics	98

6.12 Performance with First Frame Heuristic	100
6.13 Performance with False Door Heuristic	100
6.14 Performance with Mis-sized Heuristic	101
6.15 Performance with Full Heuristic Suite	101
6.16 Performance of deterministic recovery in Engineering Building	107
6.17 Overall performance summary	107
6.18 Performance of deterministic recovery in Wells Hall	107
6.19 Performance of deterministic recovery in regular polygon	107
6.20 Performance of deterministic recovery along synthetic storefront . . .	108
6.21 Performance of deterministic recovery in a modified Wells Hall	108
6.22 Comparison of domain performance	108

LIST OF FIGURES

1.1	Sample indoor navigation domain: How should this be represented? .	4
1.2	Door feature	5
1.3	Ultrasonic sensing of distance to hallway walls	5
2.1	Plot of representative cell patterns. Heavily shaded cells denote occupied cells, cross hatched cells have conflict label assignments	16
2.2	Corrected plot of representative cell patterns from previous Figure . .	16
2.3	Shortest path in 2D polygonal space	19
2.4	Viewing geometry for P4P problem	25
2.5	Algorithm steps to recover global pose and focal length	26
2.6	Sparta	28
2.7	Rome	30
3.1	StickRep example	43
4.1	Vertical Edge Ribbon Extraction Algorithm	49
4.2	Image processing suite example. (a) Original image (b) Vertical edge image (c) Thresholded edge image (d) Smeared vertical projection with darkened feature footprint	50
5.1	Field of view geometry	62
5.2	Rome's sensing geometry	68
5.3	Linear first order error model	72
5.4	Short Side Geometry, $\theta = 90^\circ$	76
5.5	Short Side Geometry, $\theta = 70^\circ$	77
5.6	Short Side Geometry, $\theta = 50^\circ$	78
5.7	Two Side Geometry, $\theta = 90^\circ$	79
5.8	Two Side Geometry, $\theta = 70^\circ$	80
5.9	Two Side Geometry, $\theta = 50^\circ$	81
6.1	Starting location ambiguity of MSU Engineering Building	89

6.2	Simulation Algorithm	92
6.3	Unique pose without error	94
6.4	Metric Relaxation Heuristic	97
6.5	Multi-Sequence with error	103
A.1	Concrete surface, 45°	117
A.2	Concrete surface, 30°	118
A.3	Concrete surface, 15°	119
A.4	Concrete surface, 0°	120
A.5	Wood surface, 45°	121
A.6	Wood surface, 30°	122
A.7	Wood surface, 15°	123
A.8	Wood surface, 0°	124
A.9	Cloth surface, 30°	125
A.10	Cloth surface, 15°	126
A.11	Cloth surface, 0°	127
B.1	$\frac{\partial}{\partial RL}, \quad \phi = 0^\circ$	130
B.2	$\frac{\partial}{\partial RR}, \quad \phi = 0^\circ$	130
B.3	$\frac{\partial}{\partial LJ}, \quad \phi = 0^\circ$	131
B.4	$\frac{\partial}{\partial RJ}, \quad \phi = 0^\circ$	131
B.5	$\frac{\partial}{\partial RL}, \quad \phi = 20^\circ$	132
B.6	$\frac{\partial}{\partial RR}, \quad \phi = 20^\circ$	132
B.7	$\frac{\partial}{\partial LJ}, \quad \phi = 20^\circ$	133
B.8	$\frac{\partial}{\partial RJ}, \quad \phi = 20^\circ$	133
B.9	$\frac{\partial}{\partial RL}, \quad \phi = 40^\circ$	134
B.10	$\frac{\partial}{\partial RR}, \quad \phi = 40^\circ$	134
B.11	$\frac{\partial}{\partial LJ}, \quad \phi = 40^\circ$	135
B.12	$\frac{\partial}{\partial RJ}, \quad \phi = 40^\circ$	135
B.13	$\frac{\partial}{\partial sdis}, \quad \phi = 0^\circ$	137
B.14	$\frac{\partial}{\partial \theta}, \quad \phi = 0^\circ$	137
B.15	$\frac{\partial}{\partial LJ}, \quad \phi = 0^\circ$	138
B.16	$\frac{\partial}{\partial RJ}, \quad \phi = 0^\circ$	138
B.17	$\frac{\partial}{\partial sdis}, \quad \phi = 20^\circ$	139
B.18	$\frac{\partial}{\partial \theta}, \quad \phi = 20^\circ$	139
B.19	$\frac{\partial}{\partial LJ}, \quad \phi = 20^\circ$	140
B.20	$\frac{\partial}{\partial RJ}, \quad \phi = 20^\circ$	140
B.21	$\frac{\partial}{\partial sdis}, \quad \phi = 40^\circ$	141

B.22	$\frac{\partial}{\partial \theta}, \quad \phi = 40^\circ$		141
B.23	$\frac{\partial}{\partial LJ}, \quad \phi = 40^\circ$		142
B.24	$\frac{\partial}{\partial RJ}, \quad \phi = 40^\circ$		142

CHAPTER 1

Introduction to the Problem

Robotics is a rapidly expanding field with a rich research landscape. Multisensor mobile robots are navigating through academic and industrial buildings and ambitious outdoor efforts are presently under study. A central requirement towards the development of capable and flexible robotic systems is the ability to develop robust spatial descriptions of their environment from sensory information, and to employ this information in planning and solving application tasks. These capabilities enable the robot to interact lucidly with its domain, both by interpreting its sensory data to draw appropriate conclusions for near term decisions, and by diagnosing and maintaining an adequate domain model for strategic planning and decisions. Systems with little or no sensing capabilities are limited to fixed sequence operations in highly controlled work areas, and cannot perform activities with any significant degree of autonomy or adaptability.

To successfully achieve these capabilities in robotic systems, several research issues must be addressed. These include fusion of information from multiple, and dissimilar sensors, problems in sensor interpretation, mapping and representing the navigator's domain, handling sensor uncertainty and errors, landmark identification, position estimation, and registering *a priori* maps with sensed information. In this dissertation we touch upon all of these issues to varying degrees in our approach to solving

a particularly challenging robot navigation task - pose recovery. Pose recovery is the estimation of the navigator's position after initial, and complete disorientation within its operating domain. A long term goal of the research presented in this dissertation is the development of minimal, yet robust mapping and navigation tools for mobile robots operating in and exploring structured domains.

In the following section, we will present an overview of the problem studied in this dissertation. A brief discussion of robot navigation and an outline of the organization of this dissertation close the chapter.

1.1 Problem Statement

To widen the application range and deployment of both industrial and research robots, a key requirement is the development of systems with greater autonomy, able to sense, plan and operate in structured environments with minimal prior knowledge. To achieve this level of independence, the robot must not only be able to accurately sense its domain, but must be able to accurately sense its *pose* in its domain. Pose estimation is a base problem to all navigation tasks. Many different paradigms exist for different applications, but all use some sensory information to overcome dead reckoning error. To be robust in any environment, a robot navigator must know where it is relative to major objects in its domain, to some degree of accuracy. This task is known as The Where-am-I Problem.

The Where-am-I Problem : *The autonomous recovery of an estimated position within a previously known, or mapped operating domain, from an initial state of complete translational and rotational uncertainty.*

Solving the Where-am-I Problem is more difficult than maintaining an estimated pose during navigation from an initial *known* position. However, a successful Where-am-I Solution will also adequately address pose maintenance. The autonomous ability

to accurately sense, and represent an operating environment are pivotal for successfully addressing this problem. In this dissertation we present our solution to the Where-am-I Problem for a class of indoor domains.

1.2 Robotic Navigation

A robotic navigator must have some mechanism for representing its surrounding environment. If not, it cannot truly navigate, but merely avoid obstacles, and its application will be specialized and limited. Domain representations, or maps, do not have to be provided as an *a priori* database, but may be constructed from sensory information obtained by the robot. What domain features are important for navigation, and how can they be encoded into a representation that enables self-location in an operating domain such as the building in Figure 1.1? Building a mobile robot application solely upon an *a priori* representation is considered fragile even within structured, indoor domains, because it fails to accommodate the inherent environmental dynamics associated with people. At a minimum, applications employing a static domain model must still sense to avoid unpredictable obstacles such as pedestrians or temporary objects. But static domain approaches fail to update these dynamic events into their representational schemes, and often fail in forgiving, but dynamic domains such as a business office [56, 156]. We will discuss the opinions of static and dynamic domain modeling in detail in the next chapter.

How much domain knowledge is necessary for navigation? This question is valid whether employing a static or dynamic representation, or whether the application constructs its own map or employs an *a priori* map. How can *a priori* domain features such as the door in Figure 1.2 be extracted and represented? What sensors are best suited for domain feature extraction? Can inexpensive sensors such as the ultrasonic ranging system shown in Figure 1.3 reliably extract feature information? We were

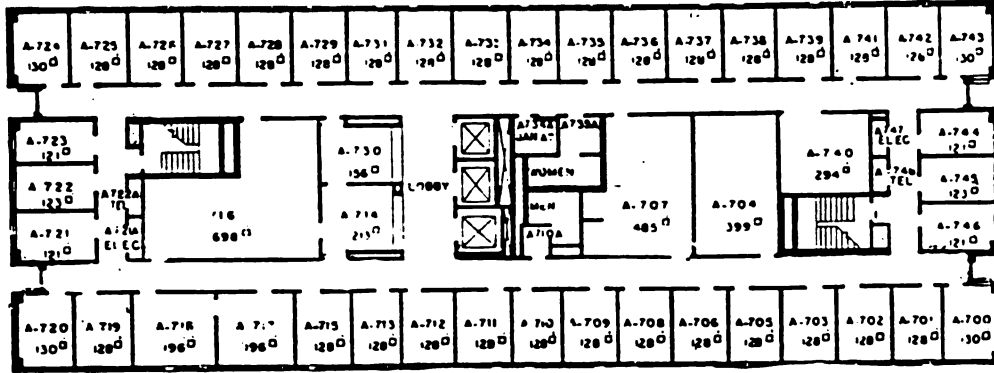


Figure 1.1. Sample indoor navigation domain: How should this be represented?

motivated by a sense of economy and desired to explore a minimalist approach. We did not wish our solution to revolve around a highly accurate metric map, because we did not feel this was necessary, and we desired real time performance using modest hardware and sensors. Human navigation operates at a higher level, and is more sensitive to feature landmarks than fine metric data [92]. Landmark based navigation will be more closely examined in Section 2.3.

Robust sensing and dynamic representations are certainly desirable system attributes. Many researchers would even argue that these attributes are indeed requirements. Feature extraction, landmark classification, and error handling are often employed in support of these goals. But these desirable sensing and matching capabilities may sometimes oppose the rapid, accurate pose recovery that is our research goal. These are some of the research issues that frame the work we present in this dissertation. Many of these issues that arise are not dealt with exhaustively, and much room for further research and exploration remains.

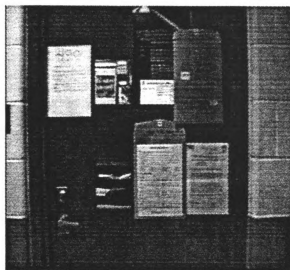


Figure 1.2. Door feature

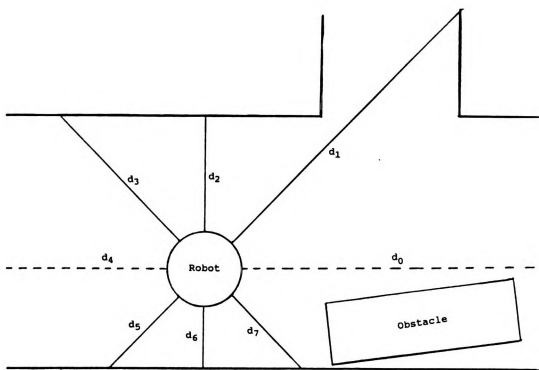


Figure 1.3. Ultrasonic sensing of distance to hallway walls

1.3 Organization of this Dissertation

The remainder of this dissertation is organized as follows. Chapter 2 examines the attributes and liabilities of commonly employed robotic sensors. A thorough discussion on the performance and approach of several research efforts from the mobile robotic literature, including both indoor and outdoor applications, will be presented. We will also survey competing methodologies to pose estimation, and examine the local efforts that preceeded our work.

In Chapter 3, we discuss the Where-am-I Problem and the various methodologies that address it. An examination of several approaches to domain representation are presented as well as StickRep, our chosen representation. Strengths and weaknesses are compared with respect to our desired solution capabilities.

An examination of landmark sensing is presented in Chapter 4. We will discuss the development of our vertical edge ribbon detector, including the performance of our feature extraction methods and the landmark classification performed. The theory of ultrasonic sensing and an exhaustive ultrasonic characteristics experiment are presented. The need for an accurate ultrasonic sensing footprint is motivated from these results.

In Chapter 5, we present two sensing geometries, *Two Side* and *Short Side*, that perform as accurate sensing frameworks. We will see that these geometries offer real time capability that extend beyond our present application to more general situations. Extensive error analysis, Monte Carlo experiments and a large simulation study are presented to define several composite error models for our data acquisition. These results suggest that our Short Side Geometry is best suited for our solution approach.

Domain ambiguity and the representational challenges that it presents are discussed in Chapter 6. Large scale simulations are presented that illustrate the challenge of the Where-am-I Problem even without the presence of sensing error. Further

heuristic error recovery experiments are compared using the developed error models, and our metric relaxation technique is discussed. Many of these heuristic approaches offer adequate pose recovery, but our deterministic error recovery experiment demonstrates the strongest solution performance in the chosen domain.

Chapter 7 concludes the dissertation by summarizing the contributions of our work and offering suggestions for future exploration.

CHAPTER 2

Literature Background and Motivation

To open up the scope of applications for both industrial and research mobile robots, greater autonomy is needed. Unstructured environments present a difficult challenge, and most application designers have responded by providing detailed *a priori* information to the mobile robot. Digital contour maps for outdoor applications [39], or encoded infrared beacons [110], and bar-coded floor stripes for indoor robots [61], are examples of providing reference information to enable the robot to determine its location in the world. To gain greater independence, the robot must acquire an understanding of what's around it, by capturing and developing an adequate environmental model. Robots with restricted sensing abilities are limited to highly structured, and sequential operations. A variety of complimentary sensors and a mechanism for extracting salient information have been widely successful in enabling autonomous operations in structured environments. Determining the presence and location of objects is a primary task in robotic navigation. Most of this sensory information will have to be compiled from multiple sensors, and a world model then constructed. This model can then serve as a basis for path planning, obstacle avoidance, landmark identification, position estimation and other essential operations. Constructing such

a model starts with the complex task of determining range information from sensing the world. Stereo vision is the most popular passive sensor with mobile robots, and ultrasonic and infrared sensors are the most broadly used active sensors.

We will briefly examine the merits of the sensors most common in robotic applications, and then survey some of the latest research and industrial applications whose results could most readily contribute to our local efforts. We will primarily focus on vision-based systems, although most applications combine stereo vision with active sensors.

2.1 Sensor Overview

Ultrasound technology has been with us for some time, but few previous applications built detailed maps. Ultrasonic sensors have now become widely used in mobile robot applications. Their main attraction is their simplicity, low cost, and the fact that distance measurements can be determined directly. A typical sensor has a useful measuring range of 1 to 35 feet, with a beam width of 15 degrees, so an array, or belt of sensors are often employed [52]. Early robotic applications include simple distance measurements [63], localizing object surfaces [24], and determining the world position of a robot [29]. In a later effort, ultrasonic sensors were used to determine a robot's position by comparing obtained distance measurements against an accurate map. This approach didn't account for the range errors that are typical in ultrasonic data [111, 112]. A subsequent method also used an onboard map, but attempted to handle noisy range data [45]. A more sophisticated effort pursued ultrasonic mapping and navigation by using a narrow beam to build a line-based description of the robot's world [36, 37]. The ultrasound readings were interpreted by fitting line segments to the detected points, and then comparing these to an *a priori* map. Noisy data once again plagued the approach. A recent effort explicitly represents this inherent

in
ap
po
S
na
in
ory

uncertainty [38]. It dynamically maintains a description of free space limits, and matches observations to a world model. A major effort overcame these earlier troubles by building its own map through a probabilistic fusion of successive readings [50]. No *a priori* information was used. A controlling microprocessor selected and fired the sensors, timed the returns and provided the corresponding range value. Other similar systems are now maturing [91].

Stereo vision systems have been traditionally used to extract depth information from two images [114]. The major difficulty of employing this technique in real world environments is the intrinsic computational expense of extracting 3-dimensional (3D) information from stereo-image pairs which limits the number of points that may be tracked [2, 16, 17, 43, 113, 133]. Real time constraints have previously precluded dense 3D descriptions, but later approaches have obtained promising results [4, 23, 120]. Traditional stereo systems have relied on high-contrast edges or points that could be easily tracked through several images [100, 115]. Most practical real world vision navigation systems only build sparse depth maps as a result of this constraint [107, 114, 145]. They select points to be matched and tracked using an interest operator, and handle 30 to 50 points, generating a 3D map in 30-50 seconds on a VAX 11/780. It's predicted that 25 percent of all robots sold this year will incorporate some type of vision system [10].

Infrared (IR) sensors are another active sensor that can directly determine proximity information. They usually complement other more capable sensors, but one approach proposes encoded IR beacons to estimate position [110]. Signaling specific pose information is not limited to the IR medium. Radio transponders costing just \$1 have now become available that may broadcast a specific domain location to a navigator. Such a manufactured landmark can be used to provide pose information in difficult, ambiguous domains. More sophisticated transponders possess the memory to record the messages of passing navigators in an environment where multiple

robots are operating. This emerging technology represents a low cost solution to our problem in domains where their use is practical, and could be particularly attractive to industry.

Laser range finders are an expensive answer to obtaining direct range information. The maximum detectable phase shift limits typical range to 64 feet. A field of view of 30 vertical degrees by 80 horizontal degrees has been achieved [39]. This sensor class is maturing rapidly with many new sensors entering the market. Models with finer resolution for indoor work are now available [10].

2.2 Navigation Survey

We will now examine in some depth the more interesting efforts of applied dynamic path planning over the last couple of years. These summaries are divided into indoor and outdoor environments. The structure provided by an indoor robot world greatly simplifies the navigational challenges. Path planning and obstacle avoidance often become a 2-dimensional (2D) problem. The additional tasks imposed by an outdoor, 3D robot world have been addressed by more complex navigational paradigms, but have limited applicability to our present efforts at Michigan State.

2.2.1 Indoor Applications

Elfes's Dolphin system is a major effort, perhaps the first to construct dense ultrasonic maps without any *a priori* maps or feature data [50, 52]. He felt that the matching errors typical in stereo vision applications were too serious a challenge to build a base robotic navigation system upon. He was concerned about how ultrasound and laser range finders were coarse and insensitive to surface characteristics. The ultrasonic sensors that were available for mapping unstructured environments are quite noisy, so he built a probabilistic occupancy system to facilitate the combination of multiple

sensor readings on the same location. He designed a grid system that assigned a probability of occupied or empty to each square. The grids could be as fine as 0.1 foot or as coarse as 1 foot square, but they were uniform throughout the robot world. To derive a probability for each grid space given a particular ultrasonic reading, Elfes defined:

- P the grid cell of interest
- R the range measurement returned by ultrasonic sensor
- R_{min} the smallest range value accepted from ultrasonic sensor
- ϵ the mean ultrasonic error
- ω the beam aperture
- S the position of ultrasonic sensor
- δ the distance from S to P
- θ the angle between the acoustic axis and segment SP

The area illuminated by the ultrasonic beam is modeled as two regions with different probability formulas for each:

Empty Region: points where $\delta < R - \epsilon$ and $\theta \leq \frac{\omega}{2}$

Occupied Region: points where $\epsilon < |R - \delta|$ and $\theta \leq \frac{\omega}{2}$

One probability is computed for angle and one for range. Whether the point of interest is in the occupied or empty region, the angle probability is:

$$P_L = 1 - \left(\frac{2\theta}{\omega}\right)^2$$

In the occupied region the range probability is:

$$P_o = 1 - \left(\frac{\delta - R}{\epsilon}\right)^2$$

In the empty region the range probability is:

$$P_e = 1 - \left(\frac{\delta - R_{min}}{R - \epsilon - R_{min}}\right)^2$$

The occupied or empty probability for each cell is calculated by multiplying its angular by its range probability. Successive ultrasonic readings were additively combined with previously collected data to capture an evolving picture of the world. He found that

these ultrasonic maps were produced an order of magnitude faster than those from stereo vision. He had great success in actual tests using this system, both inside and outside. Michigan State's Sparta system modified Elfes' approach to better handle rapidly changing situations, and an occasional bad ultrasonic return [134].

Matthies' effort built upon the occupancy grid system to incorporate stereo vision [107]. Matthies liked the occupancy grid system because it uniformly treated sensory data and handled the uncertainty of the robot's position. However, ultrasound couldn't see through clutter like stereo vision could. Ultrasound gives better detection of broad object surfaces and indicates large empty areas, but gives less definition to surface boundaries and fine surface structure. So he integrated a stereo vision system onto the system pioneered by Elfes. He extracted near-vertical edges using a Canny edge detector [28], and independently matched five scanlines centered on the robot's horizon looking for features that extend across all five. He used these edges to constrain the range of the search. The result of this process is a set of edge points on the horizon line of known depth. He joined these points with line segments to create a depth profile that approximates the scene surface structure. Matthies points out that combining these two sensor systems leads to one of three situations: complementation, correction, or conflict. Ultrasound made strong statements about emptiness of regions, but weaker statements about occupied areas. Stereo statements can be weak or strong, depending on the distribution or distance to features. When sensors conflict strongly about a region's status, he marks the grid unknown, for later determination. Maybe the most salient feature of the occupancy grid system is that it provided a means to combine raw data of totally different natures. The sensor data have qualitatively different information encoded in the range information they provide, and this prevents a simple analytic, or geometric integration approach towards building a coherent world model.

Matthies next tackled error modeling in stereo vision-based navigation [108]. His

previous system, with stereo vision-based navigation that tracked landmarks, relied on scalar models of triangulation measurement error. Here, he developed a 3D Gaussian error model, using ellipsoids to describe the 3D geometry resulting from constant probability of error contours. Scalar-weighted error models were motivated by the fact that uncertainty grows with distance [154]. This corresponds to a spherical model, but it doesn't capture the skewness and orientation of the uncertainty. With the ellipsoidal model, for nearby points, the contours will be almost spherical, but as the points get farther away, the contours grow more eccentric. He experimented with their robot using each model. He found that with the ellipsoidal model, the robot estimated its final position to within two percent of distance traveled, and one degree of orientation. Using the spherical model, accuracy was within eight percent of distance traveled, and seven degrees orientation. This work addresses an area of significant need; position uncertainty is a base problem for all navigation tasks.

Beckerman and Oblow modified the occupancy grid method to address the systematic nature of ultrasound errors [9]. Their thesis is that systematic errors are dominant compared to random errors when using wide-angle ultrasonic sensors. They observed that the occupancy grid, or stochastic approach, were local in character, reducing the label for a pixel based on information pertaining only to the single pixel. Beckerman and Oblow's method is nonlocal in character, using information about neighboring pixels and maintaining self-consistency between occupied and empty space. They label grid cells with one of four labels:

$$E = \textit{Empty} \quad O = \textit{Occupied} \quad C = \textit{Conflict} \quad U = \textit{Unknown}$$

Conflicts arise from different labels from two or more ultrasonic returns. Table 2.1 shows how labels are propagated through overlapping ultrasound scans. Beckerman

and Obrow implemented this cell-by-cell multiplication using the label representation

$$C = 0 \quad U = 1 \quad O = 2 \quad E = 3$$

and then quickly evaluate the 16 binary products to determine the new cell label C from intersecting labels A with B.

$$L_{ij}^A \cap L_{ij}^B = L_{ij}^C$$

$$if \quad L_{ij}^A = L_{ij}^B \implies L_{ij}^C = L_{ij}^A$$

$$else \quad L_{ij}^C = L_{ij}^A \times L_{ij}^B \mod 6$$

This logic facilitates rapid map construction and error attenuation. Figures 2.1 and 2.2, taken from [9] illustrate typical conflict resolution. Systematic errors are washed through a conflict resolving relabeling scheme. Most of the new labels are obtained by examining the characteristic patterns of conflict that result during scanning and processing. Their results indicated that despite collecting 80 percent less data than a stochastic method, the resolution of their maps was superior. The direct multiplication scheme updated cell labels much faster than the probabistic implementation of

		L^B			
		C	U	O	E
L^A	C	C	C	C	C
	U	C	U	O	E
	O	C	O	O	C
	E	C	E	C	E

Table 2.1. Logic for label combination

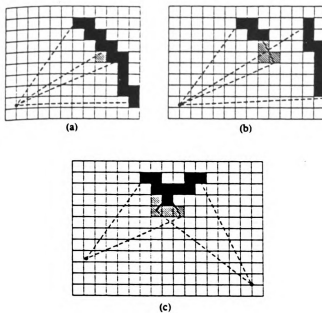


Figure 2.1. Plot of representative cell patterns. Heavily shaded cells denote occupied cells, cross hatched cells have conflict label assignments

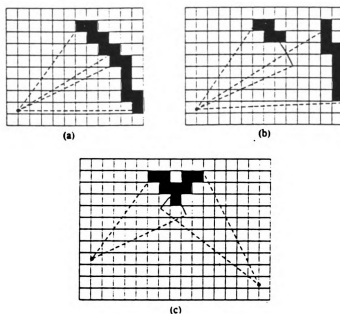


Figure 2.2. Corrected plot of representative cell patterns from previous Figure

an occupancy grid approach [50, 107, 114, 134, 151].

The work of Clark and his colleagues in developing the Harvard Head is also of great interest [32]. He implemented a motion control system to provide modal control of attention of a binocular vision head with seven degrees of freedom. He noted that visual tasks require the movement of the eyes to closely examine areas of interest for the particular task, paying little attention to the rest of the scene which is viewed peripherally. A given visual task might require the corners of an object to be detected, while another task may require that the object color be determined. Each case needs different features attended to. He defines attentive visual control in two parts: Decide where in the visual scene to attend on, and secondly, decide which motions are required to redirect the visual sensors toward that location. The control method is based on a two-level modal control technique proposed by Brockett [19]. The outer level controls the focus of attention, determining what features are going to be used to determine where to look next. The inner level directly controls camera pose through the driving motors. This is based on the human oculomotor control system. Their feature detection and localization is done in a special image processing system made by Datacube [123]. Their first experiment tracked blobs of a specific intensity range. Objects 0.5 to 2 meters from the head were fixated on, to within two pixels, and followed. The vision system could process five frames per second, but with communication delays, only three frames per second were realized. Their next experiment located geometric shapes by using the first three moments and intensity value as features. The scene was segmented with a connected components routine and then a saliency map was built. His present vision system is not yet optimized, and it takes one to ten seconds to find and fixate on the most salient object. Krotkov is now developing a similar system [89, 90].

Hung and his colleagues used multiple marks for determining 3D robot location in a complex industrial setting [75]. They placed the marks so that at least one could

be seen by the camera from any location. They integrated a pattern recognition technique with the 3D geometrical transformations so that all viewing angles could be handled. Their experimental results showed less than 3 percent average location error. This concept of placing artificial landmarks is very simple and flexible in many applications.

Frohn built VISOCAR, a mobile industrial robot, by breaking the navigational task into a hierarchy of goals obtained by functionally independent modules [61]. He constructed a hierarchy of capabilities, not a hierarchy of image processing steps. He used optical tracking of barcoded lane boundaries, and used the landmarks on the lane boundaries to update his position estimate against his onboard landmark map. His lane tracker is a fast adaptive correlation algorithm based on a proposal by Barnea [8]. He has an independent ultrasonic obstacle detector. The complete system is controlled by a M68000 microprocessor. VISOCAR appears remarkably similar to Litton Industries' Integrator.

Davis and his colleagues at Maryland are investigating a class of vision-based problems on a Connection Machine [40]. Their RAMBO project is a testbed for exploring efficient image processing and analysis, visual tracking, and visual planning. Pose estimation is one of their current interests. The independent context in which they're tackling these problems warrants interest.

Sharir and his colleagues have extensively pursued geometric path planning algorithms [135, 137]. Figure 2.3, taken from [137], illustrates the polygonal geometry. They assume that complete information is known about a static environment characterized by bounding polyhedral surfaces and objects. In a 2D application they have developed an $O(n^2 \log n)$ algorithm that guarantees to find the shortest path. In 2D space this shortest path between start and goal must be a polygonal line whose vertices are the start, goal, and the corners of the polygonal obstacles. The 2D problem is solved using a visibility graph to determine which vertices are visible

from
rou
one
Vo
ity
e

or
and p
Graz
the com

filtering is used to fuse numeric features into higher level models, and to derive self-calibration transforms. Explicit representation of data accuracy is propagated to every level of his perception process. The two sensor types operate in harmony, with the laser range finders flagging the presence of a surface while stereo edges determine the surface's location and extent. Final 3D scene modeling is done with a set of planar faces. His experimental work indicates strong promise for this modeling to assist higher-level robotic processes such as localization and scene interpretation.

2.2.2 Outdoor Applications

We now move our scope to outdoor systems. DARPA and NASA are fueling a great deal of work in outdoor systems. We won't attempt to cover subjects in any depth, but outdoor systems have delivered some significant results. There are many lessons in how these efforts approached their navigational challenges and what limitations they reluctantly accepted.

The prototype legged vehicle that may someday explore the Martian surface, called Ambler, completely avoids vision-based approaches [71]. Hebert and his colleagues opted for active range sensors because of fewer necessary computations and insensitivity to illumination conditions. The interplanetary communication lag here mandates fully autonomous operation and these researchers felt that vision was too expensive computationally.

Intractability is a recurring theme, even in relatively massive, distributed, autonomous land systems such as the ALV [48, 97, 157]. There is general difficulty in getting real time sensor-based control in unstructured environments [39, 115]. Advanced modeling techniques such as temporal stability offer just the advantage that an outdoor environment demands, but it aggravates the processing bottleneck [13]. Very complex systems even predict the appearance and disappearance of landmarks using dynamic model matching [117]. A knowledge-based landmark recognition system uses

an *a priori* map, perceptual knowledge and spatial reasoning. Model-based vision tries to reduce the computational complexity. Despite all the domain knowledge, ALV video processing is still extremely challenging [39].

If a structured road is to be followed, a fast pipeline can be laid between sensing and acting, creating a model of the perceived environment and then generating a control plan [20, 69, 70, 122, 153]. Independent parallel processes have been used to produce coherent behavior and faster response. Road modeling by spline functions and edge detection by linear recursive filters have also been used successively, achieving processing rates of 14 seconds per frame on a VAX 11/780 [25]. Other approaches match road geometry [103], or intensity [41]. These methods lend themselves well to parallel processing. Kluge makes a strong case for using strong explicit models to achieve reliable recognition [85]. He feels that there still isn't a reliable road tracking vision system despite vigorous research. Weak models with weak, or nonexistent higher level processes make them brittle to lost features or illumination changes. His thesis is that assumptions made in road modeling must be available for program access and modification. The SCARF system is a color-based, domain-modeling bicycle path follower. It proved quite reliant on known road shape, and was sensitive to illumination changes. Kluge's colleagues had trouble in constructing an explicit model since there were few inherent features. Their latest system, FERMI, uses five trackers to build explicit models with strong constraints. It uses higher level reasoning about the road. There will be two techniques of tracker fusion in their developing effort.

Maryland's system is arguably the best, and is based on using Hough transforms to find and group edges. But even their methodology fails with strong, straight shadow edges from trees and buildings, and it is sensitive to illumination changes [157]. The VITS system has followed roads at speeds up to 20 kph and detected and avoided obstacles. General capability was sacrificed for speed [152]. Dickmanns' Mercedes van ran as fast as 100 kph on the autobahn [44]. His extremely simple perception

model uses a monochrome camera and a simple edge detector. The system attempts to discard distracting edges, but the trackers can get fooled by shadows, puddles, road imperfections and varying illumination.

One of the lessons that is being learned by navigating outdoors is that an unstructured environment is extremely demanding on domain models, and processing capabilities, even with complex, multisensory approaches. Current efforts are rich in knowledge-based domain reasoning, parallel processing and direct ranging. All these systems incorporate a massive degree of *a priori* information. Successful indoor systems have been realized without using any of these tools, but extending these early indoor efforts may require adopting the discoveries from outdoor research.

2.3 Pose Estimation

Determining the position and orientation, or pose, of a robot camera is a problem that has been examined by many researchers. It's a pivotal problem since it's difficult to navigate well with pose uncertainty. If dead-reckoning travel were error-free, pose estimation wouldn't be an interesting issue, but unfortunately, this is far from the case. Pose estimation has become the cornerstone task in most mobile robotic applications.

Fischler and Bolles published a watershed paper that initially examined the Perspective-3-Point problem (P3P) [59]. In this problem the task is to determine the point in 3D space from which an image of known landmarks was obtained. This method seeks the correspondence between n image points and n control points, where $n = 3$ is the smallest number of landmarks. They characterized the P3P problem and provided a closed-form solution, but showed that it could yield as many as four solutions. The issue of the maximum number of possible solutions for the P4P and P5P problems remains open, but the authors proved that the P6P problem provides a

unique solution. Their most important result was that the P4P problem yields a single solution if the control points are coplanar. We shall examine some very interesting work on the P4P problem due to Abidi later [1].

Krotkov follows and extends Sugihara's work [143], pursuing location estimation from a single image frame [88]. A motive for this approach is to avoid the difficult reconstruction problem by using just one image. Here he matches rays to vertical edges taken from the image to an *a priori* landmark model. This approach uses the excellent angular resolution of CCD cameras, yet avoids 3D reconstruction issues and feature correspondance. Image processing beyond edge detection is avoided to allow more time to perform the landmark interpretation-tree search. Algorithm complexity is $O(n^4)$ in time and $O(n)$ in space. Noisy rays can be handled by giving up some accuracy for the sake of robustness. Spurious landmarks are handled by introducing a null model landmark. There is great potential for color here. Establishing color correspondence before searching the interpretation tree would decrease the time complexity by $O(n)$. If two color landmarks were extracted, complexity would go down $O(n^2)$.

Sarachik approaches the self-localization problem from a different angle [133]. She bases her work on three assumptions: rectangular rooms, inherent features at wall/ceiling junction, and most importantly, flat, uniform-height ceilings. These assumptions are reasonable in many indoor environments. Her uncalibrated stereo cameras are mounted with the line of sight about 45 degrees above the horizon. She is trying to take note of only those features that never change - the position of the walls. Her approach starts with the robot spinning in place taking multiple images, and using a single, composite image it determines where it is facing a wall head-on. It then uses a 1-dimensional strip from each of the cameras to match edges, and notes the vertical offset between the wall and ceiling. It is looking for the edge of greatest depth. It then images the same ceiling-edge pair from many vantage points

in the room and determines the upward tilt of each camera. This angle provides enough data to calculate the dimensions of the room, scaled by the unknown, but constant ceiling height. Performance of her robot yielded only 75 percent reliable self-calibration, and thus had trouble consistently determining room size. The failures in accurately determining the distance to the walls were the result of incorrect matches. Improved robustness in matching, and an ability to prune bad data points are being pursued to enhance results. The combination of redundant sensors and probabilistic integration could yield large improvements. Sarachik's method of navigation seems quite appropriate in environments that are busy and dynamic. Room identification by size or position of the door would be interesting to pursue.

Haralick developed a closed form solution to the P4P problem assuming both coplanar rectangular marks and a known focal length [68]. The planar geometries of the 2D perspective projection resulting from a rectangular mark determines the camera viewing parameters in the 3D world. His approach's only ambiguity in the determination is whether the camera is looking up and seeing the rectangle from below on one side, or whether the camera is peering down and seeing the target rectangle from above on the other side.

Kite developed a similar closed-form geometric approach that also presumed a coplanar target rectangle and a known focal length [83]. He combined the two solutions using heuristics based on human vision and determined the unique 3D pose of the camera.

Two other researchers with dissimilar applications represent different approaches to the same pose estimation problem. Ray describes a model-based vision method for tasks where approximate object position estimates are known [127]. Geometric models for objects in a tight workspace and object position estimates are used to predict where linear image features are expected to appear in each view. An object's pose is then estimated as the position which optimizes a figure-of-merit function which

describes correspondence between the observed features and the features predicted from the estimated position.

Han determines pose using tree annealing [67]. He formulates pose determination as a nonlinear optimization problem. The criterion function will turn out to possess local minima, and he deals with these by using a technique called tree annealing. His simulations indicated that average computing times of around 150 seconds are to be expected, so this method seems to offer little for real time applications.

Abidi introduced a new pose estimation technique based on the volume of the tetrahedra formed by three corners from the rectangular target and the lens center. Figures 2.4 and 2.5, taken from [1], illustrate the situation with the target on the right, and summarize the algorithm stages. There are actually 12 ways to calculate

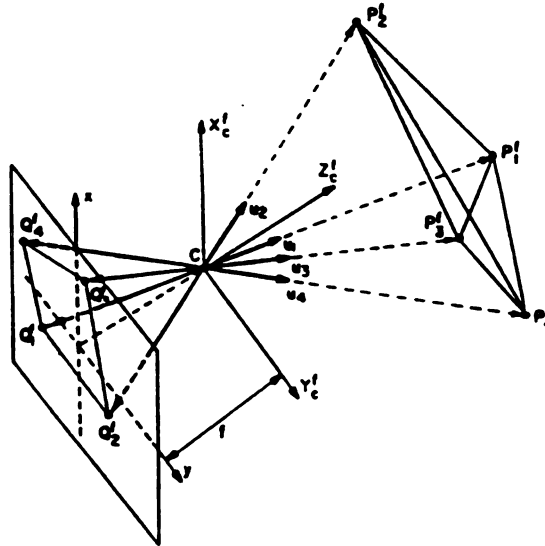


Figure 2.4. Viewing geometry for P4P problem

the focal length so all are found, and the median is used by the remaining algorithm stages. The importance of this effort is that it doesn't presume the focal length of the

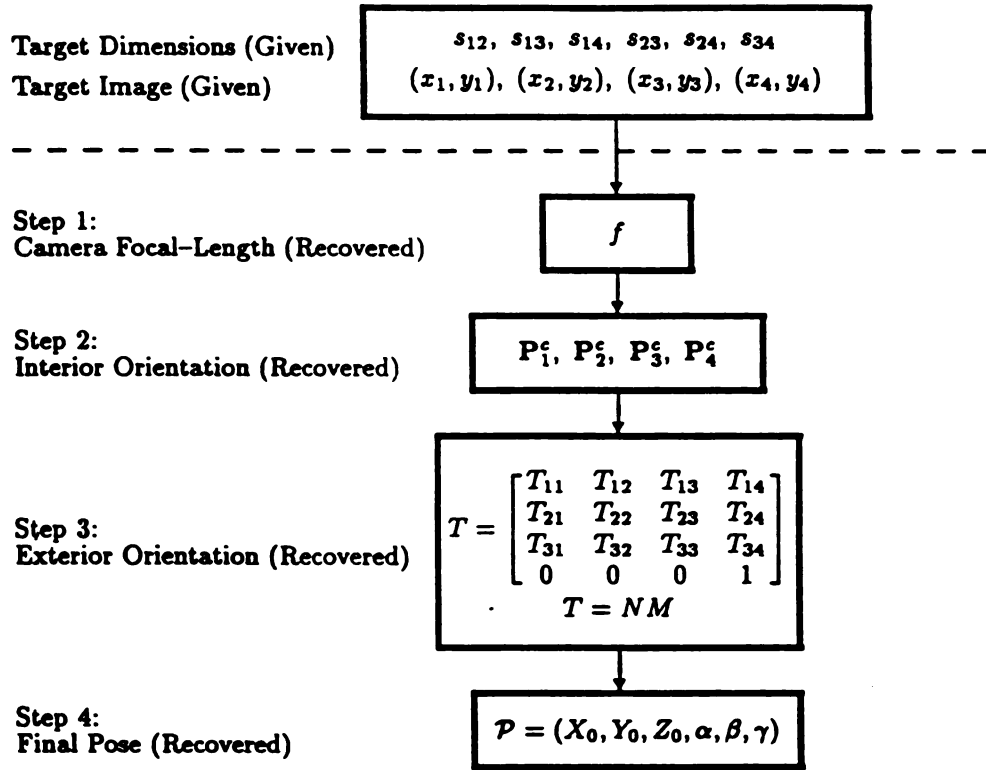


Figure 2.5. Algorithm steps to recover global pose and focal length

camera. It uses the redundancy in the inherent geometry to minimize error in pose recovery. The target dimensions are known and the only other inputs are the target coordinates on the image plane. It calculates seven outputs, the 3D global position, 3D orientation, and the effective focal length. The geometry also yields 6 redundant measures of distances to the target, and again the medians are used. A singularity exists when the target is exactly parallel to the image plane. It has also been shown to be sensitive to the exact coordinates of the image points. This sensitivity limited capability as the distance from the target increased, so Abidi developed an enhanced version that performed shape restoration using a conjugate gradient technique. The ability to extract the effective focal length is important. It reduces the error from assuming a fixed focal length, and can determine the effective focal length when using an autofocus camera.

2.4 Sparta and Rome

The Sparta system is the fruition of years of work by the Michigan State University Departments of Electrical Engineering and Computer Science [151]. Sparta is based on the occupancy grid system pioneered by Elfes, but incorporates a rotating sensor head holding eight ultrasonic sensors. Figure 2.6, taken from [151], illustrate the construction. The Transitions Research Corporation (TRC) Labmate base is capable of speeds up to around 1 meter per second, and supports ultrasonic and infrared sensors with an additional sensor board. The sensor board mates to a PC through a serial port, and the sensors and locomotion are commanded by high level function calls based on the work of Crowley [36, 37, 38]. Several computational enhancements have been included to decrease the time necessary to scan and map its immediate environment. Its path planning system is based on the work of Lumelsky [104, 105], and it has demonstrated intraroom mapping and navigation capabilities. It presently

ta

to

at

do

av

air

to

tion

of

ill

lap

so

acc

wa

and

with

land

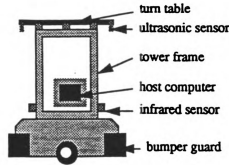


Figure 2.6. Sparta

takes about 25 seconds to scan and update its occupancy map. Work is maturing towards implementing a wall-following algorithm that will enable Sparta to navigate at higher speeds down a known hallway [14, 15, 86]. IR sensors will be used to skip down the side of a hallway at a standoff of 1.5 meters. Ultrasound will be used to avoid a frontal collision. There are no plans to incorporate vision. Present efforts are aimed at achieving the best possible speed of travel and using only prudent sensing to accomplish the task. No pose estimation is performed outside dead-reckoning.

The Sparta system represents an inexpensive platform for conducting both navigation and sensory research. Its design became the initial blueprint for the construction of Rome, the hardware platform used in the examination of this dissertation. Rome, illustrated in Figure 2.7, was designed to have an untethered, on board Zenith 386SX laptop computer, and a static array of 24 Polaroid Instrument Grade ultrasonic sensors around the perimeter of its sensor head. Decreasing the time necessary for range acquisition, and the capability for monocular frame grabbing were the primary hardware motivations. A Panasonic GP-KR202 color CCD camera mated to a 6mm lens and a Data Translation DT2803 black and white frame grabber were chosen. Our goal with Rome was to provide a mobile sensor platform that would enable us to pursue landmark based navigation, and pose recovery, while simultaneously increasing the

speed of travel and decisionmaking. As Rome matured as a hardware platform, its role as an sensory testbed attracted other experimenters and approaches.

2.5 Summary

We have examined a broad variety of robotic navigators in this chapter. Each system has its merits and liabilities, and each aids in framing the challenges we face in our approach to landmark based navigation and pose recovery. Many efforts provide insight into solving specific navigational tasks, and extend traditional methodologies toward more general, and robust problem solving capabilities. A few central trends have become apparent in robotic sensing. Computer vision is unique in offering surface detail, and spatial resolution capabilities, but at a computational cost that restricts real time applications. Ultrasonic sensors enable direct, real time ranging, but lack the necessary angular resolution to independently solve high level problems without large *a priori* databases. Multisensory approaches are becoming very common in robot navigation, but methodologies to fuze dissimilar sensory information are just now maturing. Pose estimation is the base task with respect to general robot navigation. Progress in enabling the navigator to self-locate will rapidly open up its range of applications. Most progress to date, observed with both indoor and outdoor systems, has been in domains providing at least minimal structure to the navigator.

In the next chapter we will examine and define the task of independent, autonomous localization. This discussion will expose some of the representational challenges whose solution both limits its domain and enables its utility.

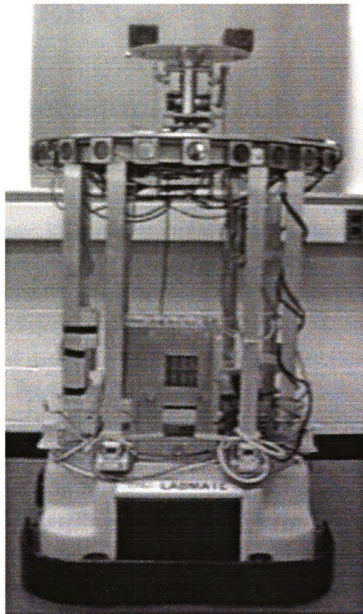


Figure 2.7. Rome

CHAPTER 3

The Where-am-I Problem

A core question in robot navigation is the robot’s ability to locate its pose relative to a frame of reference. The immediate scope of the present work is indoor navigation making the reference frame relative to a particular building coordinate frame. This chapter presents the Where-am-I Problem; a key challenge for robot navigation applications. A robust solution to this problem provides the recovery of a lost pose, and enables aggressive approaches to navigational tasks. We discuss the salient elements of recovery from disorientation, focusing on connectivity and relational information.

We then present competing approaches to representation and propose a representation for the robot’s world. Instead of establishing a dense, computationally demanding geometric model of the robot’s environment, we offer a sparse representational scheme that requires strong symbolic matching, ambiguity resolution and error rejection capabilities. Our representation is built upon the novel idea of encoding extractable environmental features as attributed edges in a sparse symbolic graph.

3.1 Navigating with a Map

A pivotal question in navigation is the robot’s ability to locate its pose relative to a reference frame. The degree of accuracy that defines an accurate pose is relative to

the application, and is correlated to the objects being manipulated. For industrial manufacture, this accuracy may be less than $\frac{1}{8}$ of an inch, but for navigation we need less metric accuracy. The degree of pose accuracy might vary from task to task within the same application. Six inch pose accuracy may be sufficient for hallway travel, but is too coarse for entering a room through a threshold. If the robot has some knowledge of a past landmark, it could use dead-reckoning to approximate its current pose. Our Labmate is wheeled and has shaft encoders. In a perfect world with no slip or slide between wheel and ground, readings from these encoders would give an extremely accurate estimate of pose [147]. Unfortunately, wheels slip and slide with magnitudes that are a function of wheel velocity and acceleration, particular ground surface composition and shape, and wheel load. All these aspects can be modeled, but the surface the robot travels upon must then be restricted and we can rarely control the surface dirt and temperature well enough to avoid errors of the same magnitude [21]. We don't see much hope or portability in trying to overcome the uncontrollable factors in a particular environment in order to construct a metric map to navigate with. So what must be done is to construct a navigator that can be periodically updated with an accurate pose taken from some passing landmark. This is similar to the way humans travel through both familiar and unknown surroundings. Humans rarely possess an accurate metric map of an environment unless it is extremely familiar. Kuipers' work supports a landmark-based cognitive model in humans [92, 93]. His thesis is that during human navigation, decision points and landmarks encountered constitute the cognitive map. Humans often construct a map based on topological and geometric properties, not on metric information. When a hiker gets lost in the Rocky Mountains without a map, he doesn't construct a metric map to navigate back to civilization, but most likely searches for recognizable landmarks that provide a coarse pose. Even with a pose provided only by the sun's position, or the Big Dipper, a hiker can intelligently navigate towards an anticipated river, road, or powerline. The

hiker then can follow this feature using recalled topologies to a safe haven. It bears emphasizing that the hiker is only completely lost if he has no metric, topological or geometric information. Soon a handheld Global Position Satellite receiver will provide our lost hiker his 3D pose to within 25 meters [130], but this is too coarse for our application, and offers no advantage indoors.

Continuing the hiker analogy, the Where-am-I Problem occurs when the hiker is placed in location with no dead-reckoning information. The hiker is free to look around and extract feature data, and recall geometric and topological relations, but the hiker is surrounded by what Brooks calls an uncertainty cylinder [21]. This is a volume centered about an estimated position with radii corresponding to the positional uncertainties along various axes. Brooks' approach is to build and maintain a rubbery, relational map. His approach closely models the human manner of navigation. A liability is that the farther the robot navigates, the larger the uncertainty cylinder. After three right turns, it is even possible to be uncertain that it has traveled at all. Without periodic pose fixes, a robot in any building would be paralyzed with uncertainty. Elfes' approach is to build an explicit 2D metric map of the robot world. Using a probabilistic construct, this method yields a very useful map *if the robot knows where it is in the map*. Sparta has already shown some weakness in dead-reckoning despite a world limited to a single room. Litton Industry's Integrator and Frohn's VISOCAR use optical floor markings to explicitly update the robot's pose as often as once a second, depending on its speed. This also represents an unacceptable handicap, too much structure for any expected utility in our application.

Ultrasound can't solve the Where-am-I Problem alone, it is too coarse in angular resolution and is corrupted by reflections and specularities. Crowley even compared sensing with ultrasound to trying to navigate through a dark house of mirrors with only a flashlight [36]. A camera can accurately extract the landmarks needed to periodically fix the robot pose. In this dissertation, we combine the metric and topological

approaches using a hybrid map. This map is an attributed graph holding the topological properties of the building in its connectivity, and coarse metric information in its attributed edges. McDermott and Davis pursued a massive artificial reasoning project using a similar approach [109]. Each feature that can be extracted by ultrasonic, monocular vision, or by IR sensors could be an edge. A discrimination tree will classify features by their properties to facilitate rapid correspondence. Fuzzy feature data, most often from ultrasonic sensors, will be passed up to an executive routine that will try to match to edge attributes within a limited uncertainty region. When a high confidence feature match occurs, most likely from the camera, the accumulated dead-reckoning error determining the radius of the uncertainty region can be briefly reset. This periodic flushing of accumulated uncertainty is pivotal to having any useful capabilities. The central interest in the Where-am-I Problem is recovering from complete disorientation.

Complete disorientation, or as complete as this generation of robots will deal with, occurs when the robot is completely uncertain of its position *within its map*. This is probably the most important problem to conquer in order to build a more flexible local capability. More challenging tasks can be tackled if our robot can gracefully recover from complete navigational failure. Test pilots will try just about anything if they have a parachute on their back. So in our application, the challenge occurs as we startup the robot at some unspecified location within a chosen domain: the robot has complete freedom to use its *a priori* map, but it could initially be anywhere on this map. Unless we know the angle the acoustic axis forms with the reflecting surface, ultrasound is not capable of a high confidence match with an edge attribute, so the robot must register a match by other means. Humans might try to locate themselves by reading room numbers, but more often would seek a topological or geometric feature such as a stairwell or hallway corner. So unless we spread enough artificial visual landmarks to always have one within the camera's field of view, the

robot will wander around looking for a visual landmark. If the robot were within a room at startup, it would first have to locate and navigate to the doorway and then enter the hallway. Since a door is an edge on the graphical map, and the robot is looking for a particular edge to fix its pose, it must traverse the nodes to seek another edge unless it registers with its initial edge. A logical follow-on effort might be to recognize a room by its dimensions with the method of Sarachik [133]. The robot must be able to fix its pose within a reasonable amount of time. A performance time goal of 1 minute seems reasonable, but this presumes that the Labmate base is capable of a reasonable forward speed. Dulimarta’s exploratory work with the Labmate base reveals that linear travel in excess of one meter per second will be difficult to achieve with the current hardware [47].

3.2 Representation

The choice of representations for a building map pivot around issues including the ability of holding information usable to the robotic navigator, flexibility in a dynamic environment, and appropriateness as a path planning database. The key decision strategy is to make the choice from the robot’s point of view, respecting its limited sensing capability and coarse navigation skills. Adopting this perspective is what cartographers would term a robot-level of use [27]. Cartographers are often challenged to build different representations of the same region for different levels of intended use. Another challenge is to separate the decision from our human skills, but not from our cognitive experience. A human spy might desire to have a complete scale map of a building, supplemented by color photographs of particular features. But this wealth of data is only useful to humans because of their massive data compression ability, and fast indexing skills. Humans need little metric information as long as they have a sense of scale. Human experience records how wide doorways are, and what elevators

look like. There are three major types of representational schemes that have been employed by various researchers. They are dissimilar in approach, each with their attributes and liabilities, and are presented in the next three sections.

3.2.1 Geometric Paradigm

Traditional approaches to robot sensor interpretation have mainly relied on the recovery and manipulation of geometric world models. Low-level sensing processes extract geometric features such as surface patches and line segments from sensor data to constrain the sensor interpretation process. The resulting deterministic geometric world models are then used as the basis for robotic activities such as obstacle avoidance, path planning, or planning assembly and grasping tasks. These approaches characterize what Elfes has labeled the geometric paradigm in robotics and computer vision, and have several shortcomings [51]. Elfes concluded that these approaches lead to brittle and sparse world models. They require early decisions in the interpretation of sensor data for the application of specific model primitives, and do not provide adequate means for handling the uncertainty and error intrinsic in the sensory data. These approaches rely heavily on the adequacy and accuracy of the prior world models and the heuristic assumptions made. This also introduces strong domain-specific dependencies. Better environment descriptions are primarily derived from the application of finer tuned prior models and further constraints to the available sensor data, not from further active sensing. A consequence of these liabilities is the gap between the two informational layers: the layer that corresponds to the imprecise and limited data actually delivered by the sensor data and the layer of abstract geometric and symbolic world models manipulated by the perception processes. If such a gap exists, the mobile robot's utility will be restricted to highly structured domains.

3.2.2 Computational Geometry

Reducing the 2D world to a collection of polygonal lines and regions has been heavily studied by the computational geometry community. Its modeling restrictions of straight lines and convex polygons aren't a severe restriction to the robot world. The main attribute of this methodology is the existence of a tractable shortest path algorithm. Sharir's algorithm [137] has a complexity of $O(n^2 \log n)$ which is not a handicap here, but its assumption that the world is completely known and static is a serious liability. Once a shortest path is computed, the robot begins travel, but if a previously unknown obstacle blocks the path, the robot must recompute its path, again at an $O(n^2 \log n)$ cost. This approach also places the path through the vertex corners of the confining polygonal regions, an area along the path very likely to contain obstacles. Pursuing a retraction approach to constructing a Voronoi diagram would move the path out through the centers of the tessellated regions, reducing the probability of stumbling into corners due to poor gross navigation. This approach also suffers from the assumptions of a known and static environment. Incorporating environmental updates is also cumbersome with this method because a new vertex may actually change the coordinates of its neighbors [135]. Floyd's algorithm could be used instead of Dijkstra's since it adds robustness to the path planner by finding a shortest path from any point to any other point. At an additional $O(n)$ cost, this would allow the navigator to immediately know the shortest way remaining to the goal after dodging an unexpected obstacle. Representing features that are extractable by the robot sensors within this methodology is unclear. To a robot, and indeed a human as well, a visual landmark such as a colorful symbol or room number could be more useful than geometric region boundaries. In our opinion, the computational geometry approach to mapping is inflexible, and too dependent on a static environment to offer an advantage in our application.

3.2.3 Occupancy Grid

Elfes pioneered the occupancy grid representation. This grid is a 2D array of square cells representing a square of floor space, each cell being assigned a probability of occupancy that can vary with time. This occupancy data is derived directly from the ultrasonic returns, and the robot literally builds its world map as it travels, with no *a priori* information. This approach can be tailored to the coarseness of the robot's sensing capabilities, and nicely supports a dynamic world. The robot navigator can fall victim to false or reflected returns, and can even become trapped by phantom obstacles, particularly in close quarters. The significant liability of this method is one of scale. The robot needs a grid cell size of about a foot square to support local navigation, but representing a large building this way is troubling. Multiple scales are needed in the representation. Moreover, path planning to goals outside the robot's ultrasonic horizon will be cumbersome with this representation, since there exists so many similar, but distinct cell trails to the goal. Local use of this representation has shown the computational overhead to be significant, and has motivated the pursuit of faster, IR-based strategies for hallway travel [151]. In our opinion, the occupancy grid representation is too fine in detail for use outside a locality such as a single room, and would seriously degrade the robot's navigation speed. The grid cell research community is very active and new approaches using this methodology appear frequently.

3.3 A New Approach - StickRep

The structure of a large building is often dominated by long hallways. This structure lends itself well to defining freespace as a $1\frac{1}{2}$ D map. The robot is most interested in freespace, not obstacles. Travel on a single floor could be represented by abstractly cutting and stretching out a hallway into a line with rooms hanging off the

hallway line. Branch hallways are nodes on the line. Features along the hallway are captured as edges between the physical nodes, with attributes describing information of interest. The cartographic research community has built databases on very similar concepts [27]. Highway segments are encoded as links and towns as nodes, each with attributes, and networks are built upward from these primitives. Political units, urban areas, and water bodies are modeled with a set of links forming a closed polygon. These basic components may have connecting relationships defining the overall topology. The building representation which we developed could be termed a feature highway. Wall and door features are represented as a sequence of attributed edges with nodes providing the topological connectivity. The robot navigator travels along a feature highway either confirming anticipated feature edges, or updating new information in the graph. We are motivated by economy; to investigate how well a navigator can perform with minimal information.

The building map forms a database: the executive routine registers matches to the inexact feature data acquired by the sensors. The type of features that are included in the database are limited by the ability of the sensors to reliably extract these features. Many richer representational schemes have been developed, but most encode features with more complexity than we can presently extract [3, 5, 6, 7, 9, 14, 15, 30, 35, 51, 57, 58, 60, 81, 84, 119, 160, 161]. Our modest experiment warrants a sparser representational scheme. Since the angular resolution of the ultrasonic sensors are poor, only coarse angular adjacency information is encoded into the database. The StickRep representation scheme sparsely represents the 3D world, capturing only topological connectivity, coarse metric data, and feature types. Such a scheme places a greater burden on subgraph matching, ambiguity resolution, and error recovery than do other more mature efforts [9, 12, 54, 87, 124].

We examined several candidate structures, and chose an attributed edge graph to represent the *a priori* building data base. Planar physical domain features such

as wall segments and doors are represented as attributed edges. Door jams and hall corners are the junction of planar physical domain features, and are represented as attributed nodes. StickRep is a departure from the majority of navigation representation schemes. Many previous efforts have been built on 2D representational structures [9, 15, 49, 51, 66, 141, 158]. StickRep is a $1\frac{1}{2}$ dimensional structure that provides a minimal representation for our robotic navigator. StickRep encodes sequences of feature information. Each edge contains some or all of the following attributes:

$$\left\{ \begin{array}{l} type \in \{corner, doorway, elevator, hallway, pattern, stairwell\} \\ metric \in \mathbb{R}^+ \\ prototype \in \{doorjam, wallsegment, pattern\} \\ pattern \in \{\clubsuit, \diamond, \spadesuit, \heartsuit\} \end{array} \right.$$

Nodes hold adjacency information of the incident feature edges and contain the following attributes:

$$\left\{ \begin{array}{l} adjacency \ angle \in \{0, \dots, 360\} \\ global \ coordinates \in \{X^W, Y^W, Z^W, \alpha, \beta, \gamma\} \end{array} \right.$$

We illustrate this representation with an example in Figure 3.1. To more clearly portray our representation, we have omitted the prototype edge attribute, and actually have drawn the edge adjacency angles to represent the angles present in the physical domain. In all the domains we examined, sequences of features were represented as rings of adjacent edges. Since features on opposing sides of a hallway are not physically adjacent, opposing features may be represented in separate rings, depending on the hallway geometry of the domain. In Figure 3.1, the three graph components illustrated are actually fragments from three separate feature rings.

Each ring is joined to another ring at only one artificial node, so a complex

domain is represented as an attributed graph with weakly connected components. This representation proved fast and flexible when matching physical feature adjacencies, but representing opposing features is more difficult, and was not pursued. This representation was used to match sensed feature data to *a priori* domain information. After a navigator determines its current pose, it must maintain this pose by periodic updates. If a high confidence match is made between sensed data and the next anticipated edge on the graph, within its current uncertainty circle, the current pose estimate can be updated to this edge. If not, the robot will continue searching for the anticipated feature edge until it reaches a tolerance limit that initiates complete pose recovery. Chapter 6 examines the use of this representational scheme to support extensive simulated navigation.

3.4 Summary

In this chapter we have presented the Where-am-I Problem. A solution to this problem provides the ability to recover pose, and enables aggressive approaches to navigational tasks because the executive routine is confident of recovery from disorientation. We discussed the salient elements of disorientation recovery, focusing on feature connectivity and relational information. We emphasize symbolic feature data, not metric properties or relationships.

We then discussed competing approaches to representation as well as our proposal for representing the robot's domain. Instead of establishing a dense, computationally demanding geometric model of the robot's environment, we offer a minimal representational scheme that requires symbolic matching, ambiguity resolution and error rejection capabilities. Our representation is built upon encoding extractable environmental features as attributed edges in a symbolic graph that sparsely models the 3D world. Our representational scheme hinges on strong, inherent assumptions about the

structure of an indoor navigation domain. In the next chapter, we will present our methodology for extracting symbolic feature information from our structured domain.

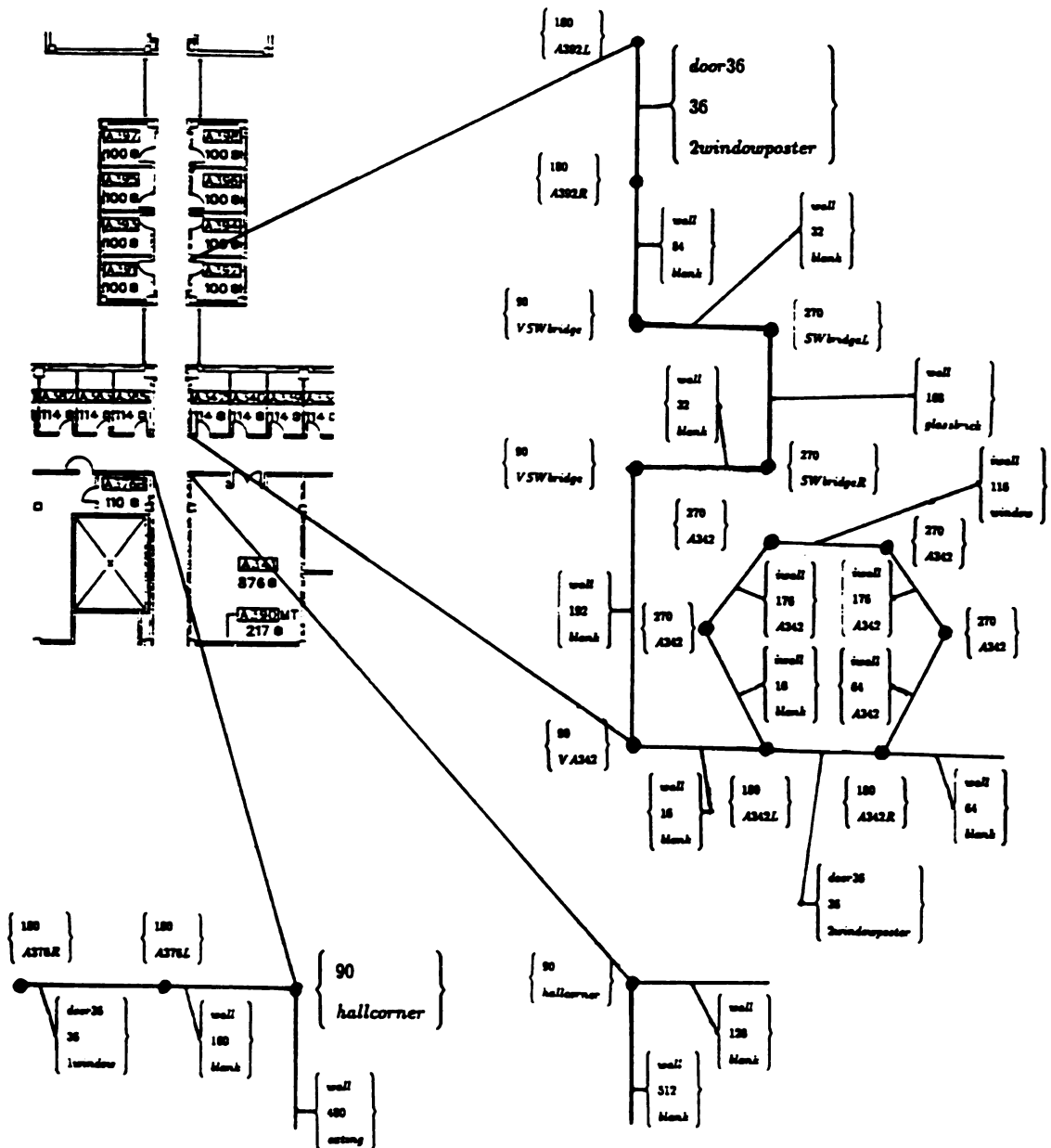


Figure 3.1. StickRep example

CHAPTER 4

Sensing Landmarks

In this chapter we will examine the capabilities and liabilities of monocular vision and ultrasonic ranging as compared to the more traditional stereo vision. We will note that stereo vision cannot provide the ranging we desire rapidly enough for our real time application. The use of monocular vision and ultrasound are not without their own liabilities, and we will explore ultrasonic ranging error experimentally. The key problem is the ability to accurately replace the depth information provided by stereo ranging with a fused monocular vision and ultrasonic methodology.

The difficulties with stereo vision systems led us to investigate other promising ranging systems [9, 15, 51, 87, 107]. We sought to combine the most capable attributes of vision and active ranging approaches into a fused acquisition methodology using a single camera and a suite of ultrasonic sensors. In this chapter we will present our landmark sensing strategies and closely examine the capabilities of ultrasonic sensing. In the following chapter we will formally present the geometric algorithms that use these sensing strategies in a structured domain.

4.1 Visual Sensing of Vertical Ribbons

Fine feature detail in the operating domain cannot be obtained through ultrasonic range sensing. Ultrasonic range data is too coarse in an uncontrolled environment and erroneous data is common. Ultrasonic sensing will be examined in detail in Sections 4.2 and 4.3. We will now discuss the extraction of fine feature landmarks from the domain using a monocular camera.

The requirement for real time processing restricts many common approaches. Many researchers have pursued stereo vision methodologies, but they struggle to overcome the computational complexity of stereo correspondance [33, 34, 87, 99, 116, 118, 121, 126, 129, 146, 149]. In the application of stereo ranging methodologies to real world navigation, the intrinsic computational expense of extracting three dimensional range data from stereo pairs limits the number of features that can be tracked. Real time constraints have traditionally prevented the obtainment of dense 3D world models [22, 26, 42, 65, 139, 142, 162]. Constant increases in computing power are now beginning to yield promising results [107, 120]. Practical stereo vision navigation systems have constructed sparse depth maps by matching high contrast points selected by an interest operator or geometric features such as edges [108, 114, 145]. Additional problems for robotic applications come from operating under available illumination and in domains with unpredictable surface acoustic characteristics.

The depth information recovered by a stereo system is vital in robotic applications, but depth data can also be provided by active ranging. We chose to pursue the extraction of symbolic features in our domain with single images fused with ultrasonic range data. Despite the avoidance of stereo correspondence, many powerful algorithms, such as Canny edge detection, also challenge real time implementation [28, 78]. Our entire image processing suite was developed under this heavy real time requirement. Many of the visual features in an indoor domain possess strong

verticality. Doors, windows, and supports present striking vertical landmarks yielding strong, long ribbons in images. We designed our approach to extract these vertical ribbon edges and classify the observed features using a template scaled to the feature range.

4.1.1 Feature Extraction

Before we examine visual edge sensing, we offer the following definition:

Vertical Edge Ribbon *A physical domain feature that may produce two parallel vertical intensity edges in an image. Both edges are capable of being simultaneously seen within the field of view of the sensing camera. The lines containing the vertical edge ribbon are normal to the floor of the domain.*

A door is a common vertical edge ribbon found in indoor domains. Doors are logical landmarks for human navigation and are consistent with the cognitive map approach [92, 93]. A door often provides the strongest intensity edge in a domain [77, 136]. Here doors are symbolic, they represent a common static feature, extractable under varied illumination throughout the structured domain. Posts are less common in some domains, but could also be easily extracted. We experimented with several standard edge detectors on training images examining the filtered images for sensitivity to filter size, noise, and camera misalignment [95, 128]. We chose a vertically oriented 5x5 Prewitt filter as the candidate that best extracted strong, long intensity edges from the training data [125]. Kriegman pursued a larger mask to smear images in a broader application [87]. Our filter not only rejected horizontal features, but it also averaged vertical features. This averaging has the following desired effects:

- Horizontal edge ribbons vanish.
- Slanted edge ribbons are blurred.
- The effect of small edge features and marks is attenuated.
- The effect of image noise on vertical edges is attenuated.

Ribbon length is a strong indicator of a large vertical feature. If we require vertical persistence, we filter out finer vertical features and marks that clutter our domain. To rapidly examine this attribute, we performed vertical integral projection. Let $I(r, c)$ be our filtered image. The vertical integral projection of $I(r, c)$ can be defined as:

$$V(c) = \sum_{r=1}^{nrows} I(r, c)$$

This column summing technique is similar to Kanade's two dimensional work on the recognition of human faces [80]. The results of the vertical integral projection required careful thresholding to properly segment the strongest vertical ribbons present in the scene. Additionally we needed to recover the footprint of the vertical projections onto the floor of our modeled two dimensional domain. With this footprint we could register fused ultrasonic range data to classify the landmark using the algorithms to be developed in Sections 5.2 and 5.3. After thresholding to a binary image, pixel column bins were clustered based on three nonempty bins within a five bin window. Pincushion lens distortion caused edges one pixel wide to bow across seven pixels at a viewing range of six feet. To accomodate this, footprints were determined by calculating the center of mass of the bins. Using the sensing geometries to be developed in the next chapter, we found that the subpixel accuracy obtained from this bin clustering technique reduced measurement variance 39 percent in a controlled experiment.

Several adaptive thresholding techniques were pursued [31, 73, 79, 132]. Chow's technique of fitting a bimodal Gaussian model to the image histogram had been implemented locally but it could not be adequately tuned to the crisp intensity gradients common to man-made environments and the varied illumination present in the training images [74]. We reluctantly turned to absolute thresholding based on the library of 157 training images taken over a four month time period in varied illumination. We found that the door jams that constitute the vertical edge ribbons typically subtend 2.5% of the image scene as viewed from Rome. We set the absolute threshold level at 5% to accept an equal degree of desired edge and accidental marks.

After we had completed the initial design that successfully identified features in our filtered image, we sought to minimally sample rows from the image to both reduce the computational load on our system, and to identify long edge ribbons. Returning to our training suite of images, we empirically determined that five rows extracted from the image were sufficient to capture vertical edge information from a door within our field of view. Three of these sampled rows were near the vertical image center, with the others toward the top and bottom. If the range to the door is six feet or less, sampling these particular rows will always be sufficient to capture the full vertical range of the feature. But our navigator may certainly view doors obliquely and at greater distances. To accomodate this behavior, a range-based scale factor controls which rows are sampled from the image as a function of the ultrasonic range data obtained coaxial with the optical axis.

The image processing algorithm is summarized in Figure 4.1. It implemented dynamic sampling of five rows as a function of range, Prewitt filtering, absolute thresholding, vertical integral projection, and clustering of the vertical ribbon to a footprint. A complete example of our processing algorithm is shown in Figure 4.2. In Figure 4.2(b)-(c) the entire image is shown for clarity, but processing was only performed on the five rows sampled. In Figure 4.2(d) the vertical projection has been

smeared for ease of illustration. The dark pixels represent weights of confidence in the feature footprint location, darker pixels representing areas of greater confidence. This processing suite demonstrated excellent tolerance to the pincushioning produced by the 6mm lens we used, and the camera misalignment often observed during Rome's travel. Clumping the vertical projection was the most significant component in the attenuation of these undesirable effects. This clumping successfully recovered long vertical ribbons with a misalignment angle as great as 10 degrees from vertical at a distance of 12 feet. This result is strongly correlated to the maximum spread of the center three sampled rows. It should be noted that roll axis camera misalignment causes all vertical and horizontal feature edges to be smeared by the averaging filter, and, provided that the three center sampled rows are relatively close in vertical proximity, their projected footprints will be successfully clumped.

- (0) Dynamic sampling of five image rows based on range to ribbon
- (1) Vertical 5x5 Prewitt filtering
- (2) Thresholding to pass top 5%
- (3) Vertical image projection
- (4) Clumped vertical ribbon footprint to subpixel accuracy
- (5) Output landmark candidates each with width and location

Figure 4.1. Vertical Edge Ribbon Extraction Algorithm

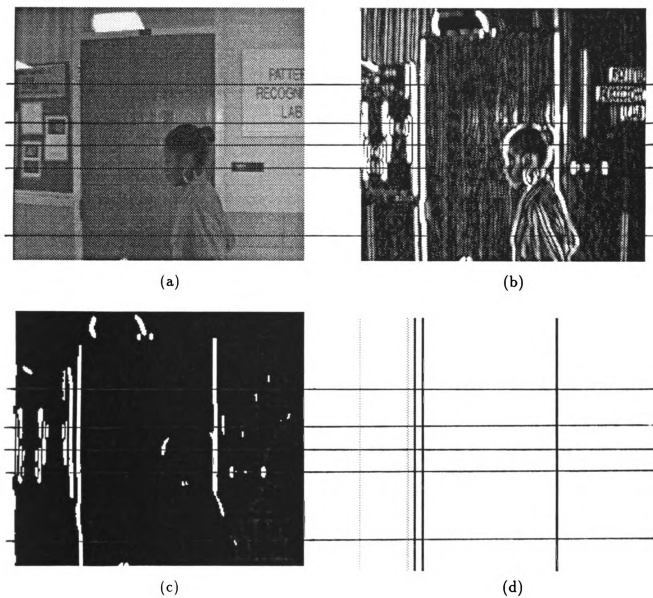


Figure 4.2. Image processing suite example. (a) Original image (b) Vertical edge image (c) Thresholded edge image (d) Smeared vertical projection with darkened feature footprint

4.1.2 Landmark Classification

The output of the first processing stage, described previously in Section 4.1.1, provided candidate footprint pairs, with confidence weights, to the landmark classification stage. Despite the thresholding and filtering, several candidate pairs are often passed forward for potential classification. In this stage, fused ultrasonic range data is used to determine the estimated size of landmark candidate pairs. If a large depth edge is noted across the ultrasonic horizon in an area corresponding to the candidate feature, then a line fitting routine interpolates an estimated depth to the feature by using only the flank depths. This occurred frequently in both the training and test data sets because several doors were partially or completely open. Outward opening doors left partially open were often ultrasonically sensed as a sharp convex depth edge. Inward opening doors similarly were sensed as a sharp concave depth edge. The structure of the hallway domain made the line fitting quite accurate in recovering the depth to the closed door position. Open doors were more difficult to accurately classify than closed doors, but this was mainly due to image clutter inside the open door misleading the first image processing stage, and not due to ultrasonic ranging shortcomings. We will discuss these results in Section 6.4.2.

Since door landmarks are only present in discrete metric sizes, their *a priori* probabilities can be determined (see Tables in Chapter 6). A Bayes decision rule for classifying candidate pairs were desired, so experiments were conducted to estimate the class conditional probabilities [62]. Using the algorithms to be presented in Sections 5.2 and 5.3, data was collected and conditional probabilities were estimated at an abscissa bin granularity of 0.05 inch. This estimation was necessary since observed data came from a continuous metric spectrum. Posterior probabilities could then be determined. Table 4.1 illustrates, in inches, the landmark classification boundaries implemented using a Bayes decision rule.

Landmark	Low Boundary	High Boundary
32	29.1	33.5
36	33.5	38.8
40	38.8	42.7
30/30	56.4	64.0
36/36	67.9	76.3

Table 4.1. Landmark classification boundaries showing ribbon width in inches

The landmark candidate pair with the highest confidence weight assigned by the initial processing stage will be classified using Table 4.1. If a candidate pair falls outside these boundaries it is rejected, and the subsequent candidate pairs, above a preset confidence threshold, are considered in turn. No tuning or training on the test image suite was done. The test suite classification performance of the two processing stages presented in the last two sections was encouraging. After we develop the necessary sensing geometry in the next chapter, we will more thoroughly examine these classification results in Section 6.4.2. In the next section we turn our discussion to a more thorough examination of ultrasonic sensing.

4.2 Theory of Ultrasonic Sensing

One important application of an ultrasonic sensor system is active ranging. Ultrasonic sensors use high frequency sound to determine the time of flight between the sensor and an object. The acoustic beam propagates from an emitter through its medium of travel as a mechanical wave and is reflected back to a receiver. The wavelength λ of the ultrasonic wave is:

$$\lambda = \frac{c}{f}$$

Here c is the speed of sound in the medium and f is the frequency of the emitter. The speed of sound is not constant, even within a given medium, as temperature and humidity are factors to consider. Typical frequencies for robotic applications range from 20 KHz to 2 GHz. Our Polaroid Instrument Grade transducers operate in the 60KHz range, with $\lambda = 5.7\text{mm}$. A voltage pulse excites the transducer which causes an acoustic pulse to be emitted. The echo reflected from an object may be detected by the same sensor that emitted it. A typical time of flight system produces a range value when the echo amplitude first exceeds a set threshold value. A range measurement, d_0 is determined from the round-trip wave travel time of flight t_f by:

$$d_0 = \frac{ct_f}{2}$$

Absorption of the ultrasonic wave by the propagation medium influences the attenuation of the wave. The degree of this absorption depends on the medium and the radiating frequency. Gaseous media attenuate ultrasonic waves significantly more than solids or liquids, and this attenuation depends greatly on the particular frequency used [51]. Currently most robotic applications employ ultrasonic systems radiating into the atmosphere. The pressure amplitude of the acoustic wave decays exponentially with distance. If p_0 is the pressure amplitude of the planar acoustic wave, and α is the absorption coefficient, then p is the pressure amplitude of the wave at a distance d :

$$p = p_0 e^{-\alpha d}$$

The absorption coefficient α clearly determines the rate of attenuation [72]. The extinction distance d_e is defined to express this attenuation of the ultrasonic wave in the atmosphere, $d_e = 1/\alpha$. At ambient temperature and humidity, with $f = 60\text{kHz}$, $d_e = 17$ meters [51].

Employing ultrasonic waves in the atmosphere is significantly less efficient than radiating through a solid or liquid medium. This is not just due to the

attenuation factor, but also because the impedance mismatch between the solid ultrasonic transducer and the air is much higher [82]. This translates to a high transmission loss compared to the energy transfer across a lower impedance boundary such as a liquid or a solid. But this same impedance mismatch on transmission becomes an advantage upon wave reflection. This phenomenon causes strong reflections, and not energy draining absorptions, by an object in air. The reflective behavior of a surface in air depends on the object texture [18]. If the period of the texture is much greater than the ultrasonic beam period, surface reflection is essentially specular. For robotic applications, most object surfaces can be considered quite smooth, and the ultrasonic beam reflection largely specular, with a small diffuse component.

Sophisticated signal processing of the ultrasonic wave allows a parametric evaluation of the returning arc formed when the sensor is scanned across an object [18]. This approach permits the differentiation of planar surfaces, corners, and edges in a specular domain. In nonatmospheric medium, higher ultrasonic frequencies may be employed that permit ultrasonic imaging that achieves resolution comparable with what can be achieved with optics in air [72]. Short of these application types, ultrasonic sensing systems provide a simple, low cost alternative for range data acquisition. Ultrasound operates independently of the environment lighting conditions and the optical characteristics of the object surfaces. Current commercial systems for atmospheric transmission still offer poor angular resolution and suffer from problems associated from multiple reflections. In the next section we will experimentally examine some of these physical issues in an attempt to develop an operating envelope supporting accurate and robust navigation.

4.3 Ultrasound Studies

Ultrasonic sensors are widely used in mobile robot applications. A strong systemic attribute is their simplicity, low cost, and the fact that distance measurements can be determined directly. A typical sensor has a useful measuring range of 1 to 35 feet, with a beam width of 15 degrees; so an array, or belt of sensors is often employed [9, 15, 14, 24, 29, 38, 37, 45, 50, 51, 52, 53, 54, 63, 91, 111, 112, 131, 134, 140]. The more mature of these efforts use probabilistic models, redundant sampling, or sampling histories to handle the range errors that are common to ultrasonic sensors. Physically based ultrasonic sensor models are just now maturing [101]. One effort elegantly and extensively characterizes a single sensor in a clean laboratory environment [18]. It notes, but doesn't explore the signal error produced by sound absorbing surfaces, specular surfaces, and surfaces sensed from oblique angles. It became apparent that surrounding Rome there existed an area of reliable sensing, or a sensing footprint, that varied with the type and orientation of the surrounding surfaces, and the sensor type. This conclusion motivated an extensive examination of ultrasonic sensor performance against various surface types at varied orientations. We will now examine this phenomenon and develop an ultrasonic sensing footprint to support accurate and robust navigation.

During preliminary navigation experiments, it was observed that Rome's ultrasonic sensing system ranged surfaces very accurately when the acoustic axis and the surface normal were approximately colinear. It was also observed that as the angle between the surface normal and the acoustic axis increased, so did the ranging error. Individuals wearing thick clothing often were not sensed at all, while at other times the same individuals were accurately ranged. An experiment was designed to examine the effect of surface characteristics on ultrasonic ranging. Three surfaces types were presented to the ultrasonic sensor at various orientations to the acoustic axis. The

hallway was initially chosen as the testing laboratory, not because we thought it was a clean environment, but to include multiply reflected signals causing ranging error. Multiple reflection error was not observed in this experiment, and the results strongly followed a pilot study conducted in a acoustically isolated environment.

Concrete was chosen as a common domain feature surface, while polished wood presented a much smoother ranging surface. A padded fabric wall partition was examined to characterize suspected energy absorption. One thousand ultrasonic samples were taken from three surface types at four orientations each for a total of 12,000 samples. Let ϕ be the angle between the acoustic axis and the surface normal, with $\phi = 0$ corresponding to the sensor directly facing the surface. Table 4.2 summarizes the percent range error observed. We repeated the entire experiment to increase the confidence in our conclusions, obtaining the results summarized in Table 4.3.

For a given surface and orientation, only two to nine discrete range values would occur. The histograms of the complete experiment are in Appendix A, and illustrate the deterministic nature of the ranging error. A total of over 40 thousand samples were taken to confirm these results. The polished wood and concrete surfaces ranged quite accurately. Cloth surfaces also ranged accurately, but only if $\phi = 0$ (the surface normal and the acoustic axis align). The standard deviation of the ranging error is exceedingly small with useful error percentages.

These results are encouragingly similar to our initial experiment. We desired to model the ultrasonic error Rome would typically encounter while navigating throughout the operating domain. Cloth clearly is a challenging surface to accurately range. If the acoustic axis and surface normal are not nearly colinear, then grossly inaccurate range values can be expected. The other surfaces can be properly ranged if the ultrasonic sensing is only performed while within this accurate ranging footprint. Accurate range acquisition can be obtained from concrete and wood if Rome's sensing can be controlled to maintain $\phi \leq 15^\circ$. In the next chapter we will propose two competing

sensing geometries that support accurate ultrasonic sensing. Multimodal Gaussian error models were developed from the ultrasonic sensing analysis presented in this section. These error models will support extensive simulation experiments that we will subsequently present.

4.4 Summary

In this chapter we have presented a fast technique for the sensing of vertical edge ribbons using a single camera. We designed our approach to extract these vertical ribbon edges and classify the observed features using a template scaled to the feature range. Range data was obtained from ultrasonic sensing. Our final design implemented dynamic sampling of five rows as a function of range, Prewitt filtering, absolute thresholding, vertical integral projection, and clustering of the vertical ribbon to a footprint. The extraction and classification suite demonstrated excellent robustness to the pincushioning produced by the 6mm lens we used, and the roll axis camera misalignment often observed during Rome’s travel. We have also extensively examined ultrasonic sensor behavior, and defined an accurate ranging footprint to provide depth information to the monocular vision system. In the next chapter we will propose two fused sensing geometries that build upon the sensing experiments presented here. Unlike many competing methodologies, these sensing strategies support real time feature extraction and landmark classification.

ϕ	Cloth		Wood		Concrete	
0	0.50	0.03	0.03	0.03	0.71	0.03
15	85	80	0.95	0.03	2.2	0.03
30	340	93	16	0.07	5	0.04
45	++	++	17	0.08	14	0.09
>45	++	++	++	++	++	++
n=1000	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ

Table 4.2. Experiment 1: Ultrasonic percent range error as a function of surface type and angle

ϕ	Cloth		Wood		Concrete	
0	0.53	0.03	0.04	0.03	0.6	0.03
15	100	95	1.2	0.04	1.5	0.03
30	390	84	8.9	0.06	4.3	0.02
45	++	++	22	0.9	15	0.08
>45	++	++	++	++	++	++
n=1000	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ

Table 4.3. Experiment 2: Ultrasonic percent range error as a function of surface type and angle

CHAPTER 5

Sensing Geometries

In the preceeding chapter we discussed a visual sensing approach capable of extracting vertical landmarks in real time and examined ultrasonic sensing towards the definition of an accurate ranging footprint. What is now needed is a methodology that fuzes these two sensing capabilities, enabling depth information to be attributed to an extracted domain feature. Constructing a sensing geometry that casts vision and ultrasonic sensing in roles where they can operate accurately, while simultaneously supporting unrestricted real time navigation, is a challenging task. In this chapter we will present two competing sensing geometries that strive towards this goal. These sensing geometries provide a framework for navigation and sensing, ensuring that sensing only occurs where the information is not only accurate, but useful to the particular task *right now*. These sensing geometries strongly influence the overall control plan of the navigating robot. The shortest path to a goal might not be the best path if the feature sensing system can not accurately extract landmarks along this route. A sensing geometry that does not restrict navigation control is a goal, but we feel that sensing capability is more important than unrestricted navigation and point to point speed. Furthermore, we desired a more general landmark extraction solution.

Ultrasonic error models, derived from the experimental results of Section 4.3 will

be presented in Section 5.4. These error models will be used in simulating the performance of the competing sensing geometries. The performance of the best sensing geometry, applied to several operating domains, will be examined in detail in Chapter 6. We begin the present chapter with a generalization of our sensing situation.

5.1 Motivation

Our present sensing problem, extracting and classifying a vertical landmark from its domain, is a common problem that is widely pursued. The sensing geometries to be presented were designed to enable ultrasonic sensors to accurately provide depth to a feature and assign absolute metric scale to the extracted landmark in real time. The domain constraints required by the sensing geometries are:

- C_1) The landmark has parallel linear edges delineating the face to be measured.
- C_2) An intensity contrast exists between the landmark and its domain.
- C_3) The orientation of the optical axis relative to the ultrasonic array is known.
- C_4) The landmark to be measured presents a planar face to the image plane.
- C_5) The angle between the landmark surface normal and the optical axis is less than 45° .
- C_6) The domain subset within the field of view is approximately planar.
- C_7) The focal length and angular field of view of the camera lens is known.

These constraints represent the domain assumptions we were required to make to accurately assign depth to a landmark. Our goal was to provide as near an unrestricted navigation and sensing environment as could be engineered, but some constraints became necessary to reduce the complexity of the problem. It is important

to note that there is no requirement for ultrasonic or camera calibration. Only the mounting position of the camera relative to the ultrasonic sensing array and the camera's field of view are required. Sensing accuracy would improve with a decrease in the angle of Constraint C_5 , and is tightly coupled to the experimental results of Section 4.3. Continued advances in low cost ultrasonic sensors, would not remove Constraint C_5 , but would improve sensing accuracy. If the angle of Constraint C_5 is known, then Constraint C_6 can be removed. Constraint C_4 implies that the sensing geometries actually measure the projection of the landmark onto the image plane. Nonplanar landmark faces would have their projection onto the image plane measured. We will examine the limits of each geometry's accuracy envelope in detail in Section 5.5.

The sensing geometries to be presented are certainly not alone in their ability to measure objects in their constrained domain. Industrial applications have demonstrated more accuracy, but gain this advantage from a predictable, and much more structured domain. Less structured domain applications have yet to obtain real time performance [87]. The sensing geometries presented in the following two sections offer a robust, low cost, real time solution to problem instantiations occupying the middle ground between a well structured, specialized industrial application, and a large scale 3D explorer.

5.2 Two Side Geometry

Ultrasonic sensors can provide accurate ranging within a sensing footprint. The pivotal task of this sensing framework is to fuse the depth information obtained from the ultrasonic sensors to extracted feature edge ribbons. In a monocular image from an uncalibrated camera, the only ground truth references available are the edges of the field of view. Constraint C_3 in the preceeding section was established specifically

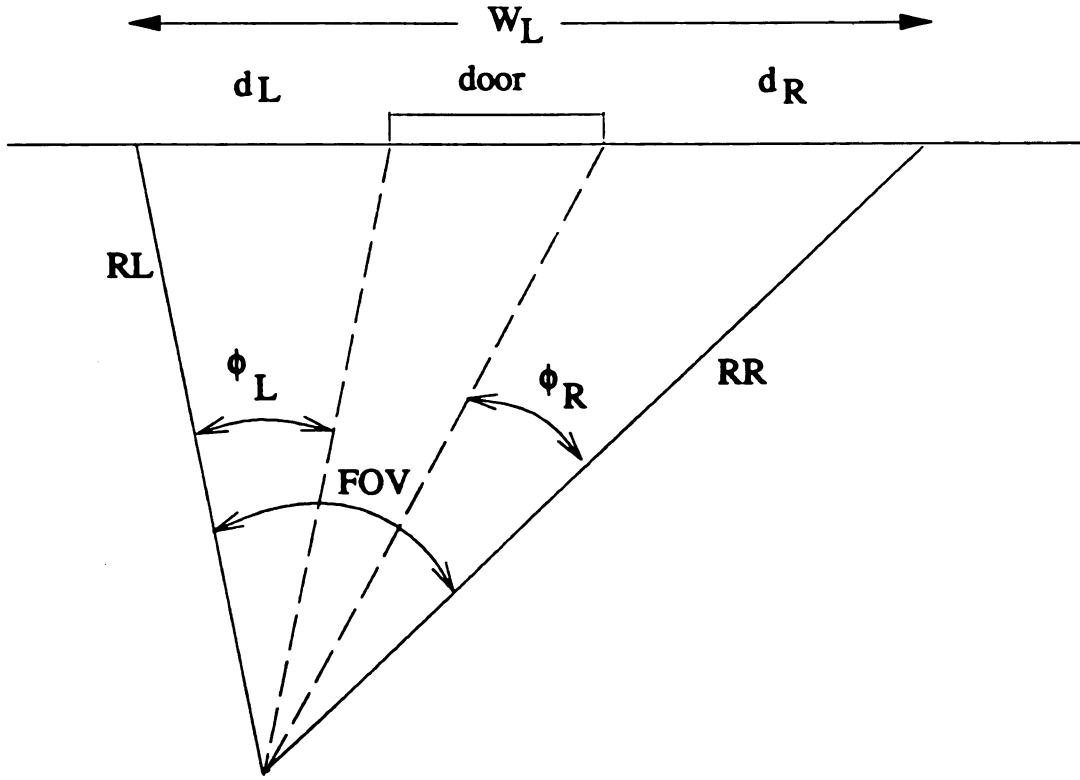


Figure 5.1. Field of view geometry

for this purpose. If a landmark is within the field of view, the sensing geometry must determine the position of the landmark relative to the edges of the field of view, and assess an absolute metric scale. If we know the angle of the optical axis relative to the acoustic axes of the ultrasonic array, then we can determine the range to the edges of the field of view. Figure 5.1 illustrates the geometry within the field of view (FOV).

Constraint C_3 provides the necessary reference to assign the correct ultrasonic range value to estimate RL and RR . Since we know the value of FOV for the application lens, the Law of Cosines provides W_L , the length of the domain context visible within the field of view.

$$W_L = \sqrt{RL^2 + RR^2 - 2RLRR \cos \text{FOV}}$$

The candidate landmark footprint, as presented in Section 4.1.1, will partition the domain context into the three annotated segments seen in Figure 5.1. Using similar triangles and the Law of Sines we can calculate the length of the segments d_L and d_R .

$$d_L = \frac{RL \sin \phi_L}{\sin(\pi - \phi_L - \alpha)}$$

$$d_R = \frac{RR \sin \phi_R}{\sin(\pi - \phi_R + \alpha)}$$

Similar triangles with bases on the image plane and the domain context will provide a means of determining ϕ_L and ϕ_R . We define LJ and RJ as the left and right footprint of the verticle edge ribbon, respectively, h as the horizontal pixel width in our CCD camera, in millimeters per pixel, and mid as half the number of horizontal pixels across the image plane. With a camera lens of focal length, f , ϕ_L and ϕ_R can be calculated.

$$\phi_L = \frac{\text{FOV}}{2} - \arctan\left(\frac{2h(\text{mid} - LJ)}{f}\right)$$

$$\phi_R = \frac{\text{FOV}}{2} - \arctan\left(\frac{2h(RJ - \text{mid})}{f}\right)$$

Combining these equations we present a composite formula for the observed landmark using the ranges RL and RR estimated by our ultrasonic sensors. RL and RR

represent our two range *sides* of the sensing geometry, and the dependence on these two sides led us to call this our Two Side Geometry.

Two Side Geometry

ribbonwidth =

$$\begin{aligned}
 & \sqrt{RL^2 + RR^2 - 2RLRR \cos FOV} \\
 & - \frac{RL \sin \left(\frac{FOV}{2} - \arctan \left[\frac{2h(mid-LJ)}{f} \right] \right)}{\sin \left(\pi - \left(\frac{FOV}{2} - \arctan \left[\frac{2h(mid-LJ)}{f} \right] \right) \right) - \arcsin \frac{RR \sin FOV}{\sqrt{RL^2 + RR^2 - 2RLRR \cos FOV}}} \\
 & - \frac{RR \sin \left(\frac{FOV}{2} - \arctan \left[\frac{2h(RJ-mid)}{f} \right] \right)}{\sin \left(FOV - \left(\frac{FOV}{2} - \arctan \left[\frac{2h(RJ-mid)}{f} \right] \right) \right) + \arcsin \frac{RR \sin FOV}{\sqrt{RL^2 + RR^2 - 2RLRR \cos FOV}}}
 \end{aligned}$$

This is an equation in four unknowns: RL and RR, the two estimated ranges along the edges of the field of view; and LJ and RJ, the two candidate landmark footprints classified by the image processing suite. This equation has been checked against ground truth with a large suite of images, and its error analysis will be presented in Section 5.4. In Section 5.5, we will examine the use of this geometry and the accurate estimation of its four variables. We will see that the ability to obtain accurate variables will define a navigation control envelope. In the next section we will attempt to improve upon this sensing geometry, and present our Short Side Geometry. We explored this geometric approach because it performed the same ranging function as our Two Side Geometry, but only required the use of one ultrasonic range reading. This motivation for economy led to an approach that eventually provided superior accuracy.

Angle with Surface		
Look	RL	RR
90°	30°	30°
70°	45°	10°
50°	60°	10°

Table 5.1. Two Side Geometry acoustic angles

5.3 Short Side Geometry

In this section we will attempt to improve upon the Two Side Geometry presented in the preceeding section. Our motivation was to more accurately obtain RL and RR shown in Figure 5.1. We have seen in Section 4.3 that ultrasonic accuracy improves as the acoustic axis approaches the surface normal. With a circular array of ultrasonic sensors, one or two of the 24 sensors will be nearer the domain surface to be ranged than the other remaining sensors. The Two Side Geometry does not use the most accurate ultrasonic readings. In fact, if one of the side readings was very accurate, the other range side is certain to be very inaccurate at most look angles. Table 5.1 illustrates acoustic axis angles with the ranged surface normal from arbitrary look angles.

Noting the oblique acoustic angles that our Two Side Geometry utilizes, together with the experimental results in Section 4.3, impressive metric accuracy should not be expected. We postpone the examination of these results until Section 5.5. The motivation for using the most accurate ultrasonic sensor reading led us to the development of the Short Side Geometry.

We previously noted that when the acoustic axis aligned with the surface normal, ultrasonic ranging was most accurate. When a frame is acquired, the shortest ultrasonic range obtained from sensors within the field of view sector corresponds

to the sensor axis nearest the surface normal. Constraint C_3 ensures that we know the orientation of the camera relative to the ranging sensors, so we can geometrically derive RL and RR from the shortest returned ultrasonic range. Let us annotate the shortest range value, or the *short side*, as $sdis$, and the angle between the optical axis and the surface normal as θ . Then the Short Side Geometry derives the two sides RL and RR from θ and $sdis$, and determines the landmark's metric attribute using the formula presented previously.

Short Side Geometry

$$RL = \frac{sdis}{\sin\left(\frac{\pi}{3} + \theta\right)}$$

$$RR = \frac{sdis}{\sin\left(\frac{\pi}{3} - \theta\right)}$$

$$\left(\frac{-\pi}{3} < \theta < \frac{\pi}{3}\right)$$

This sensing geometry amends the Two Side Geometry and ensures that only the most accurate ultrasonic range value is used. The Short Side Geometry reduces expected ultrasonic error at the price of introducing error in θ . Since the ultrasonic array is laid out along Rome's perimeter in sectors of 15° , our uncertainty in which sensor is indeed nearest to the surface is as great as 7.5° . The sine function is sensitive to error over portions of its domain, but overall we feel the Short Side Geometry is a strong improvement, as we will note in the next section.

5.4 Error Analysis

Overcoming error is a strong challenge for any robotic navigator. Some of this error is due to systemic causes such as ultrasonic range error, and some error is due to a methodological weakness towards a subset of the domain. StickRep was designed to handle erroneous data and ambiguous domain features in an efficient manner, but what is the nature of the composite error it must handle? We examined ultrasonic sensor error in Section 4.3, but what methodological weaknesses exist in our sensing geometries? Let us examine our sensing geometry in greater detail. Figure 5.2 revisits our geometrical framework.

In this Figure, RL and RR retain their previous definition, but we have now annotated the sensing and domain dynamics in finer granularity. Angle δ represents the angle between the optical axis, opax, and the surface. Rome's ultrasonic sensor locations are represented by tick marks around the perimeter, with r being the radius of Rome's sensing head. The ideal rays from Rome's center through the ultrasonic sensors are annotated as L and R. If the camera lens field of view (FOV) exactly matched the sector defined by an integer multiple of ultrasonic sensors around the perimeter, then $L = RL$ and $R = RR$. In our application the sensors are placed every 15° around the perimeter, and our chosen lens possessed a field of view of 55.817° so RL and RR approximate the desired rays L and R along the edges of our field of view in our geometry. This approximation is examined in the next section.

In the upper portion, or left sensing side of Figure 5.2, the dashed sector delimits the 23.6° sensing cone of a typical ultrasonic sensor. When an ultrasonic wavefront reflects back and is captured by the sensor, its range value is directly related to the sensor's distance from the nearest object *within its sensing cone*. Unless the ideal ray L or R is normal to the surface, the ultrasonic wavefront will first strike the surface adjacent to the point where the ideal ray meets the surface. This introduces another

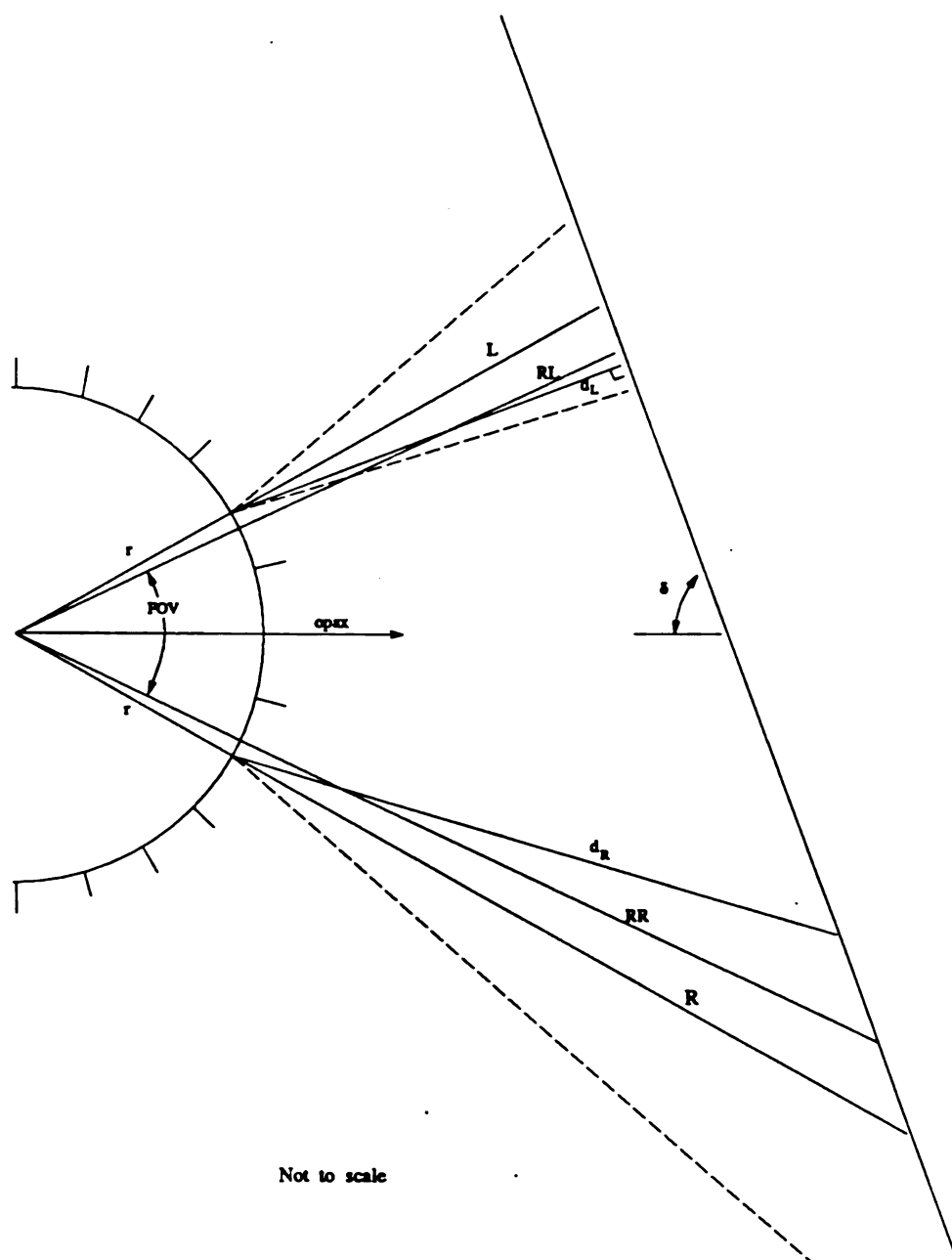


Figure 5.2. Rome's sensing geometry

source of sensing error. In Figure 5.2 d_L and d_R represent the range value actually returned by a particular ultrasonic sensor. In the left sensing side, the surface normal intersecting the sensor location lies within the sensor's cone, in the right sensing side, the surface normal intersecting the sensor lies outside the sensor's cone, and d_R represents the range along the left sensing cone limit of the reflected wavefront. A third geometric case is when the surface normal intersecting the sensor location again lies outside the sensing cone, but in this case, it lies outside the W_L -L-R sensing triangle. Note that in all three cases, the returned range value d_L or d_R is not the desired range along the ideal ray L or R. This error, exaggerated in the Figure, is a portion of the error we examined empirically in Section 4.3, and will be compounded by the gross error observed at oblique acoustic angles. We shall develop a worst case model of this behavior, and combine it with our geometric approximation error in Section 5.4.2.

5.4.1 Error Analysis of Geometric Approximation

There are three separate cases of geometric approximation error in our sensing framework. These cases were introduced in the previous section and were derived from Figure 5.2. This approximation error corresponds to the difference between the ideal rays L and R, and the rays RL and RR defined by the actual edges along the field of view. **Case 1** occurs when the surface normal to the sensor location lies within the sensor's cone, that is, $48^\circ \leq \delta < 72^\circ$. **Case 2** occurs when the surface normal to the sensor location lies outside the sensing cone, but within the W_L -L-R sensing triangle; $72^\circ \leq \delta \leq 90^\circ$. **Case 3**, $\delta < 48^\circ$, is when the surface normal to the sensor location again lies outside the sensing cone, but also lies outside the W_L -L-R sensing triangle. These angular based case definitions are symmetric about 90° , and were developed using similar triangles and the Law of Sines. We now present the *actual* geometric Definition of RL and RR that we have approximated in our Short Side and Two Side

Geometries with the returned range values d_L and d_R plus the head radius, r .

Case 1, $48^\circ \leq \delta < 72^\circ$

$$RL = \frac{d_L}{\sin\left(\delta + \frac{FOV}{2}\right)} + \frac{\sin(150 - \delta)}{\sin\left(\delta + \frac{FOV}{2}\right)} r$$

$$RR = \frac{d_R}{\sin\left(180 - \delta + \frac{FOV}{2}\right)} + \frac{\sin(\delta - 30)}{\sin\left(180 - \delta + \frac{FOV}{2}\right)} r$$

Case 2, $72^\circ \leq \delta \leq 90^\circ$

$$RL = \frac{\sin(18 + \delta)}{\sin\left(\delta + \frac{FOV}{2}\right)} d_L + \frac{\sin(150 - \delta)}{\sin\left(\delta + \frac{FOV}{2}\right)} r$$

$$RR = \frac{\sin(198 - \delta)}{\sin\left(180 + \frac{FOV}{2} - \delta\right)} d_R + \frac{\sin(\delta - 30)}{\sin\left(180 + \frac{FOV}{2} - \delta\right)} r$$

Case 3, $\delta < 48^\circ$

$$RL = \frac{\sin(138 - \delta) \sin(150 - \delta)}{\sin(\delta + 30) \sin\left(\delta + \frac{FOV}{2}\right)} d_L + \frac{\sin(150 - \delta)}{\sin\left(\delta + \frac{FOV}{2}\right)} r$$

$$RR = \frac{\sin\left(\delta + \frac{FOV}{2} - 72\right) \sin\left(\delta - 60 + \frac{FOV}{2}\right)}{\sin\left(240 - \delta - \frac{FOV}{2}\right) \sin(210 - \delta)} d_R + \frac{\sin\left(\delta - 60 + \frac{FOV}{2}\right)}{\sin(210 - \delta)} r$$

If the look angle, δ , were known by Rome, then we could enhance our metric accuracy by calculating range values using these formulas. In the next section, we

δ	$L \rightarrow RL$	$R \rightarrow RR$	$d_L \rightarrow L$	$d_R \rightarrow R$
90°	2.0%	2.0%	4.3%	5.1%
70°	4.1%	0.57%	15%	2.1%
50°	9.1%	0.72%	21%	1.5%

Table 5.2. Composite ranging error summary

will evaluate our approximation to these formulas in the worst case, and combine it with the ultrasonic error summaries we presented in Section 4.3.

5.4.2 Composite Range Error Model

Rome experiences ultrasonic ranging errors of two distinct types. The first type is the ranging error we observed in Section 4.3 with particular weaknesses on soft surfaces and at oblique acoustic axis angles. The second error type is due to the approximations our Short Side Geometry must make to the exact ranges presented in the previous section. In Table 5.2, we have evaluated this approximation error in the worst case, and summarized the empirical results we obtained in Section 4.3 at three arbitrary look angles.

Table 5.2 summarizes the composite ultrasonic error models we used to drive the extraction simulations to be discussed in Section 5.5. These error models were developed from empirical range experiments in a real domain together with a worst case analysis of our approximation to the theoretically correct range. In the next section we present a linear first order error approximation of range extraction and landmark metric inference using our sensing geometries.

5.4.3 Linear First Order Error Model

We desired to get a quantitative approximation for the percent error we might observe using the Short Side and Two Side Geometries, prior to implementing these approaches. Starting with the geometric formulas presented in Sections 5.2 and 5.3, we derived a partial derivative evaluated at a particular look angle and multiplied it by a worst case measurement error factor as shown in Figure 5.3.

Two Side Geometry

$$\Delta_{metric} \approx \left(\frac{\partial}{\partial RL}, \frac{\partial}{\partial RR}, \frac{\partial}{\partial LJ}, \frac{\partial}{\partial RJ} \right) \Big|_{stare} \cdot (\Delta RL, \Delta RR, \Delta LJ, \Delta RJ)$$

Short Side Geometry

$$\Delta_{metric} \approx \left(\frac{\partial}{\partial sdis}, \frac{\partial}{\partial \theta}, \frac{\partial}{\partial LJ}, \frac{\partial}{\partial RJ} \right) \Big|_{stare} \cdot (\Delta sdis, \Delta \theta, \Delta LJ, \Delta RJ)$$

Figure 5.3. Linear first order error model

The complete partial derivatives were obtained, but due to their length, we present them graphically in Appendix B. We evaluated these approximation error models at three arbitrary look angles, or stares. Worst case error in θ was 7.5° , as discussed earlier. Empirical experiments supported a worst case error for ΔLJ and ΔRJ to be five pixels. Table 5.3 summarizes this linear first order error approximation for both sensing geometries at three stares. Table entries are the worst case landmark metric error, in inches, using the linear first order estimate.

The sensitivity of the Short Side Geometry to error in θ can be seen particularly at the more oblique stares. The Two Side Geometry appears more robust at oblique

Geometry	90°	70°	50°
Two Side	2.4	3.9	4.7
Short Side	0.40	5.5	12

Table 5.3. Worst case landmark error using linear first order estimate

δ	90°	70°	50°
Δ metric	0.28	3.2	6.6

Table 5.4. Accuracy resulting from reducing $\Delta\theta$ to 3.75° using Short Side Geometry

stares, but its accuracy is still moderate at a 90° stare. We doubted the ability to perform inexact matching with this level of expected metric accuracy. The Short Side Geometry appeared to offer the absolute accuracy that could be pivotal to a landmark classification scheme. A control envelope was developed to take advantage of the accuracy obtained at 90° stares by the Short Side Geometry, while simultaneously discouraging oblique stares by a cost function.

Prior to selecting the Short Side Geometry for implementation, we again used the linear approximation of Figure 5.3 to explore classification and robustness questions. If we installed twice as many ultrasonic sensors, reducing $\Delta\theta$ to 3.75°, what benefit in worst case performance would be observed when employing the Short Side approach? Table 5.4 summarizes the answer. The advantage gained from the extra hardware appears to be insufficient to justify the effort.

Another question concerned how much $\Delta\theta$ error the Short Side Geometry could handle and still deliver accuracy at an arbitrary benchmark, say two inches. Table 5.5 summarizes the results at arbitrary stares. It appears very tolerant at blunt look angles, but sensitive at oblique angles.

δ	$\Delta\theta$
90°	$\leq 57.0^\circ$
70°	$\leq 1.8^\circ$
50°	$\leq 0.5^\circ$

Table 5.5. Short Side θ tolerance for error within two inches

These results motivated us to explore large scale simulations of landmark classification employing the Short Side Geometry. With a supporting navigation control scheme, accurate metric inference could be obtained. In the next section we will present our simulation results of landmark measuring using the Short Side and the Two Side Geometries. We will see that these simulation results are consistent with the error models we have explored in the previous sections.

5.5 Monte Carlo Experiments

We were very interested in the behavior of the two sensing geometries in the presence of the error models we developed in Section 5.4.2. The Short Side Geometry appeared to promise the most capability, but we were concerned about its sensitivity to θ error. Both Schneider and Wang had concluded that rotational, or angular uncertainty was the most sensitive parameter in their applications [134, 155], and we have already seen evidence to support this conclusion in our problem. Uncertainty is often a function of measurement errors that have additive effects, but their work found that angular uncertainty had multiplicative effects.

We conducted over 100 thousand Monte Carlo trials to examine the percent of metric error in measuring landmarks of known sizes using either of our proposed sensing geometries. Ten thousand trials at a given optical axis orientation were obtained using each sensing geometry and the error models developed in the last chapter. The

θ	Prob($\pm 5.55\%$)
90°	0.92
70°	0.41
50°	0.16

Table 5.6. Probability of Short Side Geometry measuring within 2 inches

following six Figures illustrate with histograms the performance of both geometries at selected arbitrary look angles.

As expected the Short Side Geometry delivered accuracy superior to the Two Side Geometry. The effect of the uniformly distributed $\Delta\theta$ can be seen in the wide shape of the Short Side results. The Two Side results have the characteristic Gaussian shape, and the offset bias varied with the camera orientation. Useful accuracy is only seen with the orthogonal look angle of the Short Side Geometry. In our domain, landmark classes are often separated by two inches, or less, in their metric attribute. Table 5.6 summarizes the probability that the Short Side Geometry measured the landmark with two inches, or 5.55%, at three arbitrary look angles.

From these results we concluded that the Short Side Geometry offered better performance in a general domain than the Two Side Geometry, and that a control envelope should be defined to ensure that sensing occurs when the data extracted from the sensors is most likely to be accurate. Obtaining range data from sharp look angles promised the highest accuracy, so an optical axis orthogonal to the axis of motion would be a good design for very fine metric work.

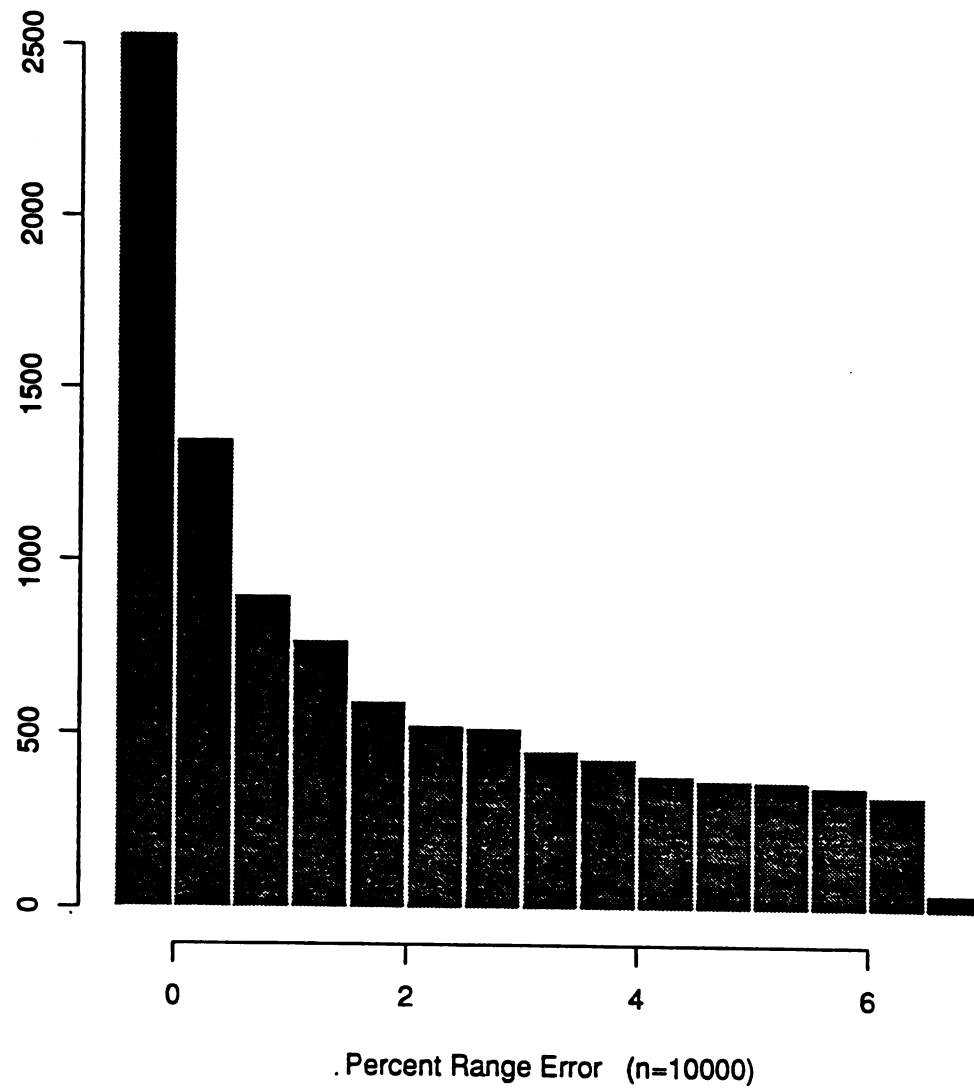


Figure 5.4. Short Side Geometry, $\theta = 90^\circ$

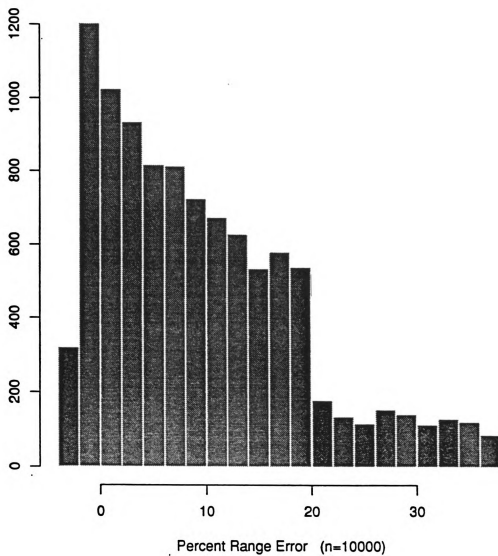


Figure 5.5. Short Side Geometry, $\theta = 70^\circ$

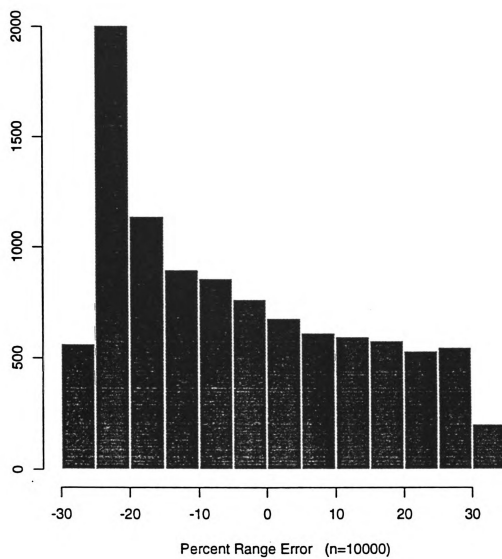


Figure 5.6. Short Side Geometry, $\theta = 50^\circ$

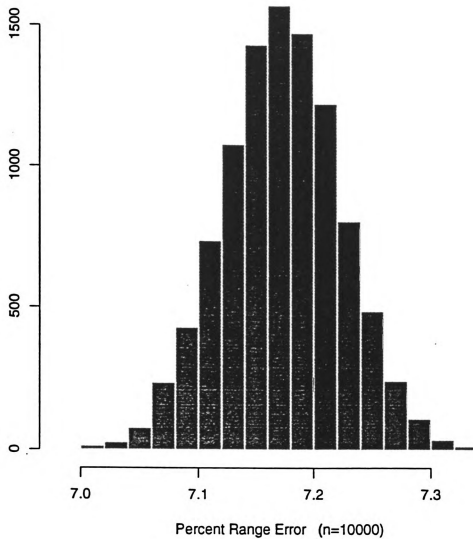


Figure 5.7. Two Side Geometry, $\theta = 90^\circ$

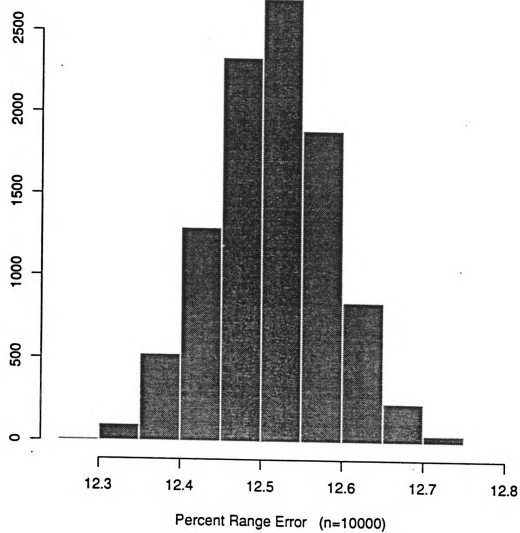


Figure 5.8. Two Side Geometry, $\theta = 70^\circ$

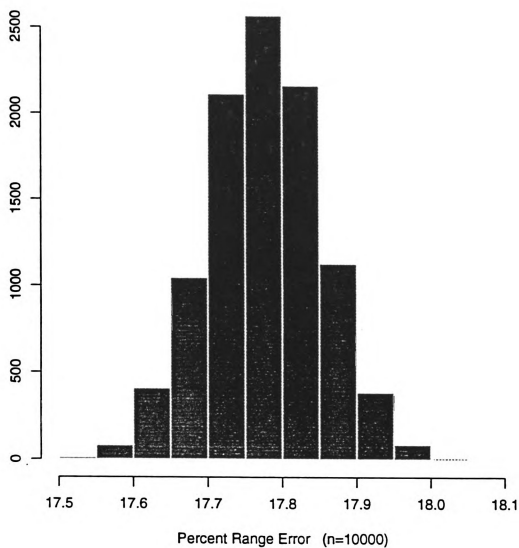


Figure 5.9. Two Side Geometry, $\theta = 50^\circ$

5.6 Summary

In this chapter we have presented two sensing geometries that support landmark classification in a structured domain. These methodologies fuse the sensing capabilities of a monocular camera and an array of ultrasonic transducers. This geometric approach enables depth information to be attributed to an extracted domain feature. These geometries successfully cast vision and ultrasonic sensing in roles where each can operate accurately, while simultaneously supporting real time navigation. While these sensing frameworks each led to an overall sensing footprint, and a navigation control envelope, locomotion and path planning issues have not been overly constrained. We developed detailed composite error models that were injected into landmark classification simulations that demonstrated the potential of both these sensing geometries. The Short Side Geometry displayed the greater accuracy potential and will be extensively employed in the studies of the next chapter.

Both of these original sensing geometries offer a more general landmark classification solution to other applications. It is noteworthy that neither sensing geometry requires ultrasonic or camera calibration, and both offer real time performance, at very affordable hardware costs.

CHAPTER 6

Inexact Graph Matching

The developments presented in the previous chapters provide the representational model and the sensing strategy to perform autonomous pose recovery in a structured domain. In the following sections we will discuss the symbolic feature ambiguity inherent in operating domains, and the representational challenges that the representation must handle. Inexact feature attributes, particularly sensed metric data, present difficult matching challenges. Heuristic and deterministic algorithms for error recovery will be presented along with simulation results obtained using realistic error models. These simulations explore different approaches toward developing a robust, working solution to the Where-am-I Problem. Performance trade-offs between various heuristics will be studied. The results show that our approach can provide *accurate* pose recovery in a *short* period of time.

6.1 Representation

In Sections 3.2 and 3.3 we examined representations used in mobile robotics and presented the StickRep structure. We saw that StickRep was a $1\frac{1}{2}$ dimensional structure that provided a minimal representation for our robotic navigator. Each edge contains some or all of the following attributes:

$$\left\{ \begin{array}{l} type \in \{corner, doorway, elevator, hallway, pattern, stairwell\} \\ metric \in \mathbb{R}^+ \\ prototype \in \{doorjam, hallsegment, pattern\} \\ pattern \in \{\clubsuit, \diamond, \spadesuit, \heartsuit\} \end{array} \right.$$

Nodes hold adjacency information of the incident feature edges and contain the following attributes:

$$\left\{ \begin{array}{l} adjacency \ angle \in \{0, \dots, 360\} \\ global \ coordinates \in \{X^W, Y^W, Z^W, \alpha, \beta, \gamma\} \end{array} \right.$$

StickRep is implemented as a doubly linked list of records. These records encode sequences of feature data. If we view the database and the sensed feature data as a one dimensional sequence of attributed edges, then feature matching becomes a string matching problem with each attributed edge corresponding to a terminal in our edge type alphabet. The sensed feature map under construction becomes a graph of attributed terminal symbols [46]. This design greatly simplifies the mechanics of the matching and a string matching scheme was modified to handle ambiguous metric information [94]. The pattern field is a dynamic entry which attempts to capture feature information that is supplemental to the feature type. For example, a poster on a door is often sensed as a pattern within a larger feature that is strongly conjectured to be of edge type door. This feature information is attributed to a door edge for subsequent disambiguation and confirmation. Feature information is dynamic, so expanding the context may provide greater confidence in a potential match. In the next section we will discuss how StickRep, together with our matching scheme, handled symbolic feature ambiguity.

$\angle = 90^\circ$	door30	door32	door36	door40	wall
door30	0	0	0	0	0
door32	0	0	0	0	0
door36	0	0	0	0	0
door40	0	0	0	0	0
wall	0	0	0	0	115

Table 6.1. Domain ambiguity: Number of edge pairs with adjacency angle of 90°

$\angle = 180^\circ$	door30	door32	door36	door40	wall
door30	11	0	0	0	11
door32	0	0	0	0	4
door36	0	0	28	0	183
door40	0	0	0	0	11
wall	10	4	181	11	0

Table 6.2. Domain ambiguity: Number of edge pairs with adjacency angle of 180°

6.2 Ambiguity

Despite the structure provided by our assumed domains, great feature ambiguity still exists even without the confusion injected by sensor error. Complete knowledge of the edge types, and the adjacency angles between the edges, is insufficient for unique identification. Ambiguity can be strong without larger context. The third floor of the Engineering Building is just one of the domains we examined, but it provides an illustrative example of ambiguity, and is portrayed in Tables 6.1 - 6.9. Each Table contains the number of occurrences of a particular edge sequence on the building's third floor. For example, Table 6.2 shows that there exist 28 instances of adjacent, coplanar 36 inch door pairs. Table 6.5 shows that there are 152 instances of a 36 inch door with coplanar wall surfaces on both adjacent sides. Strong ambiguity exists,

$\angle = 270^\circ$	door30	door32	door36	door40	wall
door30	0	0	0	0	0
door32	0	0	0	0	0
door36	0	0	0	0	1
door40	0	0	0	0	0
wall	0	0	2	0	98

Table 6.3. Domain ambiguity: Number of edge pairs with adjacency angle of 270°

$\angle = 180^\circ$	door30	door32	door36	door40
door30	4	0	4	2
door32	0	0	1	0
door36	5	2	118	3
door40	0	0	4	4

Table 6.4. Domain ambiguity: Three Edges: $\langle door \rangle \langle wall \rangle \langle door \rangle$, adjacency angles of 180°

particularly involving edges representing the most common door width of 36 inches.

Tables 6.8 and 6.9 illustrate the variety of door and wall edges within their respective edge types. Metric information is often necessary to disambiguate candidate subgraphs during matching. For example, Table 6.9 indicates that only 62 of the 431 wall edges occurring on the Engineering Building's third floor are 160 inches or greater

door30	0
door32	4
door36	152
door40	11

Table 6.5. Domain ambiguity: Three Edges: $\langle wall \rangle \langle door \rangle \langle wall \rangle$, adjacency angles of 180°

$\angle = 180^\circ$	door30	door32	door36	door40
door30	0	0	1	2
door32	0	0	0	0
door36	0	2	114	3
door40	0	0	2	4

Table 6.6. Domain ambiguity: Four Edges: $\langle door \rangle \langle wall \rangle \langle door \rangle \langle wall \rangle$, adjacency angles of 180°

$\angle = 180^\circ$	door30	door32	door36	door40
door30	0	0	0	0
door32	0	0	1	0
door36	5	1	109	2
door40	0	0	4	4

Table 6.7. Domain ambiguity: Four Edges: $\langle wall \rangle \langle door \rangle \langle wall \rangle \langle door \rangle$, adjacency angles of 90°

in length.

The Where-am-I Solution will be examined within this ambiguous operating domain. Feature ambiguity and sensing error present difficult obstacles for our solution. The robot navigator must extract a vertical ribbon edge feature and classify the landmark candidate while continuing forward motion. It will attempt to match a classified landmark to its domain map, but due to the great feature ambiguity in this domain, multiple matches are almost guaranteed. Additional landmarks must be obtained, expanding context so that a unique match can be found. Suppose the robot navigator were to start at a random location and orientation on the third floor, autonomously aligning its motion vector down the hallway axis. Figure 6.1 illustrates, for every possible starting location, the number of adjacent edges the navigator would have to accurately sense and classify, before the assembled subgraph of edges would uniquely

	blank	old window	no window	new window	2 new windows	poster
door30	22	0	0	0	0	0
door32	4	0	0	0	0	0
door36	212	54	10	3	104	29
door40	11	11	0	0	0	0

Table 6.8. Domain ambiguity: Door type census

	>0	≥ 12	≥ 16	≥ 32	≥ 48	≥ 64	≥ 80	≥ 96	≥ 112	≥ 128	≥ 160
wall	431	415	370	328	223	202	160	121	102	88	62

Table 6.9. Domain ambiguity: Wall type census

match a starting location in the master building graph. Maximum ambiguity occurs at several starting locations that correspond to 36 inch doors embedded in a highly regular metric feature sequence. These starting locations represent the areas that would require the most context before the robot navigator could determine its pose. This disambiguation becomes much more complex in the presence of sensing error and noise. We will examine this issue in detail in subsequent sections.

Ambiguity could be reduced by designing the edge type definitions much finer in granularity. Subdividing the door and wall edge types into multiple edge types would yield matching subgraphs with considerably less context. This corresponds to incorporating many of the attributes directly into the edge definitions and classification schemes. Although this methodology appears attractive, its implementation may quickly reveal an inability to accurately sense the features necessary for robust classification. The necessity of real time processing further restricts the ability to extract fine feature detail [102, 106, 138, 148].

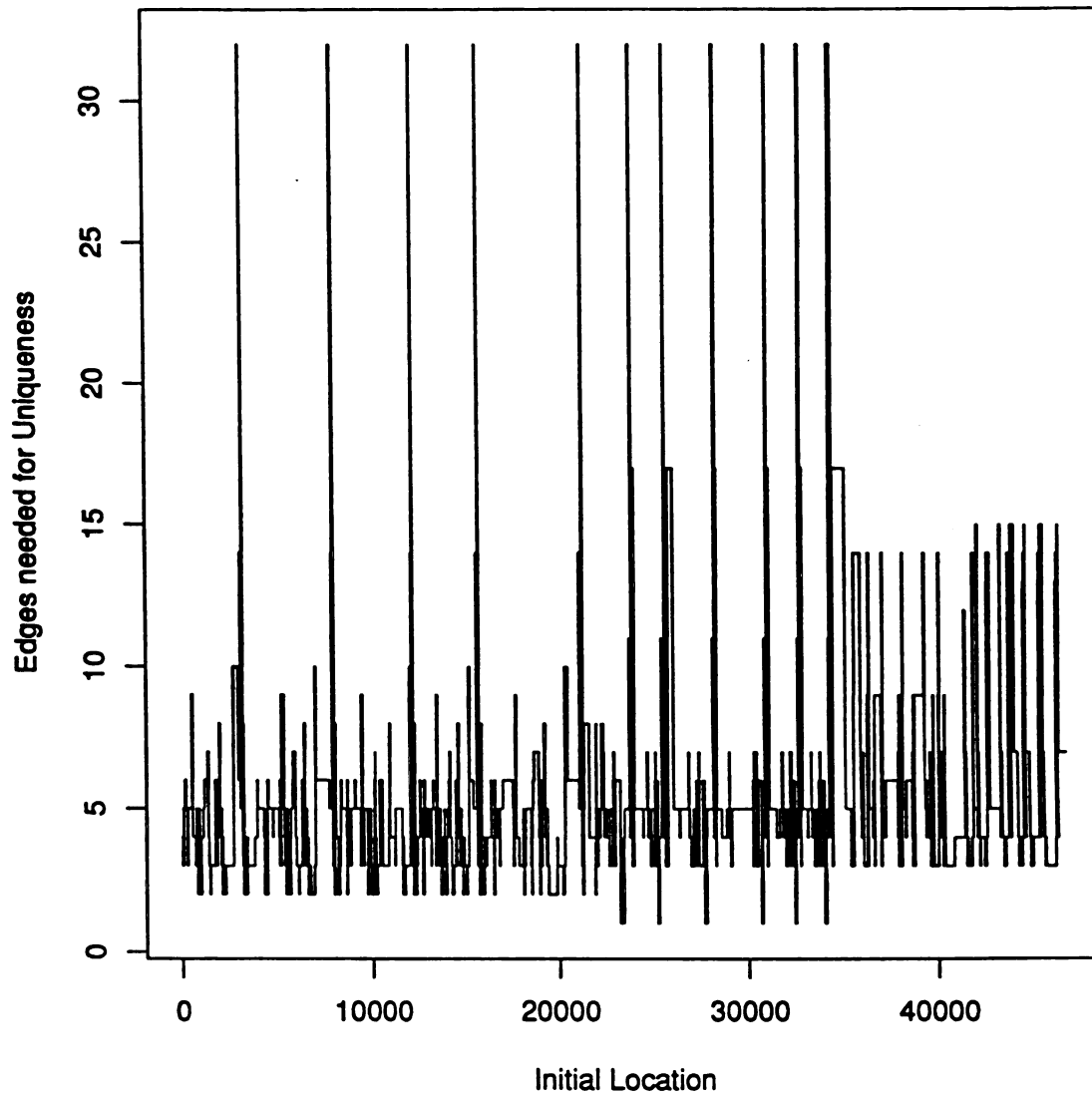


Figure 6.1. Starting location ambiguity of MSU Engineering Building

6.3 Simulations

Over 80,000 simulated navigation runs were conducted using the StickRep model database presented, under various simulated conditions. For each simulation, Rome was initially placed at a random location and orientation in the hallway freespace. The following assumptions were made throughout the simulation experiments:

- A_1) Rome has a valid StickRep database of the operating domain.
- A_2) Rome's ultrasonic and monocular sensors behave as modeled in Chapters 4 and 5.
- A_3) Rome has the capability to autonomously align its forward motion vector to within a few degrees of the ultrasonically sensed hallway axis. This assumption is based on the exploratory work demonstrated by Dulimarta [47].
- A_4) Rome has the capability to autonomously center, within a few inches, its position in the hallway freespace and maintain this centering during forward motion without significant oscillation. Again, this assumption is based on the empirical work of Dulimarta using only ultrasonic sensing.

Rome may commence travel either left to right or right to left relative to the orientation of the camera. Each simulation experiment consisted of a suite of 10 independent trials, with each trial consisting of 1000 different random starts. The experiment was implemented with a sliding window along the chosen surface of the hallway freespace. Frames were acquired each 0.2 seconds, retaining only frames containing a vertical edge ribbon. The matching process took a worst case time of only 40 milliseconds per frame acquired, and the image processing described in Section 4.1 consumed another 100 milliseconds. Conservatively then, Rome only requires 0.2 seconds between frames to solve this problem. All these times were taken

from simulations performed on a Sparc 1 workstation. Upgrading Rome's computer from the 386SX laptop to a Sparc 1, sans monitor, is currently underway. To ensure that feature overlap exists with this field of view, a subsequent frame acquired 0.2 seconds later enforces a maximum velocity of 30 feet per second. This is well over the maximum Labmate base speed of 2.6 feet per second so pursuing this approach to the Where-am-I Problem is limited by the mechanical capability of Rome's Labmate base, and not by implementation of our approach. Any hallway navigator cruising at speeds in the neighborhood of 30 feet per second would very likely be viewed as a safety hazard, particularly since this is well beyond the reaction horizon using our current ultrasonic sensors (and close to world record sprinting time). Figure 6.2 describes the simulation we conducted. In the next section we will discuss how step (2) in Figure 6.2 was implemented using the error models developed in Section 4.3.

6.4 Error

Error is inherent in any robotics application. The Where-am-I Problem is challenging not just due to ambiguity in a dynamic operating domain, but also due to unavoidable sensing errors. If we had domain ambiguity, but no error, then shaft-encoded wheels could support accurate dead reckoning and there would be no pose uncertainty *once* our navigator had disambiguated its correct pose. In the presence of sensing error, navigation and control become more difficult because we can never have complete confidence in our sensed data. We assume that our navigator does not possess its initial pose so any sensed data that is consistent with *some* location in its building database has to be accepted as long as it remains consistent with any previously sensed data.

In the next section, we will explore our Where-am-I solution in the operating domain assuming no error. In Section 6.4.2 we will develop a empirically based fused

- (0) Receive random starting location
- (1) Index actual StickRep edge from database
 - (1.1) Establish ground truth location
- (2) Inject modeled error onto actual StickRep edge
 - (2.1) Ultrasonic error
 - (2.2) False positive error
 - (2.3) Missized true postive error
 - (2.4) False negative error
- (3) Determine sensed StickRep edge
- (4) Concatenate into sensed subgraph
- (5) Index sensed subgraph from building database
 - (5.1) Assemble set of possible locations from matches with database
- (6) Examine possible location set
 - (6.1) If more than one possible location
 - (6.1.1) Increment metric travel indicator
 - (6.1.1.1) Inject mechanical wheel error
 - (6.1.2) Goto (1) for next edge
 - (6.2) Else, if one possible location
 - (6.2.1) Propose symbolic pose
 - (6.2.2) Compare proposed pose with ground truth
 - (6.3) Else, if zero possible locations
 - (6.3.1) Raise sensing error flag

Figure 6.2. Simulation Algorithm

error model that will be supplemental to our model of Section 4.3 in simulating the injected error of step (2) in Figure 6.2.

6.4.1 No Error

Examining our Where-am-I Solution in a perfect sensing domain illustrates the ambiguity in the feature space, and presents a best-case situation. This simulation experiment, like the other five we will examine later, consisted of a suite of 10 independent trials, with each trial composed of 1000 simulated navigation runs from random starting locations and orientations. Figure 6.3 is a representative histogram illustrating the number of edges acquired before context disambiguation resolved a single possible match to the building database. In all cases, the proposed pose was the correct actual pose, as expected without any sensing error.

Frames were acquired and assembled into StickRep edges. When complete domain features were assembled into a candidate subgraph, it was indexed into the building database. The average number of edges acquired before convergence to a unique pose was 5.28. The average StickRep edge in the operating domain had a metric average of 67.64 inches, so the average distance Rome travelled before acquiring its pose was 29.76 feet. We saw in the previous section that computationally we could acquire this average pose fix in just over one second, but Rome's Labmate base is the limiting constraint. Using the present hardware, this Where-am-I Solution would require an average of 11.45 seconds. Let us again emphasize that this simulation result was without sensing error.

This experiment was designed to evaluate the viability of the StickRep model in representing a large building. It also examined the feature ambiguity of the operating domain. These experimental results provide a comparative baseline for the evaluation of the subsequent simulation experiments. In the next section we will discuss the development of our fused sensor error models.

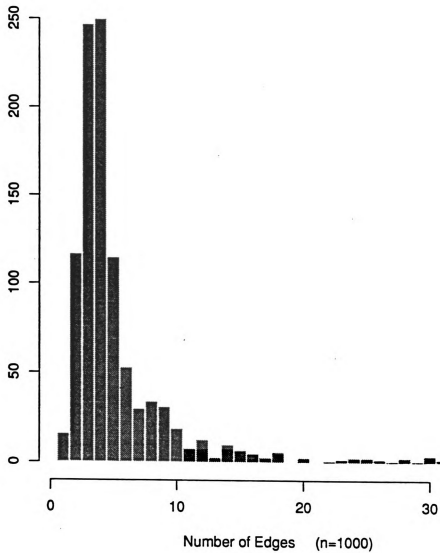


Figure 6.3. Unique Pose without Error

Correct Classifications	83.87 %
True Positives	83.67 %
True Negatives	84.10 %
Misclassifications	16.13 %
False Positives	15.91 %
Missized True Positives	10.20 %
False Negatives	6.12 %

Table 6.10. Empirical Based Error Model

6.4.2 Empirical Based Error

In Sections 4.3 and 5.4 we discussed the modeling of individual sensor error. What was needed to construct a realistic simulation was a fused sensor error model; a model that explains how the Short Side feature extraction algorithm developed in Section 5.3 fails with respect to both vertical ribbon detection and ultrasonic ranging. The Short Side Algorithm was trained on five different suites of coordinated image/ultrasonic frames totalling 157 fused sensor readings. It was then tested on a fused image/ultrasonic suite of 94 fused sensor readings. These data were acquired using the actual Rome platform under various illumination conditions during a 9 month time period. Recalling our individual ultrasonic sensor error model from Section 4.3 we define our control envelope within the domain as a position within 1 foot of the hallway freespace axis, with the optical axis within 7.5° of the surface feature normal. A significant amount of this evaluation data was taken from outside this control envelope, so we feel that these results are conservative. Table 6.10 presents the summary of the Short Side Algorithm's performance on this large library of data. These results constitute the injected error model for the remainder of the simulation experiments as shown in step (2) in Figure 6.2. For classification purposes, recall from Section 4.1 that the vertical ribbon detector was looking for doors, and classifying them into one of six legal sizes.

We feel that these results could be further improved by tuning the parameters of the feature extractors, but our main interest here is exploring the performance of our Where-am-I Solution, not in tuning the sensor suite to an individual problem domain.

In the next section we will discuss how the symbolic StickRep model was used to match subgraphs to the building database despite the inexact feature metric attributes that were the result of sensor error.

6.5 Metric Relaxation Matching

Fuzzy set theory and fuzzy control theory have active, maturing research communities. Much work has been done since Zadeh’s pioneering work in 1965 [159]. The task of interest here is developing the capability to accurately infer a match between a deterministically derived metric attribute and a sensed metric attribute that may or may not contain sensing error. Many applications in fuzzy control handle varying inputs by representing the inputs by their fuzzy membership values [76, 144]. Other image processing applications develop sufficient and ϵ -sufficient statistics [11], or ranking procedures [150] to perform fuzzy pattern classification. An implementational weakness of these approaches revolves around the class membership definition. In our application, although a door type edge must be a member of one of six metric classes, a wall type edge has no such restriction on the metric attribute. How do we define our wall type edge class membership? Lee suggests that relaxation is a much simpler approach in this situation, and may actually yield superior results [98]. We implemented a metric relaxation approach.

If no sensing error were present, a metric relaxation approach would not be necessary. However, significant error is not always present, particularly when Rome operates within the defined control envelope. The presence of significant error can only be noted when sensing a known feature, or when the sensed feature fails to match

- (0) Subgraph fails to match any region of building database
- (1) If relaxation = 0%, set relaxation = 10%
- (2) If relaxation = 10%, set relaxation = 20%
- (3) If relaxation = 20%, FAIL

Figure 6.4. Metric Relaxation Heuristic

a feature edge in the building database. When this matching failure occurs, Rome's executive knows that a sensing error has occurred, but it does not know what edge, or edges, in the assembled subgraph contain significant error. To ensure that the assembled subgraph is without error, the entire sensed subgraph must be discarded. We were concerned that such a decision strategy would threaten our real time control requirements. We will revisit this issue in Section 6.7. We initially developed a stepped metric relaxation heuristic that is invoked when the executive routine notes that significant error has produced a failure to match condition. Figure 6.4 illustrates the heuristic that operates on the feature metric attribute in the building database. Note that after step (2) the relaxed granularity is coarser than the metric accuracy delivered within the specified operating envelope. If our assembled subgraph fails to match despite an arbitrary 20 percent relaxation in metric attributes, the executive routine registers a sensing failure. During the simulation experiments, this heuristic was used in conjunction with other heuristics and when such a failure was noted the attempted pose fix was labeled a failure. We will discuss these experiments in detail in the following section.

43.80% of frame sequences contain error
0.00% of erroneous frame sequences recovered pose
41.70% of frame sequences did not converge
57.30% Correct Poses
1.00% Incorrect Poses

Table 6.11. Performance without heuristics

6.6 Heuristic Error Recovery Experiments

In this section we will discuss in detail the extensive simulations that were conducted using the StickRep Solution to the Where-am-I Problem. We conducted these heuristic error recovery experiments on only one domain - the third floor of the Engineering Building. We felt that this domain offered the most general, and varied real environment. The deterministic method to be developed in the next section will be examined by multiple domains. These experiments were conducted according to the assumptions in Section 6.3. Rome can only react to sensing errors if it knows that an error has occurred. Thus, the executive routine invokes a heuristic only if the subgraph matcher has failed to produce any pose candidates. These heuristic invocations are a modification to step (6.3.1) in our general simulation algorithm presented in Figure 6.2. We emphasize that no heuristic is invoked until a matching failure occurs. Using the error model from Table 6.10 for step (2), we initially conducted an experiment without heuristics to examine the performance of the algorithm under simulated error. Table 6.11 summarizes the results of the median trial of 1000 attempts from the suite of 10 such trials.

The results presented here represent the probability of Rome correctly determining its symbolic pose in the presence of sensing and navigation error. The first line of Table 6.11 illustrates the percent of the simulated frame sequences that contained

error before a pose was concluded, or matching was abandoned due to error. The second line shows the percent of frame sequences that did contain error, zero here, that subsequently converged to a match and a pose was concluded. The third line indicates the percentage of frame sequences that did not converge to a unique match due to error, and were abandoned. In the fourth line the percent of correct poses concluded is noted, and the last line shows that Rome incorrectly determined its pose a small percent of the time because sensing error in its assembled subgraph accidentally matched a region of the building database. We feel that this behavior, corresponding to a situation of false pose confidence, could represent a dangerous situation. Unless this false pose is subsequently corrected by expanded context, undesired behavior certainly could result. The percent of incorrect poses clearly is a figure to watch closely.

6.6.1 First Frame Error

Here we examine StickRep’s performance enhanced by the decision to restart the data acquisition process if an error is noted from a matching failure, but only if this error occurs in the first acquired frame. This approach foreshadows the discussion in the next section. The results in Table 6.12 summarize our solution’s performance. Statistical variation among the 10 trials in this experiment were minor, and were correlated to the injected error rate. The median error rate trial is given. Only a mild overall improvement in performance was observed.

6.6.2 False Door Error

In this experiment we examined a recovery heuristic that was developed to overcome the most common error in our empirical error model. The reason that false doors are observed is due to spurious edge features in the operating domain and an aggressive

43.30% of frame sequences contain error
15.70% of erroneous frame sequences recovered pose
34.80% of frame sequences did not converge
63.50% Correct Poses
1.70% Incorrect Poses

Table 6.12. Performance with First Frame Heuristic

43.67% of frame sequences contain error
40.30% of erroneous frame sequences recovered pose
23.40% of frame sequences did not converge
74.00% Correct Poses
2.60% Incorrect Poses

Table 6.13. Performance with False Door Heuristic

vertical ribbon detector. Here our heuristic removed the door type edges in our assembled subgraph that corresponded to the door type whose *a priori* probability was the lowest. Referring to Table 6.8, in this domain, the 32 inch door is the least likely to be sensed. This deletion occurs sequentially arbitrarily starting with the most recently sensed door, backwards in time. Between each door edge removal, a matching occurs, attempting to recover candidate poses. Again, statistical variation was minor among the 10 trials, and Table 6.13 summarizes performance. A significant performance improvement was observed, but at a cost of almost tripling the dangerous incorrect poses.

6.6.3 Mis-sized Door Error

Here we isolate our examination of metric feature relaxation as discussed in the previous section. The desired performance enhancement is to recover from the mis-sizing,

44.30% of frame sequences contain error
28.37% of erroneous frame sequences recovered pose
31.90% of frame sequences did not converge
66.90% Correct Poses
1.20% Incorrect Poses

Table 6.14. Performance with Mis-sized Heuristic

42.20% of frame sequences contain error
62.80% of erroneous frame sequences recovered pose
8.20% of frame sequences did not converge
84.30% Correct Poses
7.50% Incorrect Poses

Table 6.15. Performance with Full Heuristic Suite

or misclassification of door type edges. An actual door was sensed, and correctly identified as a door, but was misclassified as a door type edge of incorrect width. Table 6.14 presents the performance summary. This heuristic was unimpressive when not used in conjunction with other heuristics, but only mildly increased the number of incorrect poses.

6.6.4 Full Recovery Suite

In Table 6.15 we summarize our solution's performance when all three heuristics are used in combination. An 84.3 percent error recovery rate was observed, but a significant percentage of false poses was observed. A 20 percent increase in matching time was observed, but this 8 milliseconds per frame is not significant in our application.

The correct pose performance noted in Table 6.15 is impressive, but we felt that the incorrect pose percentage represented a significant risk. Reducing the risk of

incorrect pose estimation without degrading correct pose performance would yield a solution approach with strong potential. In the next section we will discuss just such an approach.

6.7 Deterministic Error Recovery

Table 6.15 shows the best heuristic performance towards error recovery. While the applied heuristics were powerful in reducing the percentage of frame sequences that did not converge, incorrect poses occurred frequently. Table 6.16 summarizes results from an experiment where no heuristics were used. Here we discard the entire sensed subgraph if an error is noted by a failure to match the building database, and simply construct another sensed sequence. The observed multi-sequence performance exceeded the accuracy of all previous experiments. The matching was fast enough that despite a large increase in the number of matching requests precipitated by this approach, the absence of heuristics allowed this experiment to actually run faster than the experiments using heuristics. Figure 6.5 is a histogram illustrating the number of frames sensed before a final pose was determined using this approach. Table 6.17 summarizes the comparison of our solution's average performance with, and without modeled error. The average number of edges required before convergence to a unique pose was observed to be 6.84, or 38.55 feet travelled before acquiring its pose. Using the present hardware, this Where-am-I Solution would require an average of only 14.83 seconds to acquire its pose. That pose would be accurate 97.80 percent of the time. This performance compares very favorably with the results we observed without sensing error in Section 6.4.1.

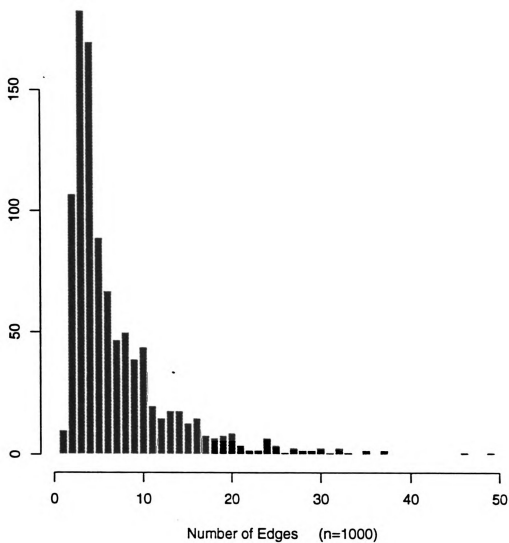


Figure 6.5. Multi-Sequence with Error

6.8 Examining Additional Domains

Our examination of error recovery approaches in the previous sections led us to conclude that our deterministic recovery approach was superior to the heuristic methods. The deterministic error recovery performance was summarized in Tables 6.16 and 6.17. We now will examine the performance of our solution applied to additional domains. We felt that although the Engineering Building domain is very large, many other domains, while smaller in scale, possess even greater feature ambiguity that will further challenge our approach. We conducted the examination of these other domains with only our deterministic error recovery algorithm and used the composite error model we developed for our previous domain. The performance summaries presented represent the median trial of 1000 attempts from a suite of 10 such trials. We will discuss one additional real, one manufactured, and two synthetic domains:

- MSU Engineering Building, Third Floor: $\frac{3}{8}$ mile hallway with both lab and office areas and varied features. Domain previously discussed.
- MSU Wells Hall, Seventh Floor: Office building roughly 15 percent the scale of the Engineering Building with strong feature regularity.
- Synthetic regular polygon with 46 sides and only one unique feature provides a very ambiguous domain.
- Synthetic storefront $\frac{1}{10}$ mile long as viewed from the street provides a structured outdoor domain.
- MSU Wells Hall, Seventh Floor: Modified by introducing a single artificial vertical edge ribbon at the feature edge location farthest from existing reference landmark.

The next application domain was the seventh floor of Wells Hall as shown in Figure 1.1. This domain is roughly 15 percent the scale of the Engineering Building domain, but contains regular feature sequences that demand a much larger context to enable an accurate match. The major feature sequence ring contains a single feature landmark to provide reference context for all other ring features. Table 6.18 summarizes the observed accuracy. The ambiguous domain demanded long frame sequences for reference context and these sequences were thus more likely to contain an sensing error.

We next constructed two synthetic domains to explore particular performance questions. The regular polygon was arbitrarily chosen as a difficult domain with a single reference feature placed to avoid complete ambiguity. The storefront sequence is comparable to the feature sequence viewed during hallway navigation. We assume that street navigation is safe and possible, and designed our simulation to recover pose in a domain scaled to a city block. Real storefronts were used to guide the construction of this domain. Tables 6.19 and 6.20 summarize the accuracies observed in these two domains.

We returned to the Wells Hall domain to explore the potential of disambiguating the domain with the addition of a reference landmark. We introduced a single artificial vertical ribbon edge at the feature edge location most distant from the existing reference landmark. We hoped that this feature manufacturing would decrease the scope of the context searching observed in the actual domain. Table 6.21 illustrates the observed mild improvement in accuracy performance, but the time performance is a more important summary figure.

In Table 6.22 we summarize the time performance observed in all five domains. The regular polygon was clearly a difficult navigation domain even without sensing error, and our approach appears unsuitable for applications of this type. Wells Hall, despite its smaller size, was observed to be a more difficult domain for pose recovery

than the Engineering Building. The artificial landmark placed in Wells Hall improved pose recovery performance significantly from the existing domain, particularly when sensing error is considered. Although this manufactured domain provides only a single example, it demonstrates the potential for improving domain performance with a small degree of domain tailoring. The synthetic storefront provides an example for the application of our solution approach in structured outdoor domains, provided that locomotion and safety issues can be overcome. An outdoor environment still offers insufficient structure for navigation of this nature to be currently practical. If special situations, such as hazardous materials handling, were to remove human dynamics from this domain, then useful pose recovery would become possible. Advances in locomotion hardware could improve these results significantly before the applications approached a real time processing constraint.

41.50% of frame sequences contain error
 94.70% of erroneous frame sequences recovered pose
 0.00% of frame sequences did not converge
 97.80% Correct Poses
 2.20% Incorrect Poses

Table 6.16. Performance of deterministic recovery in Engineering Building

	No Error	Modeled Error
Edges	5.28	6.84
Distance (ft)	29.76	38.55
Time (sec)	11.45	14.83
Accuracy (%)	100.00	97.80

Table 6.17. Overall performance summary

69.70% of frame sequences contain error
 90.82% of erroneous frame sequences recovered pose
 0.00% of frame sequences did not converge
 93.60% Correct Poses
 6.40% Incorrect Poses

Table 6.18. Performance of deterministic recovery in Wells Hall

96.00% of frame sequences contain error
 97.92% of erroneous frame sequences recovered pose
 0.00% of frame sequences did not converge
 98.00% Correct Poses
 2.00% Incorrect Poses

Table 6.19. Performance of deterministic recovery in regular polygon

47.40% of frame sequences contain error
 93.15% of erroneous frame sequences recovered pose
 0.00% of frame sequences did not converge
 96.10% Correct Poses
 3.90% Incorrect Poses

Table 6.20. Performance of deterministic recovery along synthetic storefront

62.40% of frame sequences contain error
 91.35% of erroneous frame sequences recovered pose
 0.00% of frame sequences did not converge
 94.60% Correct Poses
 5.40% Incorrect Poses

Table 6.21. Performance of deterministic recovery in a modified Wells Hall

Domain	Error Type	Edges	Distance (ft)	Time (sec)	Accuracy (%)
Engr Bldg	None	5.28	29.76	11.45	100.0
	Modeled	6.84	38.55	14.83	97.8
Wells Hall	None	13.72	69.97	26.91	100.0
	Modeled	37.05	188.96	72.68	93.6
Modified Wells	None	12.61	64.31	24.74	100.0
	Modeled	14.76	75.28	28.95	94.6
Regular Polygon	None	46.00	383.33	147.44	100.0
	Modeled	66.62	555.19	213.54	98.0
Store- front	None	14.28	73.76	28.54	100.0
	Modeled	18.04	92.19	37.88	96.1

Table 6.22. Comparison of domain performance

6.9 Summary

This chapter has presented the simulation results of our StickRep Solution to the Where-am-I Problem. We have seen that in a domain with great symbolic feature ambiguity, StickRep provides a strong representational framework to determine the pose of a robot navigator. We have examined several heuristic schemes to recover from sensing error, but all these heuristics fall short of the performance of our deterministic approach. Metric relaxation of ambiguous feature attributes were matched using the StickRep representation. Even under the influence of realistic noise models, our solution converged to a pose only 30 percent slower than under ideal, error-free sensing conditions. Furthermore, even with significant error, our solution converges to an *accurate* pose in a *short* period of time. We feel that this solution approach is robust, and provides strong performance even on low cost hardware navigators like Rome. No competitive solution to this problem has appeared in the literature.

Capable, but low cost sensors and navigation platforms have opened up the scope of applications for both industrial and research mobile robots. Our solution methodology enables these systems to become more autonomous and to pursue more challenging applications when provided the structure of an indoor environment. StickRep enables less ambitious tasks to be pursued in a structured outdoor environment. Our Where-am-I Solution does not rely on fast or specialized hardware, so it can grow in performance as advances in hardware continue. With such a strong solution in hand, more ambitious navigational tasks can be attempted since the robot possesses the capability to autonomously recovery its pose.

CHAPTER 7

Summary, Conclusions and Recommendations for Future Research

7.1 Summary and Conclusions

The research addressed in this dissertation dealt with robotic navigation and pose recovery using a symbolic landmark map. Sensory information was obtained from an ultrasonic sensing system and a monocular camera to form a vertical landmark model. This model served as a basis for pose estimation, a key challenge for mobile robot applications. Our solution to the Where-am-I Problem enables map-based path planning, obstacle avoidance, landmark identification and other essential navigation operations.

Constructing such a model starts with the complex task of determining range information from the robot's sensing suite. We presented alternative approaches to representing the robot's sensory world, but instead of pursuing a dense, computationally demanding geometric model, we offered a sparse representational scheme that required strong symbolic matching, ambiguity resolution and error rejection

capabilities. Our StickRep representation was built upon the idea of encoding extractable environmental features as attributed edges in a sparse symbolic graph. Our StickRep representation hinges on strong assumptions about the structure of an indoor domain. Visual landmarks in this domain must be sensed, and a real time image processing requirement restricts many approaches. We chose to pursue the extraction of symbolic features in our domain with a single intensity image fused with ultrasonic range data. Many visual features in an indoor environment possess strong verticality. We developed a processing suite that extracted vertical edge ribbons, and classified symbolic landmark candidates using the fused sensory data. These experiments also led to the definition of an ultrasonic sensing accuracy footprint.

Constructing a sensing framework that casts vision and ultrasonic sensing in roles where each can operate accurately, while simultaneously supporting unrestricted real time navigation was our goal. We presented two sensing frameworks, our Two Side and Short Side geometries, that strove towards this goal. These geometries were developed and trained with a fused image/ultrasonic suite of 157 real sensor readings, and over 100 thousand Monte Carlo experiments. Detailed composite error models were determined from lengthy experiments employing these sensing geometries. Both geometries have a more general application, but extensive error analysis and simulations revealed that the Short Side Geometry was best suited for our task. To accurately establish its pose, our navigator must acquire its sensory information at instances where particular sensor error is near its minimum. We defined an accurate navigation envelope that enhances sensing accuracy while only mildly restricting travel and locomotion.

Using the StickRep representation and a test suite of 94 real fused sensor readings, extensive simulations were conducted to examine the performance of our approach in domains containing great feature ambiguity. Metric attributes were often required to successfully disambiguate among, and classify candidate landmarks. We presented

a stepped metric relaxation scheme to enable feature matching to proceed despite significant metric sensing error. From over 80,000 simulated navigation runs we determined that from a random initial and unknown pose, our navigator could recover its pose in less than 12 seconds *in the absense of error* and in less than 15 seconds *with realistic modeled error* in a large domain. Other domains also illustrated strong results, and the introduction of artificial landmarks offer to further improve performance in difficult domains. This time performance was limited not by our solution approach, but by the mechanical limitations of the robot base. Our solution would support implementation in a navigation system traveling at speeds up to 30 feet per second using a Sparc 1 processor. A battery of heuristic based pose recovery experiments were presented, but the overall performance of our deterministic error recovery approach exceeded the former results in all measures of speed and accuracy.

7.2 Contributions

The primary contribution of this research is a **solution to the Where-am-I Problem** in the examined domains. Our solution was examined using 94 real, fuzed test image/ultrasonic readings with over 80,000 simulated navigation trials. An accurate pose was recovered in 97.8 percent of the attempts in a large, varied domain. Accurate and rapid solution performance was also obtained in experiments with additional domains. We observed accurate and rapid pose recovery performance with a navigator constructed with low cost hardware. A Where-am-I Solution enables navigation systems to become more independent and to pursue more difficult applications in a structured, indoor environment.

Our Short Side and Two Side Geometries **measure vertical edge ribbon features in real time with an uncalibrated camera**. These sensing frameworks represent real time solutions to the more general problem of object measuring.

Provided the objects have at least two parallel linear edges, and are viewed with a contrasting background, their primary dimension may be determined without expensive, specialized hardware or a customized measurement jig.

We have presented StickRep, a $1\frac{1}{2}$ dimensional structure that supports rapid symbolic matching with minimum *a priori* information. StickRep sparsely represents the 3D world. This representation supports fusing ultrasonic range data to images of symbolic features from the mapped domain. StickRep successfully handled inexact feature attributes in ambiguous feature domains throughout our extensive simulations. This representation reduced feature matching to an $O(n)$ string matching problem, and supports dynamic feature attributes. StickRep avoids massive domain databases and the limiting assumption of a static environment.

We presented a real time vertical edge ribbon detector using available illumination. This detector was robust to rotational misalignment of the camera, and correctly extracted and classified almost 84 percent of the landmark candidates from a suite of 94 real test images. This detector only required inexpensive sensing hardware and computational support with processing speeds comparable to a Sparc 1.

Another contribution is the definition of an accurate ultrasonic sensing footprint. Forty thousand ultrasonic samples were obtained from various ranging surfaces and orientations. Definite limitations were observed that motivate acquisition strategies to ensure accurate ranging. Poor angular resolution has plagued ultrasonic sensing applications in the literature, but no sensing footprint has been established. Such a footprint led to the design of an overall navigation envelope that protects methodologies from frequent spurious ranging data.

A small contribution was made towards the physical construction of two robotic navigators to serve as testbeds for future experiments. They can serve as testbeds for theoretical work and simulations, and stimulate further studies.

Rome and Sparta represent mature hardware platforms well prepared for future sensory and navigation experiments. Future designers will be relieved of a majority of the hardware concerns and instabilities inherent in robotic applications. Rome now possesses a stable power backplane, ready for evaluation and experimentation of systems such as transputers, or a Sparc 1 system box. Rome forms the initial backbone for more mature work, pursuing more ambitious tasks.

7.3 Recommendations for Future Research

We presented a large simulation study of our Where-am-I Solution in Chapter 6. We examined two real and three synthetic domains. While these domains are large, and contain varied, and ambiguous features, expanding the simulation domains to include other buildings could be illuminating. It would be desirable if generalizations could be made linking navigation capabilities with building types, but the existence of sensing error will precipitate observations that are probabilistic in character. No performance guarantees can be made. We might expect pose recovery to be more rapid in some smaller buildings, since they would require less matching attempts than in our domains. But this presumes that feature ambiguity would not be great.

Successfully extending our solution into limited outdoor domains would also be very useful. As we discussed in Section 2.3.2, many efforts are pursuing structured outdoor applications, and adapting StickRep's sequential feature representation to support more general image viewpoints would greatly empower autonomous navigation. Such an effort would be more general than past outdoor navigators at Michigan State [55], and at many other research centers. Unlike most published approaches, our effort is limited, not enabled, by the hardware on which it is implemented. Performance certainly may be degraded by an outdoor generalization, but strong performance

potential exists.

An enhancement to an indoor navigator would be using two cameras with opposing optical axes to search *both* hallway walls for features. Our matching routine could be tasked to match two separate landmark candidates and then search for a domain location where these symbolic sequences occur on opposing walls. StickRep must be carefully modified to encode the domain relationships to include an opposing feature edge. Feature matching would resemble two dimensional string matching. Although this expansion of domain context may greatly decrease pose recovery times, it might rather become computationally intense due to the more complex edge adjacencies that may exist at a feature node. This approach could prove useful, and certainly warrants exploring.

We presented our vertical edge ribbon detector in Section 4.1. While this detector was fast, and tolerant to roll axis misalignment of the camera, it produced false positives on blank walls. A better detection procedure to eliminate these false positives represents a good opportunity for increased performance. Several attempts were made to implement more adaptive thresholding, but the crisp, uniform intensity histograms typical of a man made environment caused bimodal Gaussian fitting to be problematic and poor in performance. Other adaptive thresholding techniques should be explored. A successful adaptive thresholding technique that operated under natural, and varied illumination, would enhance the robustness of our detector.

APPENDICES

APPENDIX A

Ultrasound Histograms

The following histograms are summarized in Tables 4.2 and 4.3 in Section 4.3. 1000 ultrasonic samples were obtained from each of three surface types at three or four orientations each. The indicated angle is the angle between the acoustic axis and the surface normal. The choice of angles was arbitrary, and the following histograms are representative of the results observed.

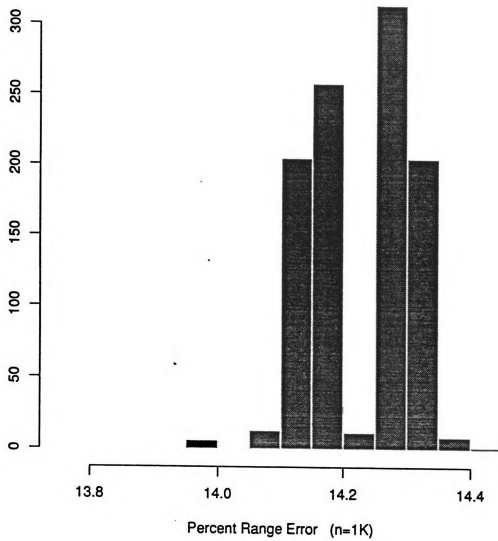


Figure A.1. Concrete surface, 45°

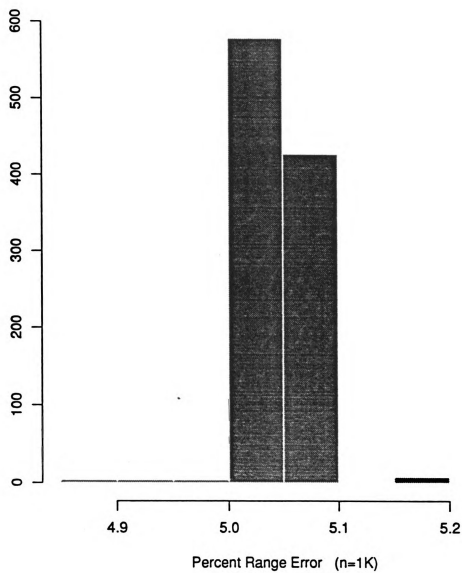


Figure A.2. Concrete surface, 30°

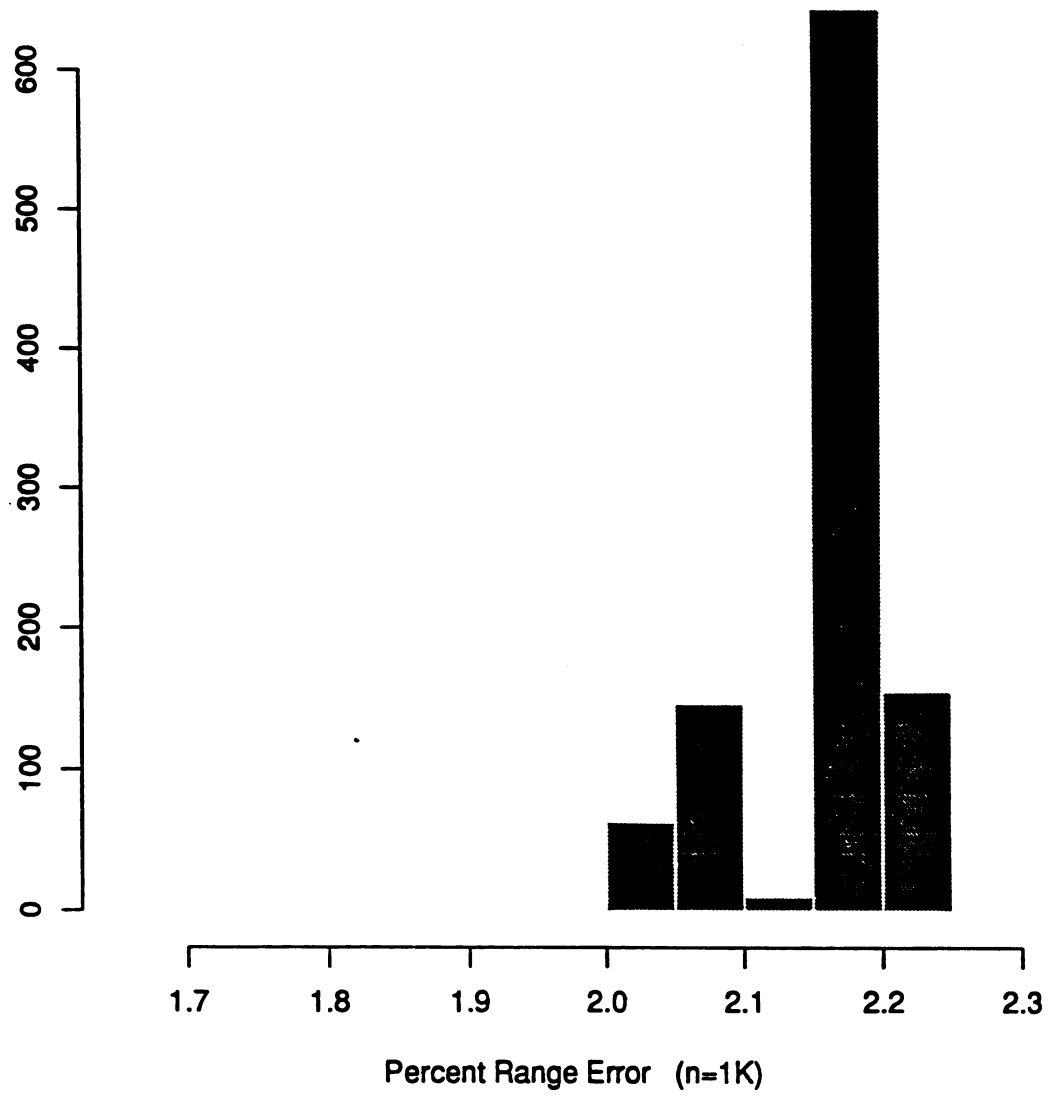


Figure A.3. Concrete surface, 15°

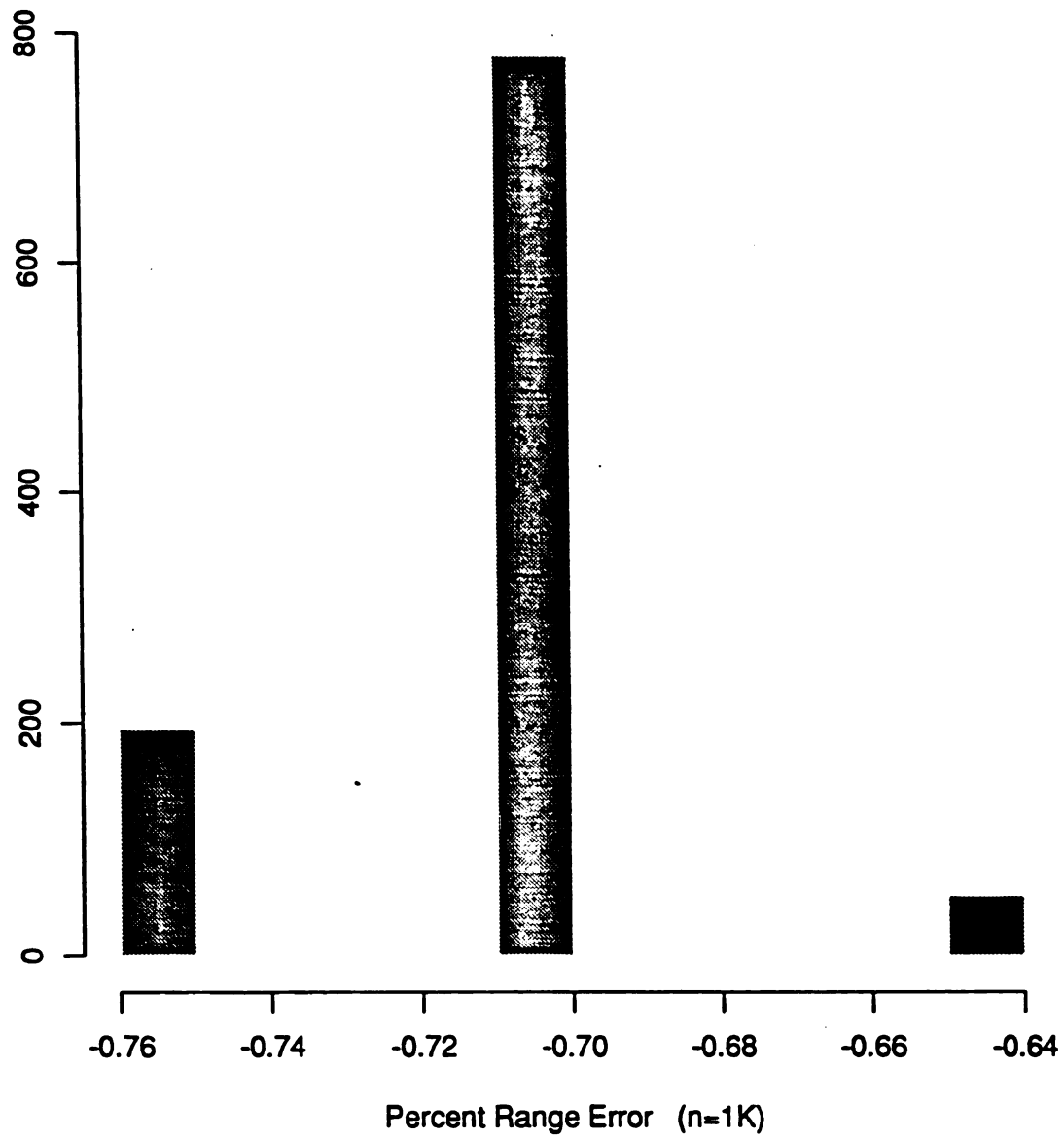


Figure A.4. Concrete surface, 0°

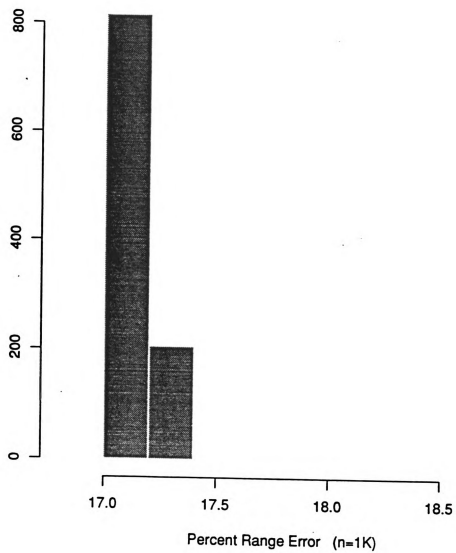


Figure A.5. Wood surface, 45°

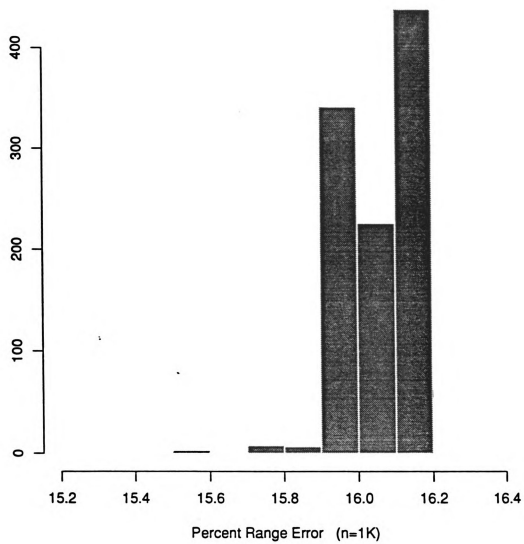


Figure A.6. Wood surface, 30°

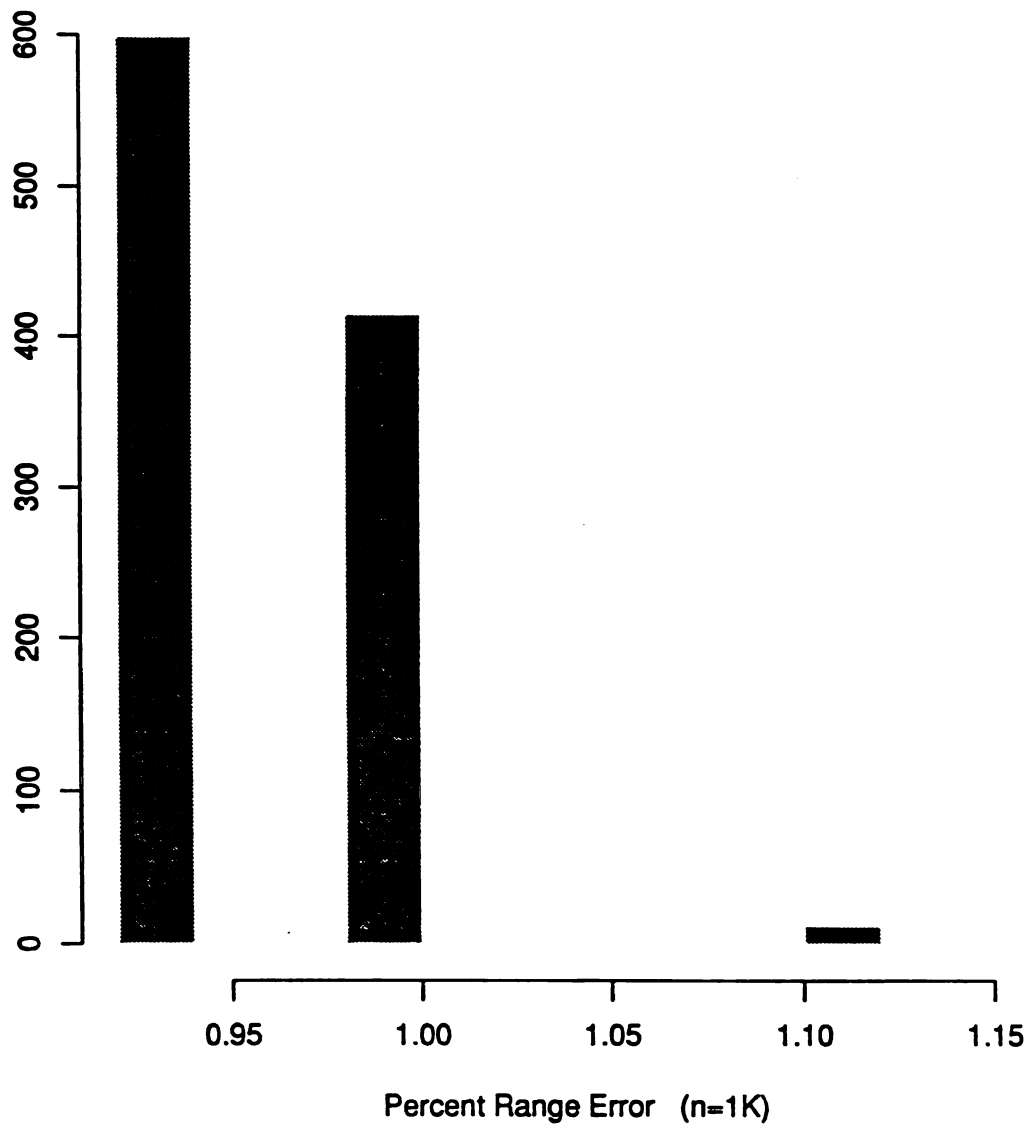


Figure A.7. Wood surface, 15°

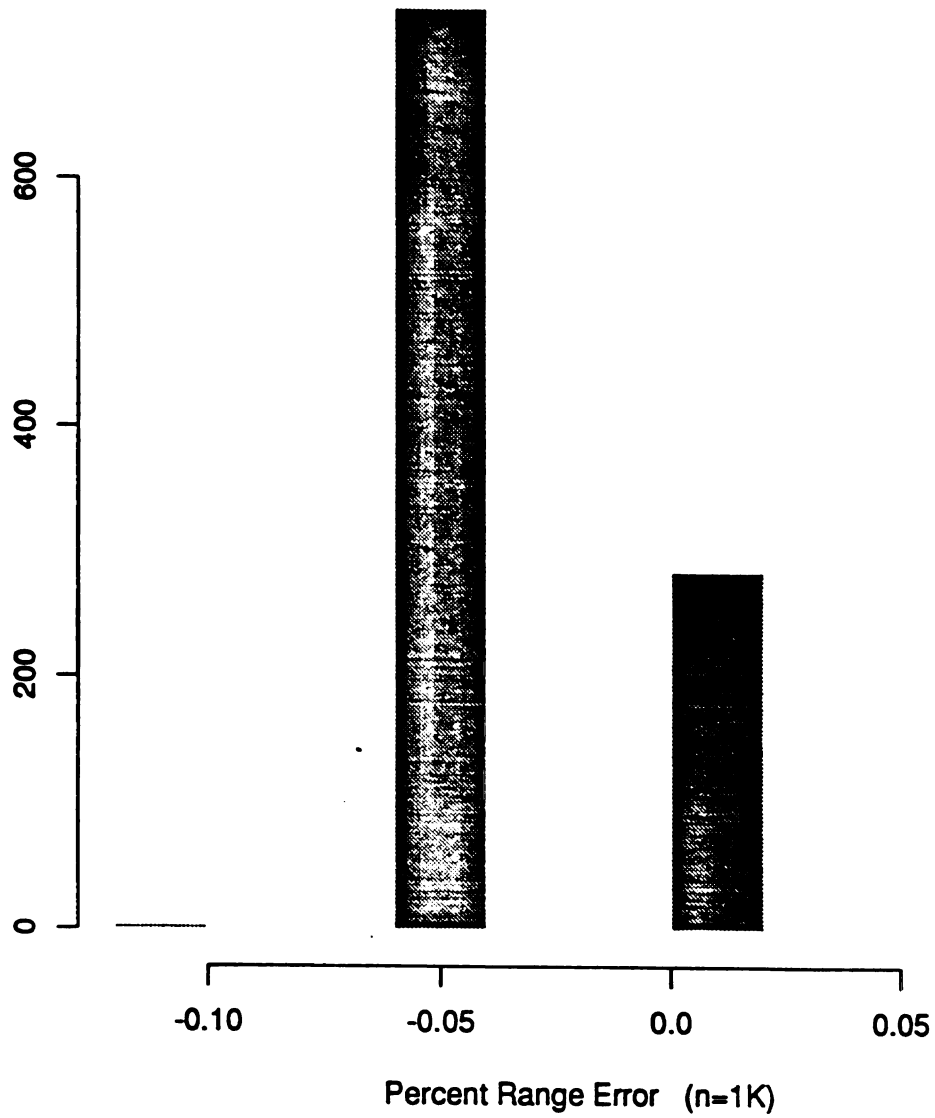


Figure A.8. Wood surface, 0°

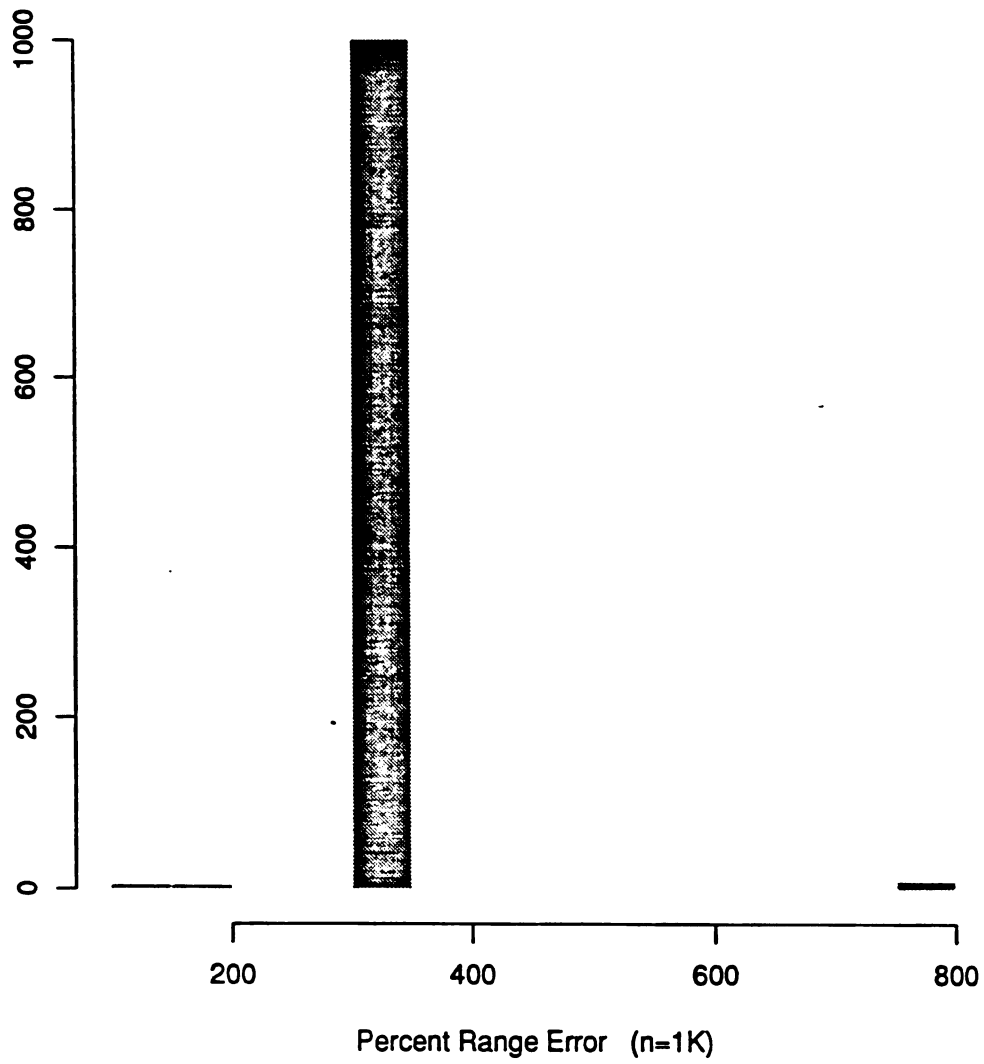


Figure A.9. Cloth surface, 30°

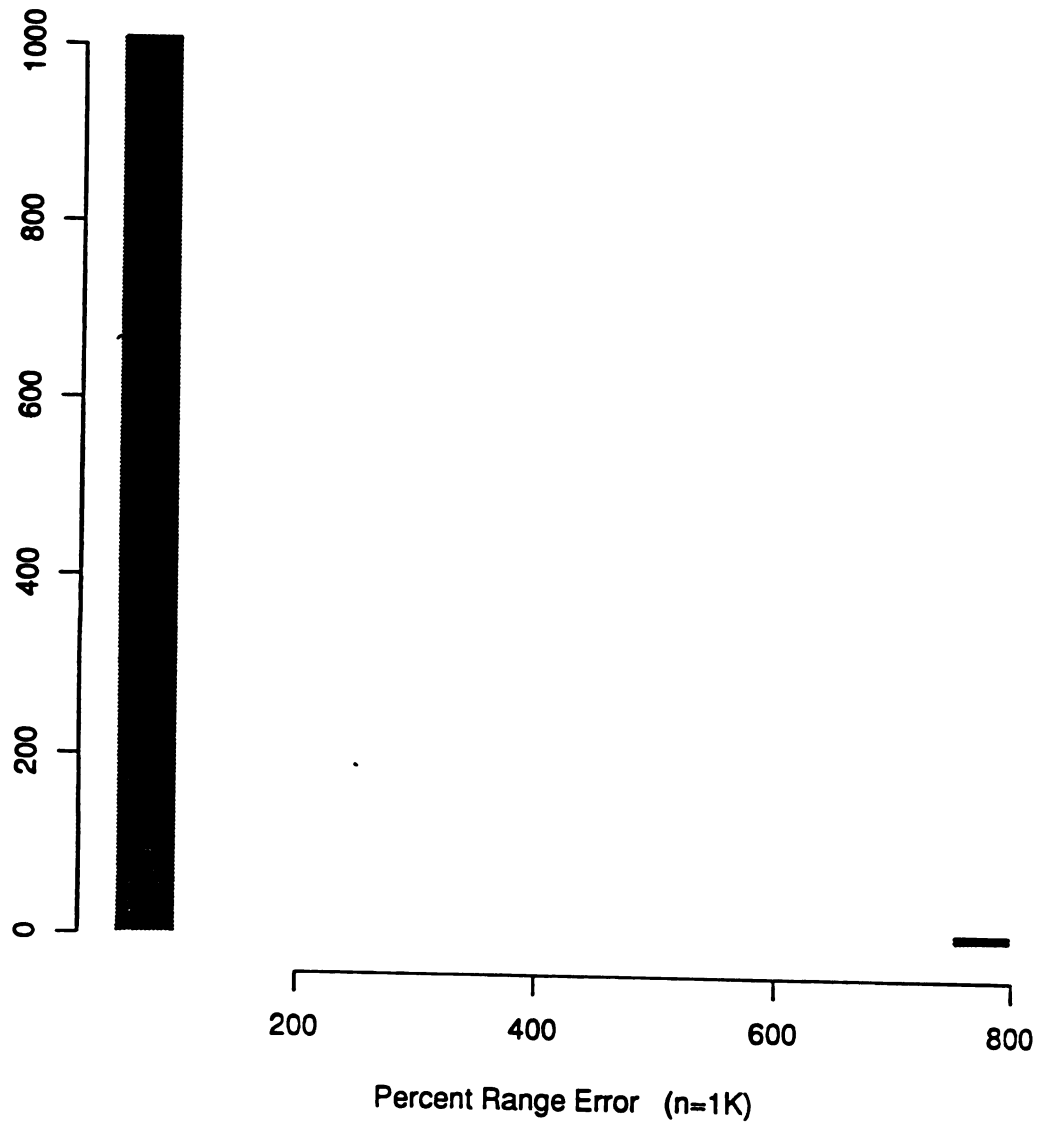


Figure A.10. Cloth surface, 15°

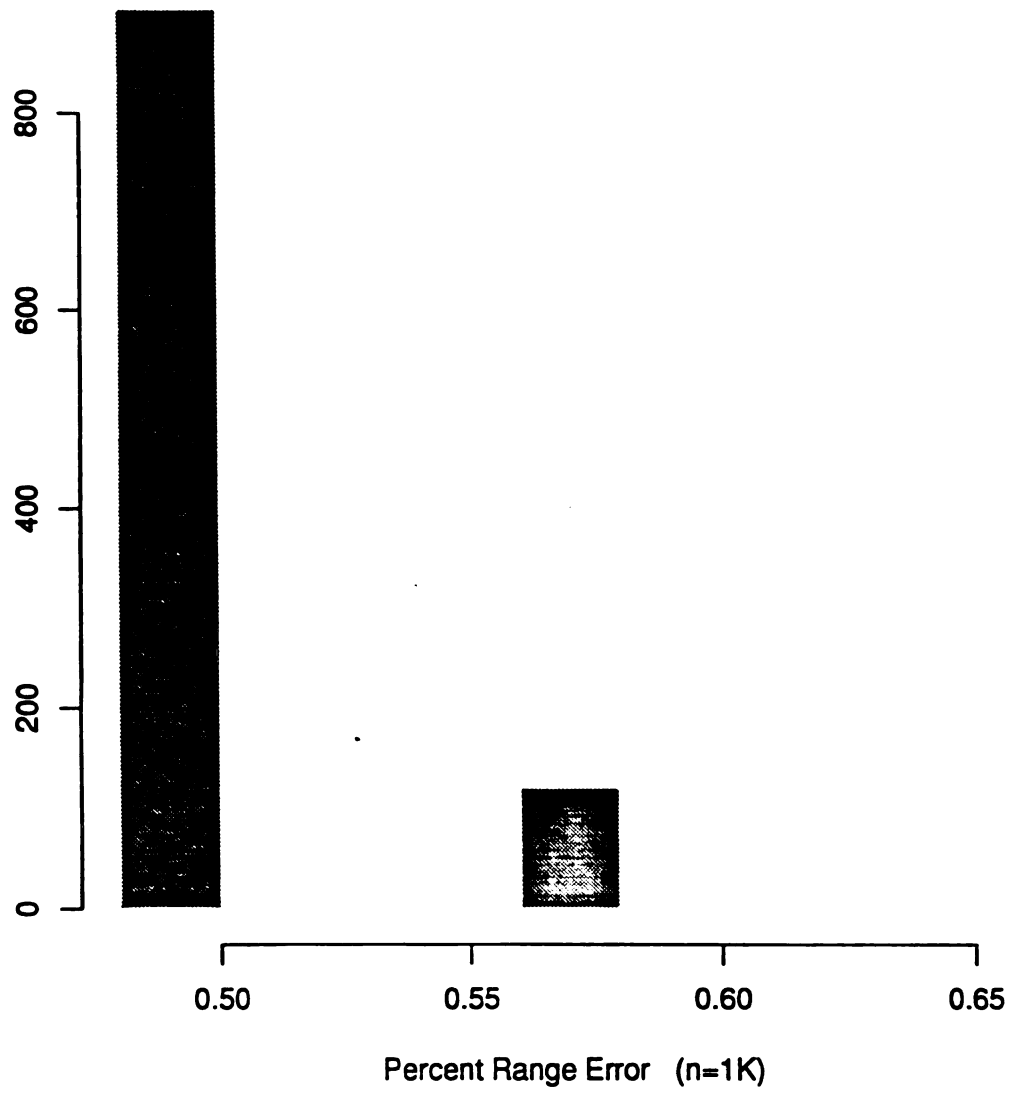


Figure A.11. Cloth surface, 0°

APPENDIX B

Partial Derivatives from Sensing Geometries

In Section 5.5.3 we presented the first order linear error model for our landmark measuring geometries. The closed form representation of these partial derivatives surpassed four pages in length each ! We present the following figures to summarize the behavior of the partial derivatives for each of the two sensing geometries. Note that as the arbitrary look angle increases, the nonlinear gain portion of the curve approaches the linear region of the curve that we are operating on.

Two Side Geometry

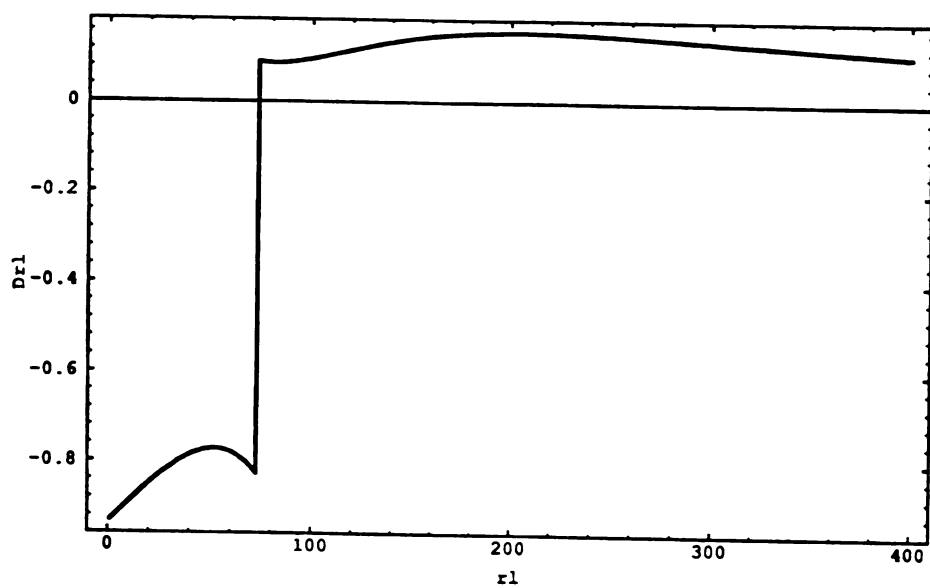


Figure B.1. $\frac{\partial}{\partial RL}$, $\phi = 0^\circ$

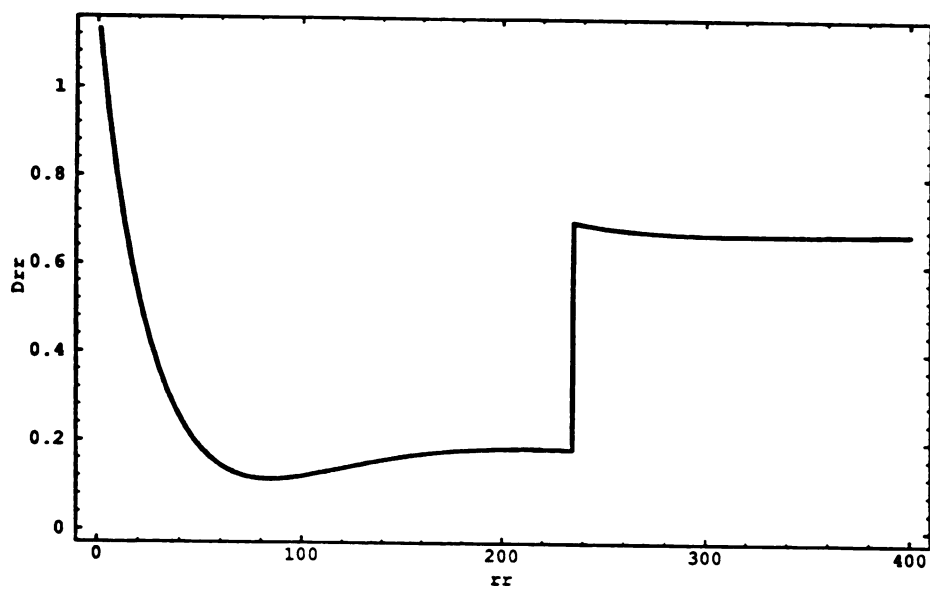


Figure B.2. $\frac{\partial}{\partial RR}$, $\phi = 0^\circ$

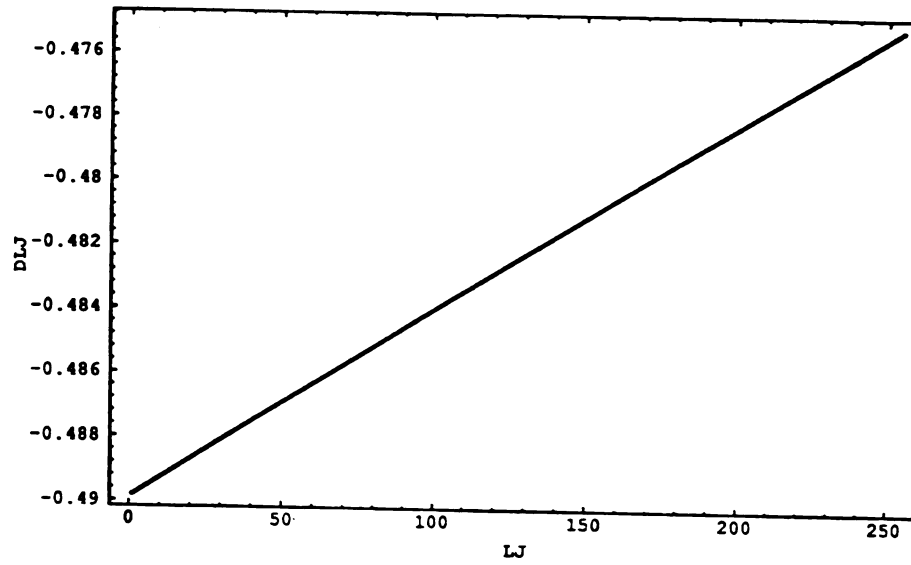


Figure B.3. $\frac{\partial}{\partial L_J}$, $\phi = 0^\circ$

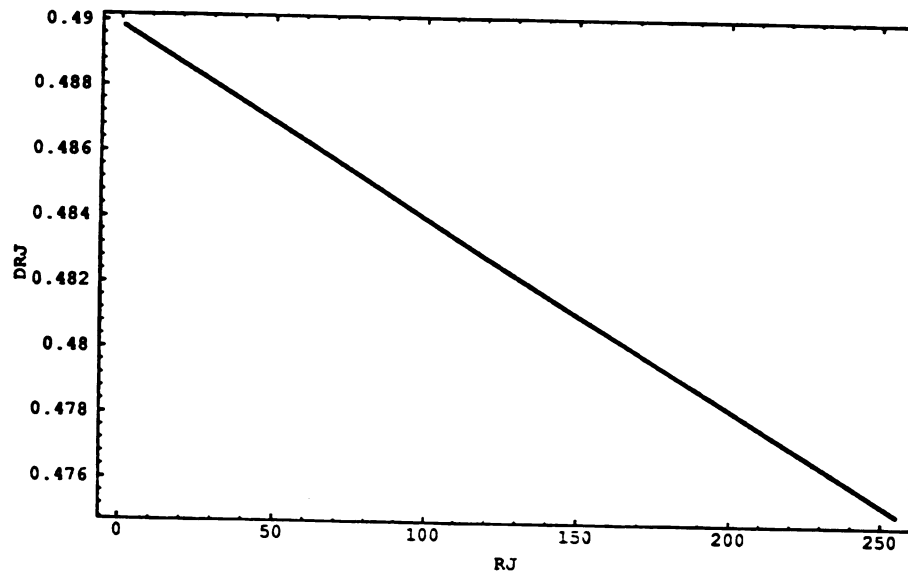


Figure B.4. $\frac{\partial}{\partial R_J}$, $\phi = 0^\circ$

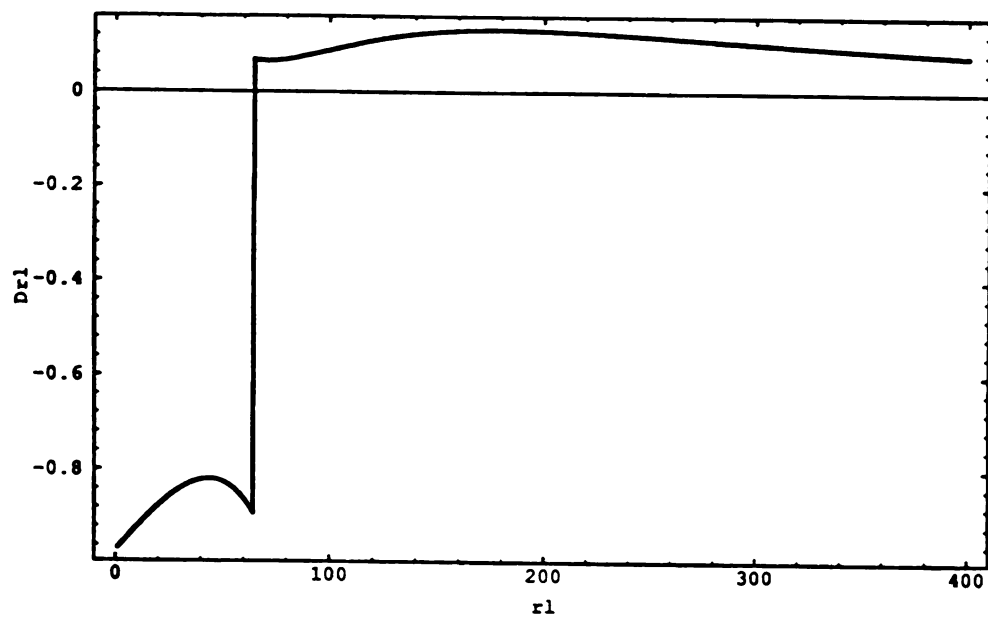


Figure B.5. $\frac{\partial}{\partial RL}$, $\phi = 20^\circ$

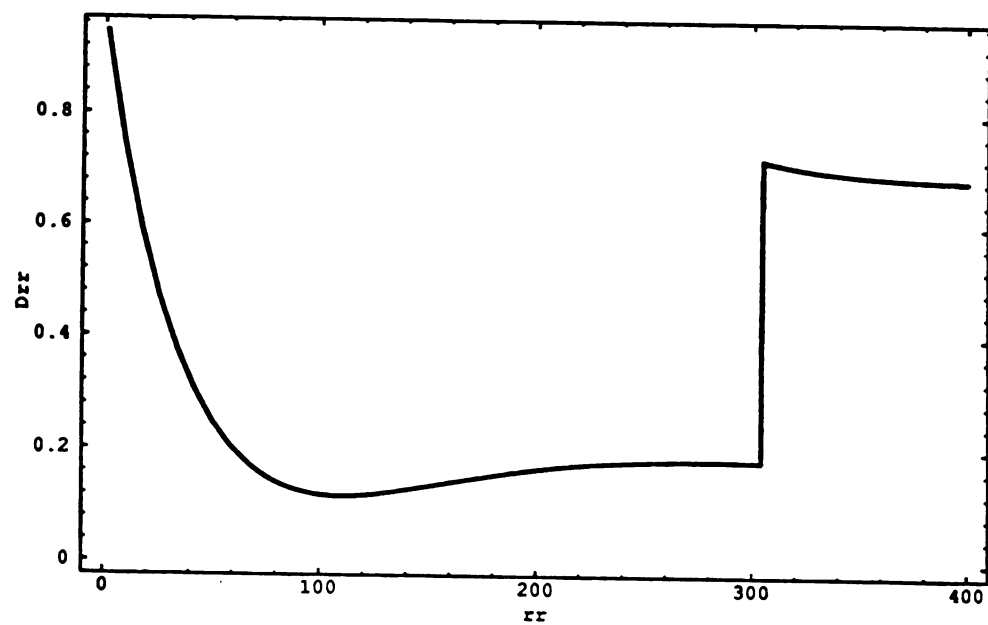


Figure B.6. $\frac{\partial}{\partial RR}$, $\phi = 20^\circ$

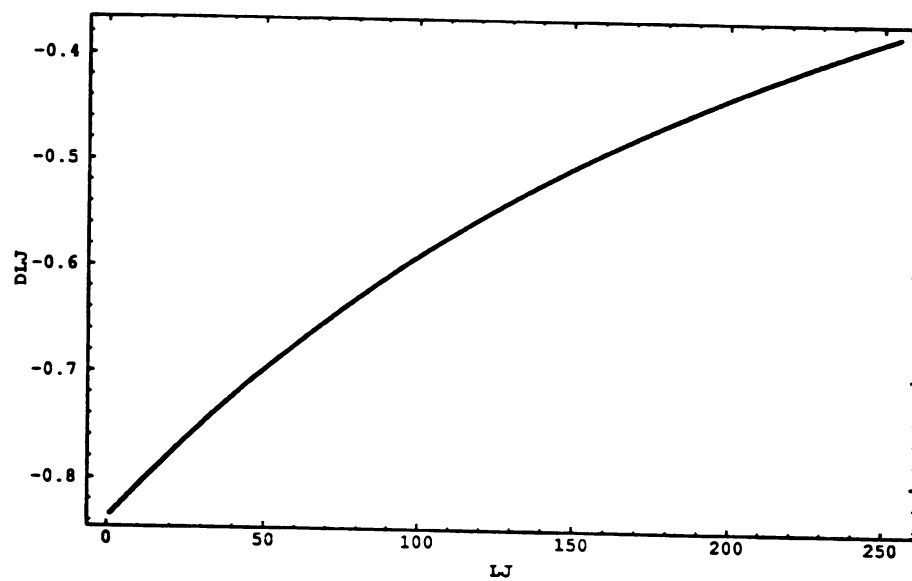


Figure B.7. $\frac{\partial}{\partial LJ}$, $\phi = 20^\circ$

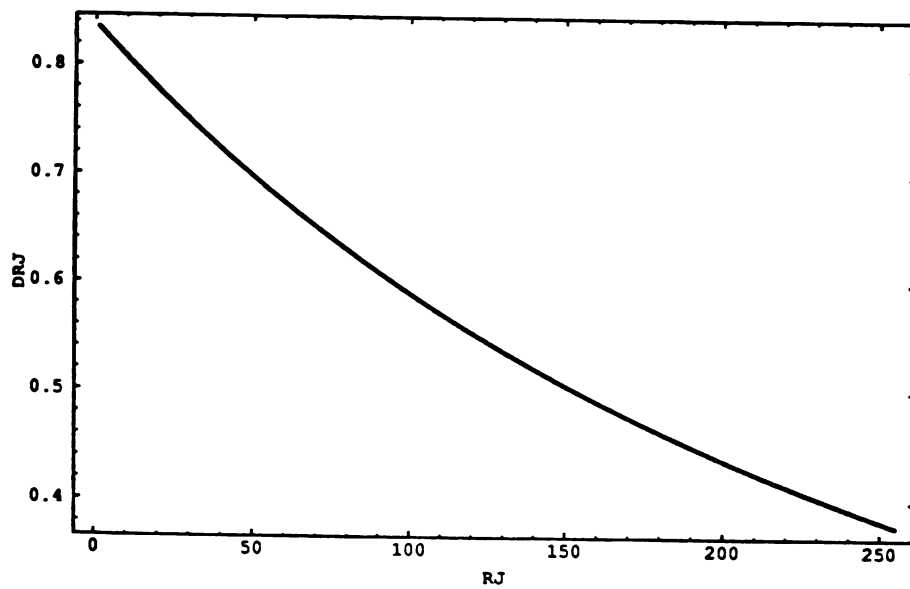


Figure B.8. $\frac{\partial}{\partial RJ}$, $\phi = 20^\circ$

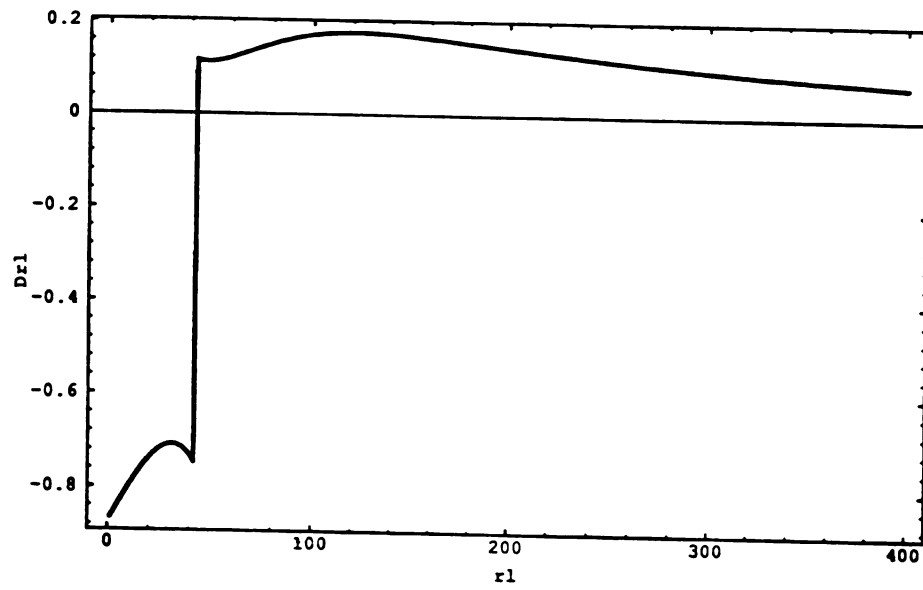


Figure B.9. $\frac{\partial}{\partial RL}$, $\phi = 40^\circ$

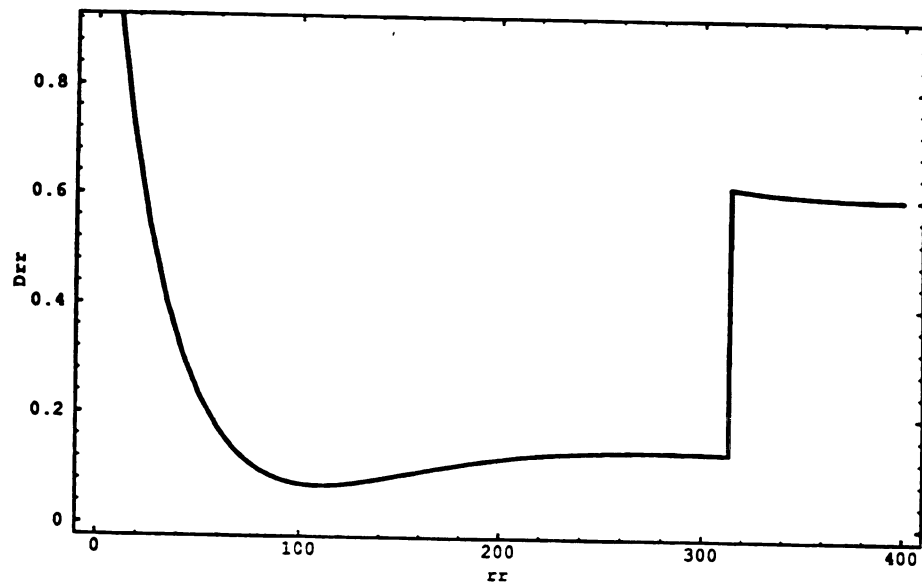


Figure B.10. $\frac{\partial}{\partial RR}$, $\phi = 40^\circ$

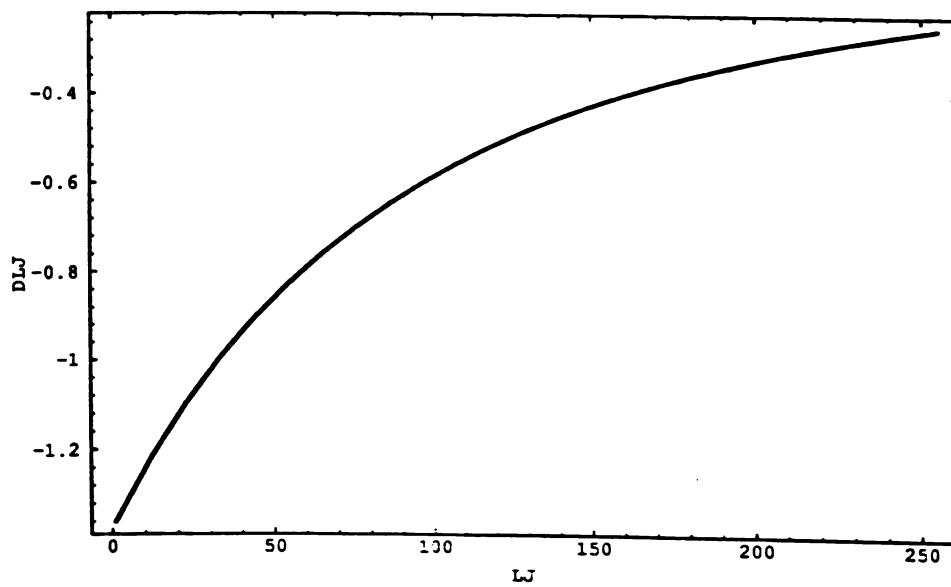


Figure B.11. $\frac{\partial}{\partial LJ}$, $\phi = 40^\circ$

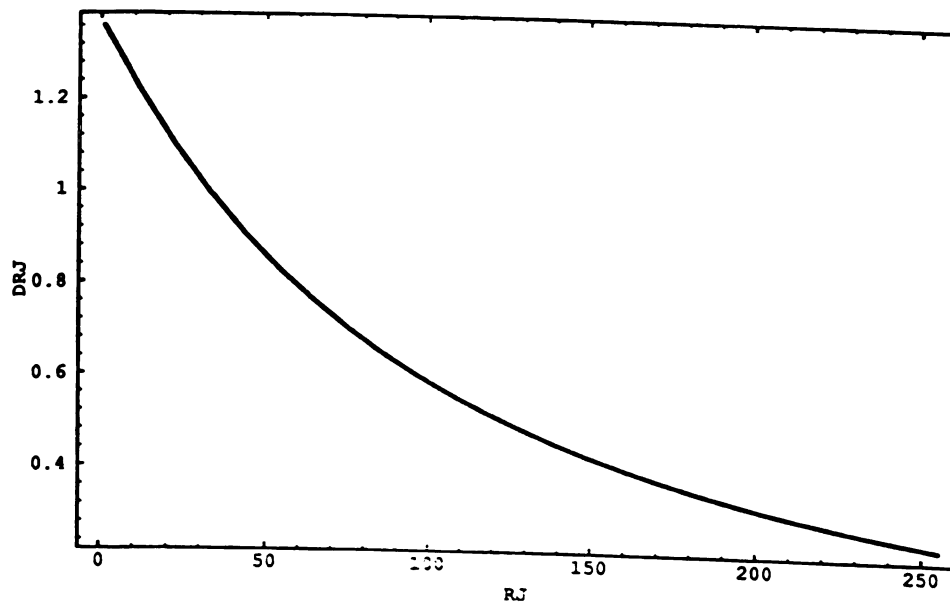


Figure B.12. $\frac{\partial}{\partial RJ}$, $\phi = 40^\circ$

Short Side Geometry

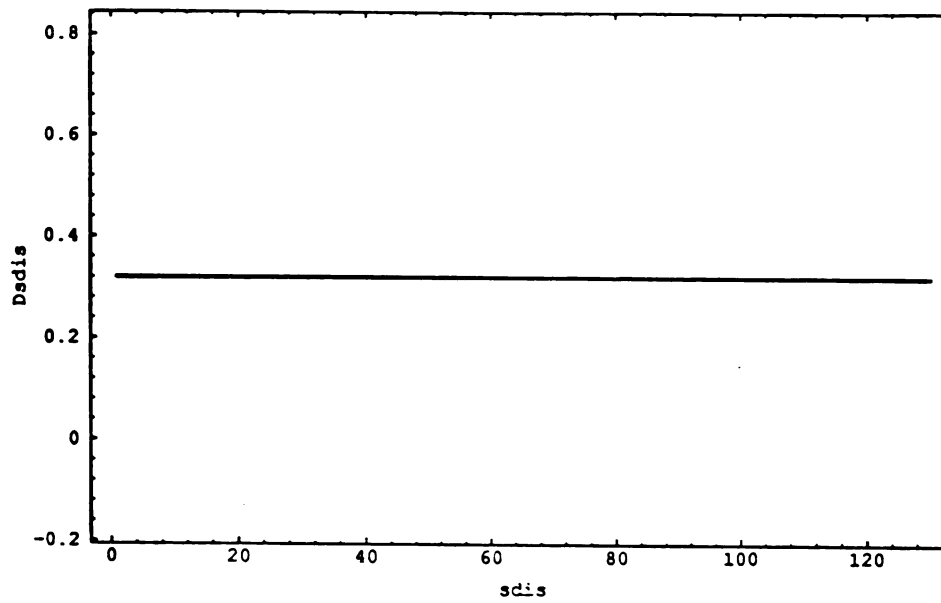


Figure B.13. $\frac{\partial}{\partial s_{dis}}$, $\phi = 0^\circ$

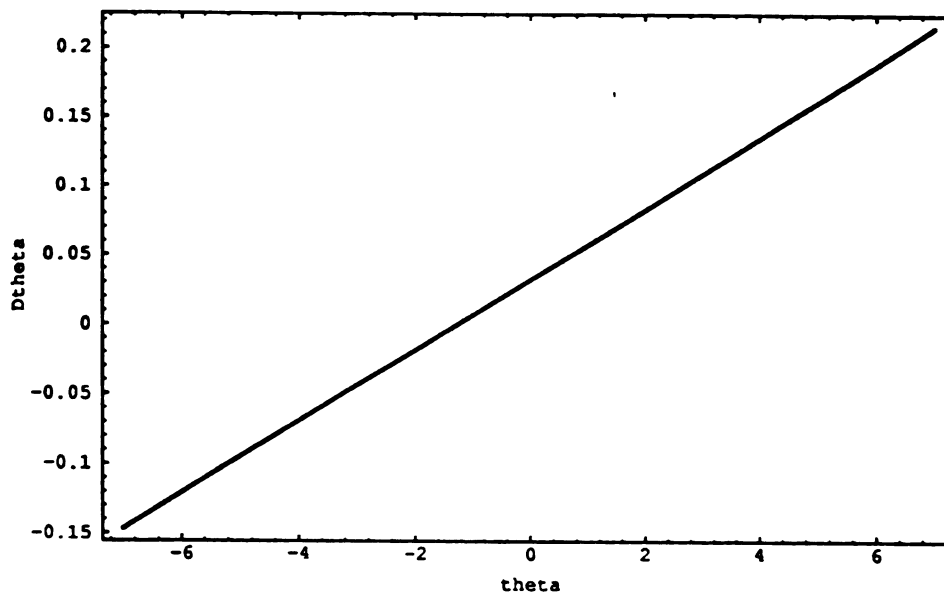


Figure B.14. $\frac{\partial}{\partial \theta}$, $\phi = 0^\circ$

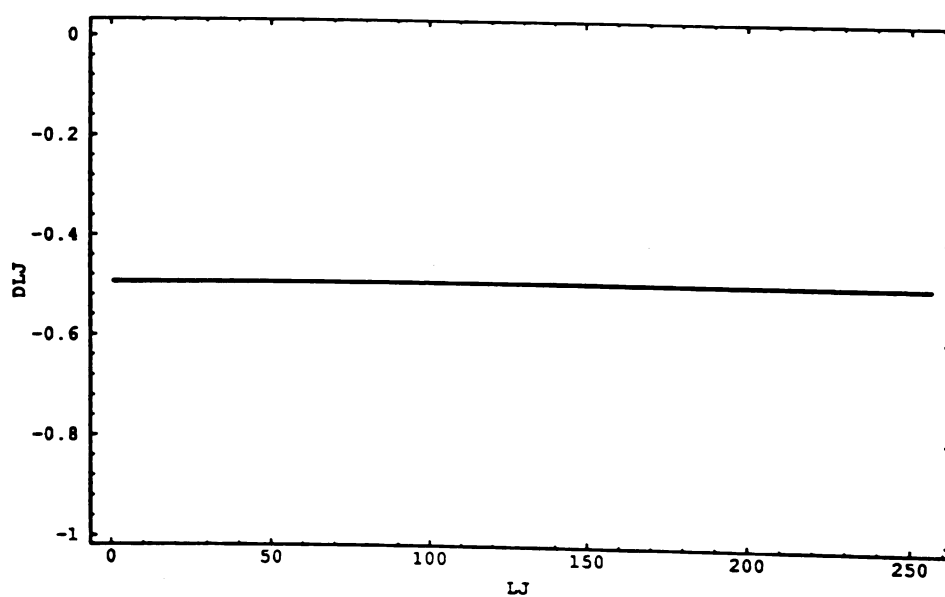


Figure B.15. $\frac{\partial}{\partial LJ}$, $\phi = 0^\circ$

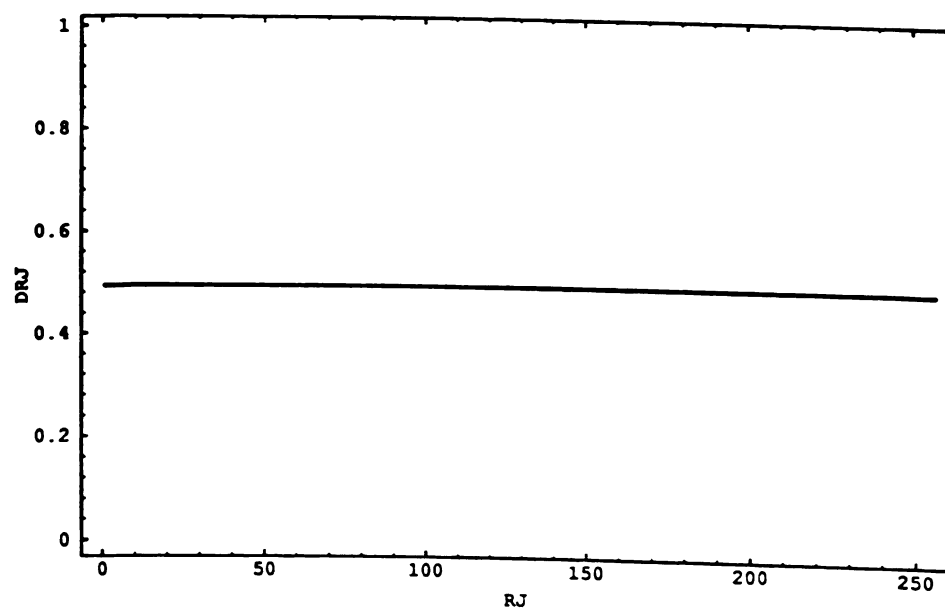


Figure B.16. $\frac{\partial}{\partial RJ}$, $\phi = 0^\circ$

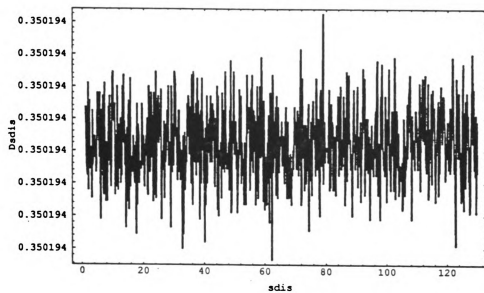


Figure B.17. $\frac{\partial \theta}{\partial r}$, $\phi = 20^\circ$

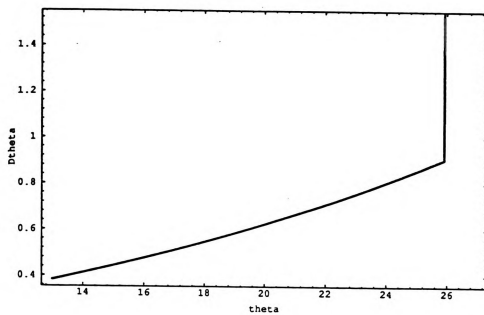


Figure B.18. $\frac{\partial \theta}{\partial \theta}$, $\phi = 20^\circ$

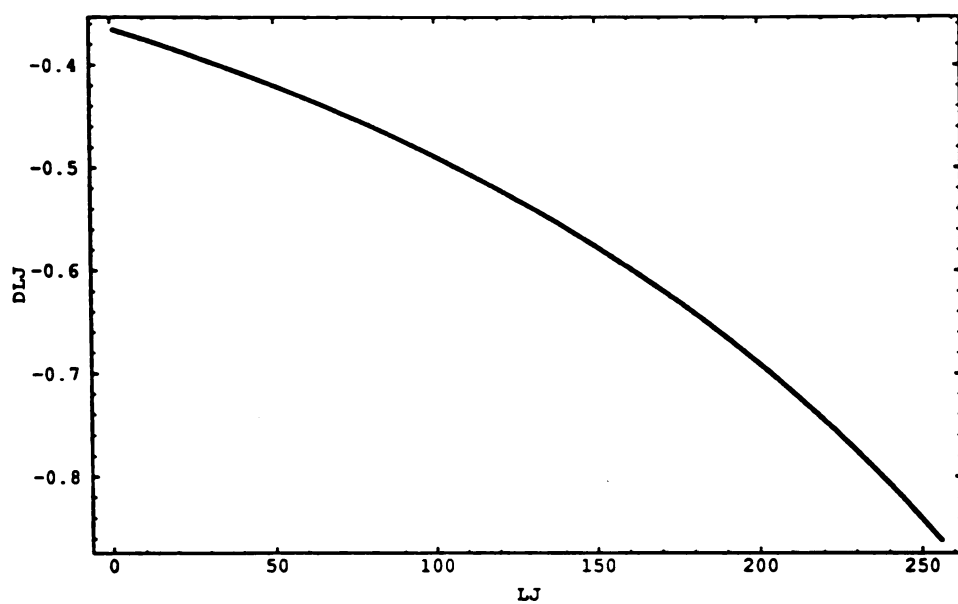


Figure B.19. $\frac{\partial}{\partial LJ}$, $\phi = 20^\circ$

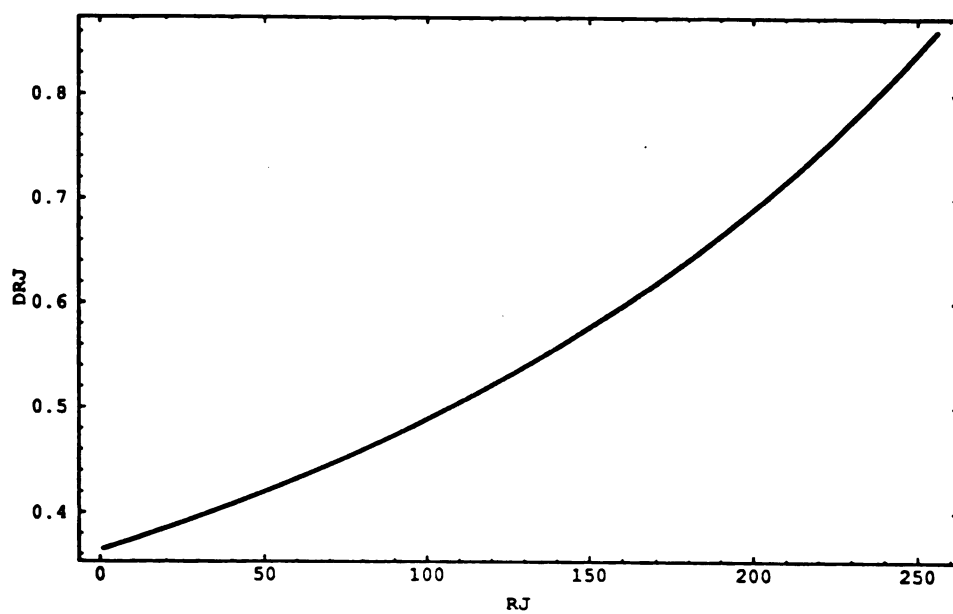
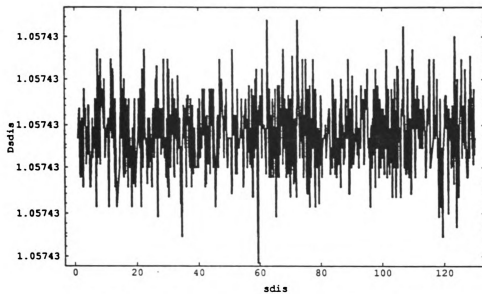
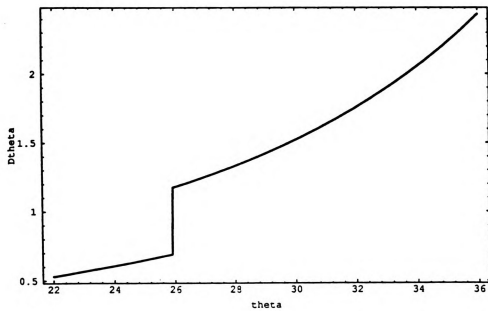


Figure B.20. $\frac{\partial}{\partial RJ}$, $\phi = 20^\circ$

Figure B.21. $\frac{\partial}{\partial sdis}$, $\phi = 40^\circ$ Figure B.22. $\frac{\partial}{\partial \theta}$, $\phi = 40^\circ$

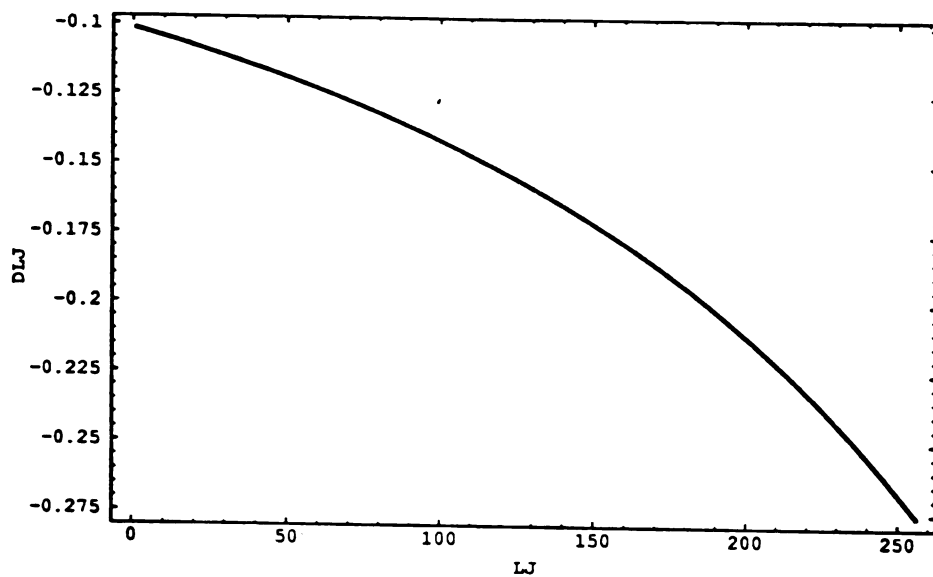


Figure B.23. $\frac{\partial}{\partial LJ}$, $\phi = 40^\circ$

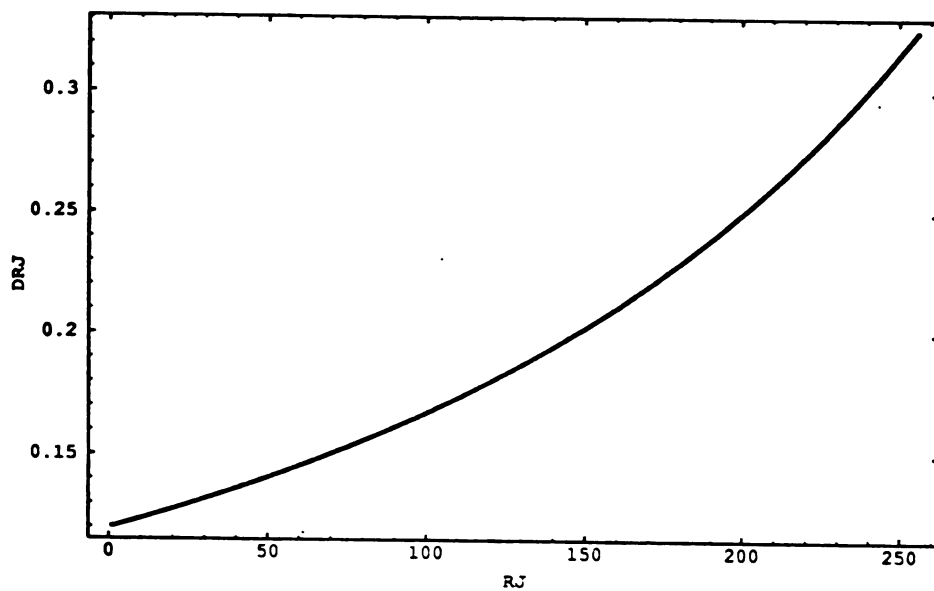


Figure B.24. $\frac{\partial}{\partial RJ}$, $\phi = 40^\circ$

BIBLIOGRAPHY

BIBLIOGRAPHY

- [1] M. Abidi and T. Chandra. Pose estimation for camera calibration and landmark tracking. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.
- [2] N. Alvertos and D. Brzakovic. Camera geometries for image matching in 3-d machine vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, September 1989.
- [3] R. Arkin and R. Murphy. Autonomous navigation in a manufacturing environment. *IEEE Transactions on Robotics and Automation*, August 1990.
- [4] J. E. Arnold. Experiences with the subsumption architecture. *Proceedings IEEE International Conference on Artificial Intelligence Applications*, 1989.
- [5] M. Asada. Building 3-d world model for a mobile robot from sensory data. *Proceedings IEEE International Conference on Robotics and Automation*, 1988.
- [6] M. Asada. Map building for a mobile robot from sensory data. *IEEE Transactions on Systems, Man, and Cybernetics*, November 1990.
- [7] M. Asada et al. Representing global world of a mobile robot with relational local maps. *IEEE Transactions on Systems, Man, and Cybernetics*, November 1990.
- [8] D. Barnea and H. Silverman. A class of algorithms for fast digital image registration. *IEEE Transactions on Computers*, February 1972.
- [9] M. Beckerman and E. Oblow. Treatment of systematic errors in the processing of wide-angle sonar sensor data for robotics navigation. *IEEE Transactions on Robotics and Automation*, April 1990.
- [10] B. Bhanu. CAD-based robot vision. *IEEE Computer*, August 1987.

- [11] J. Bialasiewicz. Sufficient and epsilon-sufficient statistics in pattern recognition and their relation to fuzzy techniques. *IEEE Transactions on Systems, Man, and Cybernetics*, September 1989.
- [12] F. Blais et al. Optical range image acquisition for the navigation of a mobile robot. *Proceedings IEEE International Conference on Robotics and Automation*, 1991.
- [13] R. Bolles and A. Bobick. Exploiting temporal coherence in scene analysis for autonomous navigation. *Proceedings IEEE International Conference on Robotics and Automation*, 1989.
- [14] J. Borenstein and Y. Koren. Histogramic in-motion mapping for mobile robot obstacle avoidance. *IEEE Transactions on Robotics and Automation*, August 1991.
- [15] J. Borenstein and Y. Koren. The vector field histogram - fast obstacle avoidance for mobile robots. *IEEE Transactions on Robotics and Automation*, June 1991.
- [16] K. Boyer and A. Kak. Robotic manipulation experiments using structural stereopsis for 3d vision. *IEEE Expert*, Fall 1986.
- [17] K. Boyer and A. Kak. Structural stereopsis for 3d vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, March 1988.
- [18] O. Bozma and R. Kuc. Building a sonar map in a specular environment using a single mobile sensor. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, December 1991.
- [19] R. Brockett. On the computer control of movement. *Proceedings IEEE International Conference on Robotics and Automation*, 1988.
- [20] R. Brooks. A robust layered control system for a mobile robot. *IEEE Transactions on Robotics and Automation*, March 1986.
- [21] R. Brooks. Visual map making for a mobile robot. In *Readings in Computer Vision*. Morgan Kaufmann, 1987.
- [22] R. Brooks. New approaches to robotics. *Science*, September 1991.
- [23] R. Brooks and A. Flynn. Self calibration of motion and stereo vision for mobile robot navigation. Technical report, Massachusetts Institute of Technology Artificial Intelligence Laboratory, 1987.

- [24] M. Brown. Locating object surfaces with an ultrasonic range sensor. *Proceedings IEEE International Conference on Robotics and Automation*, 1985.
- [25] D. Brzakovic and L. Hong. Road edge detection for mobile robot navigation. *Proceedings IEEE International Conference on Robotics and Automation*, 1989.
- [26] A. Califano et al. Data and model driven foveation. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.
- [27] H. W. Calkins and M. D. F. The transition to automated production cartography: Design of the master cartographic database. *The American Cartographer*, February 1987.
- [28] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, November 1986.
- [29] A. Chattergy. Some heuristics for navigation of a robot. Technical report, Robotics Research Lab University of Hawaii, 1984.
- [30] S. Chen and W. Tsai. Determination of robot locations by common object shapes. *IEEE Transactions on Robotics and Automation*, February 1991.
- [31] C. K. Chow and K. T. Automatic boundary detection of the left ventricle from cineangiograms. *Computers and Biomedical Research*, May 1972.
- [32] J. Clark and N. Ferrier. Control of visual attention in mobile robots. *Proceedings IEEE International Conference on Robotics and Automation*, 1989.
- [33] D. Coombs and C. Brown. Intelligent gaze control in binocular vision. *Proceedings IEEE International Symposium on Intelligent Control*, 1990.
- [34] D. Coombs et al. Gaze control and segmentation. *Proceedings AAAI Qualitative Vision Workshop*, July 1990.
- [35] I. Cox. Blanche - an experiment in guidance and navigation of an autonomous robot vehicle. *IEEE Transactions on Robotics and Automation*, April 1991.
- [36] J. Crowley. Position estimation for an intelligent mobile robot. Technical report, Carnegie-Mellon University, 1983.
- [37] J. Crowley. Dynamic world modeling for an intelligent mobile robot using a rotating ultrasonic ranging device. *Proceedings IEEE International Conference on Robotics and Automation*, 1985.

- [38] J. Crowley. World modeling and position estimation for a mobile robot using ultrasonic ranging. *Proceedings IEEE International Conference on Robotics and Automation*, 1989.
- [39] M. Daily et al. Autonomous cross-country navigation with the ALV. *Proceedings IEEE International Conference on Robotics and Automation*, 1988.
- [40] L. Davis et al. RAMBO - vision and planning on the connection machine. *DARPA Image Understanding Workshop*, 1989.
- [41] L. Davis and T. Kushner. Vision-based navigation: A status report. *DARPA Image Understanding Workshop*, 1987.
- [42] D. DeMenthon and L. Davis. New exact and approximate solutions of the three-point perspective problem. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.
- [43] M. Dhome and M. Richetin. Determination of the attitude of 3-d objects from a single perspective view. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, December 1989.
- [44] E. Dickmanns and A. Zapp. A curvature-based scheme for improving road vehicle guidance by computer vision. *Proceedings SPIE Conference on Mobile Robots*, 1986.
- [45] M. Drumheller. Mobile robot localization using sonar. Technical report, Artificial Intelligence Lab Massachusetts Institute of Technology, 1985.
- [46] G. Dudek and M. Jenkin. Robotic exploration as graph construction. *IEEE Transactions on Robotics and Automation*, December 1991.
- [47] H. Dulimarta. Personal communication. July, 1992.
- [48] R. Dunley. Obstacle avoidance perception processing for the autonomous land vehicle. *Proceedings IEEE International Conference on Robotics and Automation*, 1988.
- [49] C. Durieu et al. Localization of a mobile robot with beacons taking erroneous data into account. *Proceedings IEEE International Conference on Robotics and Automation*, 1989.
- [50] A. Elfes. Sonar-based real-world mapping and navigation. *IEEE Transactions on Robotics and Automation*, June 1987.

- [51] A. Elfes. *Occupancy Grids: A Probabilistic Framework for Robot Perception and Navigation*. PhD thesis, Carnegie-Mellon University, 1989.
- [52] A. Elfes. Using occupancy grid for robot perception and navigation. *IEEE Computer*, June 1989.
- [53] A. Elfes. Occupancy grids: A stochastic spatial representation for active robot perception. *Proceedings of the 6th Conference on Uncertainty in AI*, July 1990.
- [54] A. Elfes. Dynamic control of robot perception using stochastic spatial models. *Proceedings International Workshop on Information Processing in Mobile Robots*, March 1991.
- [55] B. Fehr and J. Gerrish. Vision-guided off-road vehicle. *ASAE International Meeting*, December 1989.
- [56] J. Fei and C. Isik. Adapting to environmental variations in a rule-based mobile robot controller. *Proceedings IEEE International Conference on Systems, Man, and Cybernetics*, 1989.
- [57] C. Fennema and A. Hanson. Experiments in autonomous navigation. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.
- [58] C. Fennema and A. Hanson. Model-directed mobile robot navigation. *IEEE Transactions on Systems, Man, and Cybernetics*, November 1990.
- [59] M. Fischler and R. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, June 1981.
- [60] K. Fok and M. Kabuka. An automatic navigation system for vision guided vehicles using a double heuristic and finite state machine. *IEEE Transactions on Robotics and Automation*, February 1991.
- [61] H. Frohn and W. Seelen. Visocar: An autonomous industrial transport vehicle guided by visual navigation. *Proceedings IEEE International Conference on Robotics and Automation*, 1989.
- [62] A. Fukunaga. *Statistical Pattern Recognition*. Academic Press, 1990.
- [63] G. Giralt et al. An integrated navigation and motion control system for autonomous multisensory mobile robots. *Proceedings International Symposium on Robotics Research*, 1984.

- [64] P. Grandjean. 3-D modeling of indoor scenes by fusion of noisy range and stereo data. *Proceedings IEEE International Conference on Robotics and Automation*, 1989.
- [65] W. Grimson. Sensing strategies for disambiguating among multiple objects in known poses. *IEEE Transactions on Robotics and Automation*, December 1986.
- [66] N. Griswold and J. Eem. Control for mobile robots in the presence of moving objects. *IEEE Transactions on Robotics and Automation*, April 1990.
- [67] Y. Han et al. Pose determination using tree annealing. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.
- [68] R. Haralick. Determining camera parameters from the perspective projection of a rectangle. *Pattern Recognition*, March 1989.
- [69] S. Harmon. USMC ground surveillance robot: A testbed for autonomous vehicle research. *Proceedings University of Alabama Robotics Conference*, 1984.
- [70] M. Hebert. Building and navigating maps of road scenes using an active sensor. *Proceedings IEEE International Conference on Robotics and Automation*, 1989.
- [71] M. Hebert et al. Terrain mapping for a roving planetary explorer. *Proceedings IEEE International Conference on Robotics and Automation*, 1989.
- [72] R. Hickling and S. Marin. The use of ultrasonics for gauging and proximity sensing in air. *Journal of the Acoustical Society of America*, April 1985.
- [73] B. Horn. *Robot Vision*. MIT Press, 1986.
- [74] Q. Huang. Personal communication. May, 1992.
- [75] K. Hung et al. Robot location determination in a complex environment by multiple marks. *Pattern Recognition*, June 1988.
- [76] T. Huntsberger et al. Representation of uncertainty in computer vision using fuzzy sets. *IEEE Transactions on Computers*, February 1986.
- [77] T. Hwang and J. Clark. On local detection of moving edges. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.
- [78] T. Hwang and J. Clark. A spatio-temporal generalization of canny's edge detector. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.

- [79] A. Jain and T. Newman. Range-intensity histogram for segmenting ladar images. *Pattern Recognition*, January 1992.
- [80] T. Kanade. Picture processing by computer complex and recognition of human faces. Technical report, Kyoto University, Department of Information Science, 1973.
- [81] T. Kanade and S. Shafer. Image understanding research at carnegie-mellon. *DARPA Image Understanding Workshop*, 1989.
- [82] P. J. Kielczynski et al. Ring piezoelectric transducers radiating ultrasonic energy into the air. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, January 1990.
- [83] D. Kite and M. Magee. Determining the 3D position and orientation of a robot camera using 2D monocular vision. *Pattern Recognition*, August 1990.
- [84] P. R. Klarer. Autonomous land navigation in a structured environment. *IEEE Aerospace and Electronic Systems Magazine*, March 1990.
- [85] K. Kluge and C. Thorpe. Explicit models for robot road following. *Proceedings IEEE International Conference on Robotics and Automation*, 1989.
- [86] Y. Koren and J. Borenstein. Potential field methods and their inherent limitations for mobile robot navigation. *Proceedings IEEE International Conference on Robotics and Automation*, 1991.
- [87] D. Kriegman and E. Triendl. Stereo vision and navigation in buildings for mobile robots. *IEEE Transactions on Robotics and Automation*, December 1989.
- [88] E. Krotkov. Mobile robot localization using a single image. *Proceedings IEEE International Conference on Robotics and Automation*, 1989.
- [89] E. Krotkov et al. An agile stereo camera system for flexible image acquisition. *IEEE Transactions on Robotics and Automation*, February 1988.
- [90] E. Krotkov and R. Kories. Adaptive control of cooperating sensors: Focus and stereo ranging with an agile camera system. *Proceedings IEEE International Conference on Robotics and Automation*, 1988.
- [91] R. Kuc and B. Barshan. Navigating vehicles through an unstructured environment with sonar. *Proceedings IEEE International Conference on Robotics and Automation*, 1989.

- [92] B. Kuipers. Modeling spatial knowledge. *Cognitive Science*, February 1978.
- [93] B. Kuipers and T. Levitt. Navigation and mapping large-scale space. *Artificial Intelligence*, Summer 1988.
- [94] S. Kuo and G. Cross. A two-step string-matching procedure. *Pattern Recognition*, July 1989.
- [95] M. Landy. *HIPS*. Human Information Processing Laboratory, NYU, New York, New York.
- [96] J. Latombe. *Robot Motion Planning*. Kluwer Academic, 1991.
- [97] J. Le Moigne. Domain-dependent reasoning for visual navigation of roadways. *IEEE Transactions on Robotics and Automation*, August 1988.
- [98] C. C. Lee. Fuzzy logic in control systems: Fuzzy logic controller, part ii. *IEEE Transactions on Systems, Man, and Cybernetics*, March 1990.
- [99] S. Lee and Y. Kay. An accurate estimation of 3-D position and orientation of a moving object for robot stereo vision: Kalman filter approach. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.
- [100] J. Leonard and H. Durrant-Whyte. Mobile robot localization by tracking geometric beacons. *IEEE Transactions on Robotics and Automation*, June 1991.
- [101] J. J. Leonard and H. F. Durrant-Whyte. *Directed Sonar Sensing for Mobile Robot Navigation*. Kluwer Academic, 1992.
- [102] M. Leung and T. Huang. Detecting wheels of vehicle in stereo images. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.
- [103] S. Liou and R. Jain. Road following using vanishing points. *Computer Vision, Graphics, and Image Processing*, March 1987.
- [104] V. Lumelsky et al. Dynamic path planning in sensor-based terrain acquisition. *IEEE Transactions on Robotics and Automation*, August 1990.
- [105] V. Lumelsky and T. Skewis. A paradigm for incorporating vision in the robot navigation function. *Proceedings IEEE International Conference on Robotics and Automation*, 1988.

- [106] R. Luo and M. Lin. Robot multi-sensor fusion and integration: Optimum estimation of fused sensor data. *Proceedings IEEE International Conference on Robotics and Automation*, 1988.
- [107] L. Matthies and A. Elfes. Integration of sonar and stereo range data using a grid-based representation. *Proceedings IEEE International Conference on Robotics and Automation*, 1988.
- [108] L. Matthies and S. Schafer. Error modeling in stereo navigation. *IEEE Transactions on Robotics and Automation*, June 1987.
- [109] D. McDermott and E. Davis. Planning routes through uncertain territory. *Artificial Intelligence*, 1984.
- [110] C. McGillem and T. Rappaport. Infrared location system for navigation of autonomous vehicles. *Proceedings IEEE International Conference on Robotics and Automation*, 1988.
- [111] D. Miller. Two dimensional mobile robot positioning using onboard sonar. *Proceedings Pecora IX Remote Sensing Symposium*, 1984.
- [112] D. Miller. A spatial representation system for mobile robots. *Proceedings IEEE International Conference on Robotics and Automation*, 1985.
- [113] A. Mitiche and G. Habelrih. Interpretation of straight line correspondences using angular relations. *Pattern Recognition*, March 1989.
- [114] H. Moravec. The Stanford cart and the CMU rover. *Proceedings IEEE*, July 1983.
- [115] H. Moravec and A. Elfes. High resolution maps from wide angle sonar. *Proceedings IEEE International Conference on Robotics and Automation*, 1985.
- [116] N. Nandhakumar and J. Aggarwal. Thermal and visual information fusion for outdoor scene perception. *Proceedings IEEE International Conference on Robotics and Automation*, 1988.
- [117] H. Nasr and B. Bhanu. Landmark recognition for autonomous mobile robots. *Proceedings IEEE International Conference on Robotics and Automation*, 1988.
- [118] N. Nasrabadi et al. Stereo vision correspondance using a multi-channel graph matching technique. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.

- [119] H. Nguyen and P. Heckman. Real-time pattern recognition for guidance of an autonomous undersea submersible. *Proceedings IEEE International Conference on Robotics and Automation*, 1988.
- [120] H. Nishihara and T. Poggio. Stereo vision for robotics. *Proceedings International Symposium on Robotics Research*, 1984.
- [121] T. Olson and D. Coombs. Real-time vergence control for binocular robots. Technical report, Department of Computer Science University of Rochester, 1990.
- [122] D. Payton. An architecture for reflexive control. *Proceedings IEEE International Conference on Robotics and Automation*, 1986.
- [123] T. Poggio et al. The MIT vision machine. *DARPA Image Understanding Workshop*, 1988.
- [124] A. Preciado and D. Meizel. Fusion of multi-sensor data: A geometric approach. *Proceedings IEEE International Conference on Robotics and Automation*, 1991.
- [125] J. Prewitt. Object enhancement and extraction. In B. Lipkin and A. Rosenfeld, editors, *Picture Processing and Psychopictorics*. Academic Press, 1970.
- [126] P. Puget and T. Skordas. An optimal solution for mobile camera calibration. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.
- [127] L. Ray. Estimation of modeled object pose from monocular images. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.
- [128] L. G. Roberts. Machine perception of three-dimensional solids. In J. Tippett et al., editors, *Optical and Electro-Optical Information Processing*. MIT Press, 1965.
- [129] M. Rokey. Remote mission specialist: A study in real-time, adaptive planning. *IEEE Transactions on Robotics and Automation*, August 1990.
- [130] J. Roos. GPS comes onboard. *Armed Forces Journal International*, July 1990.
- [131] Y. Roth-Tabak and R. Jain. Building an environment model using depth information. Technical report, University of Michigan, 1988.
- [132] P. K. Sahoo and S. S. A survey of thresholding techniques. *Computer Vision, Graphics, and Image Processing*, May 1988.

- [133] K. Sarackik. Characterising an indoor environment with a mobile robot and uncalibrated stereo. *Proceedings IEEE International Conference on Robotics and Automation*, 1989.
- [134] J. Schneider and H. Dulimarta. Robot navigation via ultrasound sensors. Technical report, Computer Science Department Michigan State University, 1989.
- [135] J. Schwartz and M. Sharir. A survey of motion planning and related geometric algorithms. *Artificial Intelligence*, 1988.
- [136] T. Shakunaga. 3-d corridor scene modeling from a single view under natural lighting conditions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, February 1992.
- [137] M. Sharir and A. Schorr. On shortest paths in polyhedral spaces. *SIAM Journal of Computing*, February 1986.
- [138] S. Shaw et al. Fusion of radar and optical sensors for space robotic vision. *Proceedings IEEE International Conference on Robotics and Automation*, 1989.
- [139] Z. Shiller and Y. Gwo. Dynamic motion planning of autonomous vehicles. *IEEE Transactions on Robotics and Automation*, April 1991.
- [140] K. Skifstad. *High-Speed Range Estimation Based on Intensity Gradient Analysis*. Springer-Verlag, 1991.
- [141] B. Steer and T. Atherton. Design for navigation. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.
- [142] F. Stein and G. Medioni. Efficient two dimensional object recognition. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.
- [143] K. Sugihara. Some location problems for robot navigation using a single camera. *Computer Vision, Graphics, and Image Processing*, April 1988.
- [144] H. Tahani and J. Keller. Information fusion in computer vision using the fuzzy integral. *IEEE Transactions on Systems, Man, and Cybernetics*, May 1990.
- [145] C. Thorpe. The CMU Rover and the FIDO vision and navigation system. *Proceedings Symposium Autonomous Underwater Robots*, 1983.
- [146] A. Tirumalai and B. Schunck. Dynamic stereo with self-calibration. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.

- [147] Transitions Research Corporation, Danbury, CT. *TRC Labmate User Manual*, 1989.
- [148] M. Trivedi et al. Developing robotic systems with multiple sensors. *IEEE Transactions on Systems, Man, and Cybernetics*, November 1990.
- [149] R. Tsai and R. Lenz. Real time versatile robotics hand/eye calibration using 3D machine vision. *Proceedings IEEE International Conference on Robotics and Automation*, 1988.
- [150] T. Y. Tseng and C. M. Klein. New algorithm for the ranking procedure in fuzzy decisionmaking. *IEEE Transactions on Systems, Man, and Cybernetics*, September 1989.
- [151] R. Tummala and C. Lin. Mobile robot navigation using sensory feedback. *Proceedings IEEE International Conference on Systems Engineering*, 1991.
- [152] M. Turk et al. VITS - a vision system for autonomous land vehicle navigation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, May 1988.
- [153] R. Wallace et al. First results in robot road following. *Proceedings International Joint Conference on Artificial Intelligence*, 1985.
- [154] C. Wang. Location estimation and uncertainty analysis for mobile robots. *Proceedings IEEE International Conference on Robotics and Automation*, 1988.
- [155] C. M. Wang. Error analysis of spatial representation and estimation of mobile robots. Technical report, General Motors Research Laboratories, 1986.
- [156] Y. Watanabe and S. Yuta. Position estimation of mobile robots with internal and external sensors using uncertainty evolution technique. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.
- [157] A. Waxman et al. A visual navigation system for autonomous land vehicles. *IEEE Transactions on Robotics and Automation*, April 1987.
- [158] S. Xie. View planning for mobile robots. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.
- [159] L. A. Zadeh. Fuzzy sets. *Informat. Contr.*, August 1965.
- [160] W. R. Zhang et al. Pool2: A generic system for cognitive map development and decision analysis. *IEEE Transactions on Systems, Man, and Cybernetics*, January 1989.

- [161] Z. Zhang and O. Faugeras. Building a 3D world model with a mobile robot: 3D line segment representation and integration. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.
- [162] J. Zheng and S. Tsuji. Panoramic representation of scenes for route understanding. *Proceedings IEEE International Conference on Robotics and Automation*, 1990.