A Bayesian Approach to Optimal Sensor Placement

A.J. Cameron

Robotics Research Group

Department of Engineering Science

University of Oxford

Michaelmas Term, 1989

This thesis is submitted to the Department of Engineering Science, University of Oxford, in fulfilment of the requirements for the degree of Doctor of Philosophy. The research reported in this thesis is my own work, except where otherwise stated.

Author signature removed

Abstract

A.J. Cameron D.Phil.

University College Trinity Term, 1989

A Bayesian Approach to Optimal Sensor Placement

By "intelligently" locating a sensor with respect to its environment it is possible to minimize the number of sensing operations required to perform many tasks. This is particularly important for sensing media which provide only "sparse" data, such as tactile sensors and sonar. In this thesis, a system is described which uses the principles of statistical decision theory to determine the optimal sensing locations to perform recognition and localization operations. The system uses a Bayesian approach to utilize any prior object information (including object models or previously-acquired sensory data) in choosing the sensing locations.

Acknowledgements

I would like to thank my supervisor, Dr Hugh Durrant-Whyte, for his guidance and encouragement over the last three years. His enthusiasm for my research has been a great source of inspiration. Dr Ron Daniel also provided supervision during the early stages of my work, whilst I was undertaking research into tactile sensors. Professor Mike Brady has been a great help in providing constructive criticism of my work, and all the other members of the Robotics Research Group have assisted me in some way.

I would finally like to thank the Rhodes Trust for providing me with the opportunity to study in Oxford, by funding my scholarship.

Contents

Chapter 1

Introduction

1.1 Sensing and Robotics

The Concise Oxford Dictionary defines a robot as an "apparently human automaton, intelligent and obedient, but impersonal machine" [47]. Although some aspects of this definition bear more relation to the android robots depicted in science fiction than those found in industry and research laboratories, it is the attempt to bestow machines with **intelligent** (or flexible) behaviour, which characterizes the field of robotics.

In order to be able to perform operations intelligently, robots will need to be able to perceive their environment, and react accordingly. Tasks such as navigating in a cluttered environment, identifying and selecting objects randomly scattered on a conveyer belt, and manipulating unfamiliar objects, represent the type of functions that it is hoped that robots will be able to perform reliably. The ability to function in an unstructured environment, however, requires the robot to make decisions in real-time, rather than simply performing a sequence of predetermined operations. This adaptability characteristic is an important aspect of intelligence. To perform such tasks, the robot must be able to sense its environment before it can respond to it. A popular definition of robotics is "the intelligent connection of perception to action" [9]. This perception will rely on detailed information provided by a range of sensor systems.

The interaction of separate subsystems performing different roles leads to the concept of a robot system. A typical robot system will consist of actuators for acting in or on the robot's environment, sensors to obtain knowledge about the state of the actuator and the environment, a controller and drivers to move the mechanism and sensors, and a reasoning system to select suitable actions in response to sensory inputs [16]. Without sensing, there is no function for the reasoning system, and the resulting apparatus is no more than a reprogrammable machine.

It has been recognized by many researchers that mounting a single sensor on a robot in a passive mode will not provide sufficient data to give it the perceptual abilities necessary to perform everyday tasks. Just as humans rely on five sensors (sight, hearing, touch, taste and smell), so truly-flexible robots will need to respond to a range of sensory modalities. This leads to the problem of multi-sensor fusion which has been the subject of much recent sensor research. (See, for example, [1,16,45].)

Another method by which the sensory power can be increased is to control and position the sensors actively during data acquisition. This is a method commonly used by humans and other animals to extract data from the environment. Eyes can be refocussed and redirected, and fingers can explore along chosen paths to obtain desired sensory information.[1] In robotics, by changing the location, orientation and focus of a camera, a different – and perhaps more informative – view will be obtained. Such sensor control is termed "active" sensing and has application to almost all sensing media. The strategies required for "intelligent" active sensing are the subject of this thesis.

1.2 Characteristics of different sensing media

Many different sensing media have been implemented in robotics, including vision, tactile sensing, force sensing, sonar, radar, infra-red sensing, and other range sensors. These robot sensors can be characterized as performing one of two roles – internal monitoring of the state of the robot (such as joint angle measurement, or joint torques), or external sensing of the robot's environment. The external sensing media can in turn be divided into contact and non-contact sensors, according to the nature of their interaction with the object they are sensing [19]. An alternative division of the external sensors (although along similar lines) is proposed by Beni et al. [5], who characterize sensors as performing two basic functions: acquiring an object; and (once the target object has been acquired) inspecting and guiding manipulation of the object. In general, they claim, the

[1] A significant amount of robotics research has been devoted to an analysis of human exploratory haptic movements, with a view to understanding the strategies by which humans collect tactile data. A physiological study of active human touch is provided in [21], and various interpretations of the relevance of human motion strategies to robotics are provided by Bajcsy [4], Klatzky et al. [37], Hollerbach [34] and Stansfield [43,44].

sensors used for acquisition are non-contact sensors, and those used for inspection and manipulation contact the object.

Another classification of external sensors is proposed in this thesis, based on the nature of the data that the sensor returns from the environment. If the field of view of the sensor is generally larger than the sensed object of interest (as is often the case in robot vision), with the result that the whole object can be sensed at any one instant, then this type of sensor will be termed a "dense" sensor. On the other hand, if the scope of the sensor is, in general, smaller than the object of interest (as is often the case with contact sensors), with the result that the sensor must be repositioned to accumulate information about the sensed object, then the sensor will be termed a "sparse" sensor. The type of data processing that is required for sparse sensory data is fundamentally different from that used for dense data.

For "dense" sensors, processing generally involves intensive analysis of a single image (or set of data) to extract the desired information. For "sparse" sensors, there will generally be significantly less information present in a given data set, and hence even intensive processing is unlikely to extract sufficient information to perform a recognition task. (Another way of understanding this distinction, is that each instance of sparse sensory data will, on its own, be underconstrained for the task at hand.)[2]

[2] Although sparse data cannot be handled by the pattern recognition techniques often applied to dense data, it is possible to process a large array of dense data using "sparse techniques" by considering sparse patches of the dense data. This is an approach that has been used in vision, where attention is focussed on small patches of data, and recognition is performed on the basis of these sparse regions.

An important feature of many such sensing systems, where only a small portion of the object of interest can be sensed at any instant, is the necessity of relocating the sensor to build up a model of the object. Typical systems include tactile sensing, sonar and range sensing systems. Some robot vision systems also return sparse data. One example is the field of object recognition in the presence of overlapped objects, where identification must be performed on the basis of data returned from unobscured portions of the object of interest. Another situation is where the camera is relatively close to the object and only a small portion of the object fills the camera's field of view at any one instant.

It is this class of sensors, which rely on relocation or redirection to accumulate sufficient data for a given task, that are dealt with in this thesis.

1.3 Outline

This thesis investigates the problem of acquiring information using sensors which provide only sparse data, and was motivated by previous research into tactile sensors, described in [13,14]. In particular, the problem of processing tactile sensor data, and choosing optimal sensing locations in this field were addressed. (An initial investigation of these topics was reported in [12].) In this thesis, the approach is generalized for all sparse sensing media, and strategies are proposed for relocating a sensor to accumulate information optimally.

The approach taken to determine optimum placement strategies is to apply

(See, for example, [48,25].)

Bayesian statistical decision theory to the choice of sensing actions. This approach makes optimal use of any prior information, which for sensing tasks may include object models, previous sensory data, and any preconceptions about the likelihood of different alternatives.

Chapter 2 describes previous approaches to object recognition and localization from sparse data. As most of the work on this problem has been from the perspective of tactile sensing, this field receives the most attention. The merits of previous approaches are discussed and the merit of a **statistical approach** and **intelligent sensor placement** are advocated. In particular, the interpretation tree approach, which has proved popular for this task, is described and discussed.

Chapter 3 provides a brief introduction to statistical decision theory and the Bayes decision principle. It also introduces the important concepts of state and prior information functions and the utility function. The manner in which these concepts can be applied to sensor placement problems in robotics are described.

Chapter 4 introduces a function termed the probabilistic membership function (PMF). This function was developed to perform the role of a prior information function in the decision strategy. The PMF contains all the information about object location that is currently possessed utilizing object models (which may be uncertain) and previous sensory data (which may also be uncertain). The derivation of this function is described and several examples given.

Chapter 5 deals with utility functions. The utility function assigns a value to information obtained in response to a given sensory action. As the aim of statistical decision theory is to choose the action that maximizes (in some statistical sense) the utility function, this function is determined by the nature of the task to be performed. Several utility functions are proposed for typical sensory tasks.

Chapter 6 shows how the current information about the likelihoods of alternative interpretations can be represented in a probabilistic interpretation tree. This enables an ordering to be placed on the admissible interpretations at any stage of the recognition process. As uncertainty in sensor data may never enable the object to be identified with absolute certainty, this ordering is important for arriving at a conclusion within a finite time. The techniques of sequential analysis are also introduced as a method for determining when a solution to the recognition problem can be concluded within previously-stated error bounds.

Chapter 7 describes several implementations of the ideas presented in Chapters 3 to 6. All the implementations involve a "sparse data" object recognition system consisting of a camera with restricted field of view mounted on a robot arm. Specific implementations addressed include location sensing and path sensing, and efficient methods are developed and tested to reduce significantly the processing time required to select the sensor placement strategy. The results of the implementations are presented in support of the theory developed in the preceding chapters.

Further work and refinements are described in Chapter 8. In particular, the generalizations of the theory necessary to deal with three dimensional sensing problems are discussed. No hardware was available to test these techniques, however, theoretically, they differ only slightly from the two-dimensional methods. Differences result from more degrees of uncertainty in the feature uncertainty functions in three dimensions, which require attention, particularly to remove rotational uncertainty, and an increase in the number of parameters which are required to specify the sensing action. This chapter also includes an analysis of the implementations performed, and the limitations exposed by the experimental work. Remaining problems are discussed, and finally a summary of the major contributions of this work is presented.

Chapter 2

Sparse Sensing and Object Recognition

2.1 Background

The problem of using data from sparse sensing media has been studied most intensively for tactile sensors. Although all researchers are not in agreement about the characteristics that will be possessed by tactile sensors when they become readily available to the robotics community, it is accepted that in most sensing situations, the contact area between the sensor and sensed object will be small in relation to the object's size [31,28,30,29]. Hence the sensory data will be sparse, and most algorithms proposed for using tactile data have recognized the characteristic that several sensory actions will be required to accumulate sufficient information to perform a given task. (An exception is the work of Hillis [32,33], which assumed a large tactile sensor sensing small planar objects, such as washers and screw heads, such that the whole object could be sensed at any one instant. Hillis used pattern recognition techniques to determine the identity and location of the sensed object from a given set, however these techniques cannot be applied to situations where a sensory "image" of the whole object is not available.)

Several other researchers have also addressed the problem of fusing the sparse data provided by such systems, and using it to perform tasks such as object recognition, localization and path planning. The approach adopted by different researchers has depended to a large extent upon the nature of the data they assumed would be provided by the tactile sensor.

Much of the work on tactile object recognition has envisaged tactile sensors working in conjunction with machine vision [1,3,45]. In these cases, the tactile device was intended to perform the role of confirming previously acquired visual data, or resolving between interpretations proposed by the visual data. In either event, the tactile sensing was directed to obtain specific data by the visual data. Thus the sensing was guided, however there was little uncertainty in the location of the object, as the location was strongly constrained by the prior (visual) data.

Gaston and Lozano-Pérez [20] and Grimson and Lozano-Pérez [26] addressed the problem of a tactile sensor working independently of other sensors. They assumed that the tactile sensor would only be capable of returning the location of a single point of contact between the sensor and object, and the direction of the normal to the object's surface at this point. Their algorithm for object recognition involves the construction of an interpretation tree for each object in the object set (as described in Section 2.2), whereby each sensed point is assigned to each face of the object model, and local constraints are used to cull inadmissible combinations of points and faces from the tree.

The concept of a sensor returning only a point contact with the sensed object represents the most extreme form of "sparse" data.[1] Other researchers have assumed that a tactile sensor will be able to return data from a small contact region. This results in small patches of "dense" data and raises the possibility of detecting geometric surface features such as edges, corners and holes. It may be that more sophisticated processing is required to extract these sensed features, however, it is still assumed that the size of the sensed patch is small in relation to the sensed object, and hence that relocation is still required.

Browse [11] assumed tactile sensory data from a contact region, and considered the constraints imposed upon possible object locations by detecting specific features. As more data is acquired, fewer interpretations are admissible, until the object is eventually identified and localized.

It has been recognized by several researchers that object features can play an important role in object recognition. The ability of a specific feature to constrain an object interpretation has been termed its saliency and has been utilized in recognizing visually-occluded objects by Turney, Mudge and Volz [48]. The work of Bolles and Cain [8] also uses certain object features as a focal point for recognizing and locating partially-visible objects. There is a similarity between the problems of occluded visual data and other sensing methods in which the whole object

[1] There is some contradiction here in the assumptions of Grimson and Lozano-Pérez in that the ability to measure the surface normal at a point implies the ability to sense over a region of finite extent.

cannot be sensed at any instant, and it is necessary to base any conclusions on data from certain subsets of the object. (This similarity has also been recognized by Grimson and Lozano-Pérez who have applied their sparse data techniques to the problem of localization on the basis of occluded visual data [25]).

Both the sparse techniques of Grimson and Lozano-Pérez and the saliency methods of Turney, Mudge and Volz have merit depending on the nature of the sensory data. Stein [46] has developed a method of combining these techniques to utilize optimally any sensory data that is available for the task of recognizing occluded objects. This ability to use all data in an optimal manner is seen as an important goal of a sensing system, and is one that we set out to achieve with the systems described in this thesis.

2.2 Interpretation trees and their limitations

The concept of the interpretation tree (IT) as a tool for object recognition was introduced by Gaston and Lozano-Pérez [20] and Grimson and Lozano-Pérez [26] and has since been used for this task by several other researchers (including Ellis [17] and Schneiter [41]). In this section the generation of the IT is described, along with its application to object recognition.

Consider the task of identifying a two-dimensional object from a set of n polygons, O_j, having e_j sides each. (The treatment of the three-dimensional problem is similar, with faces performing the role of sides [26].) Once the first sensed point is obtained, an IT is generated for each object with the root node of each tree having e_j descendants. This first level represents each possible interpretation for the first sensed point, which can (in the absence of any other information) be interpreted as any object edge. When a second sensed point is obtained, a second level is added to the tree with each of the e_j branches of the first level sprouting into e_j branches at the second level. Thus, after two sensed points, there are a total of $(e_j)^2$ branches for each object O_j, corresponding to different pairings of sensed points and object edges. After f sensed points, the total number of possible interpretations will have multiplied to $\sum_{j=1}^{n}(e_j)^f$.

In practice, almost all of the interpretations will be inconsistent with the sensed data. By considering local constraints between pairs of sensed points, branches and whole sub-trees can be pruned. One such local constraint is the distance between the sensed points compared with the possible range of distances between points lying on the designated sides of the object model. This leads to a significant reduction in the number of possible interpretations.

A common misunderstanding of the IT is that it is a search tree (as is commonly used in some Artificial Intelligence tasks [40,51]). As the IT can only be constructed in a breadth-wise manner, there is no possibility of pursuing specific branches in seeking solutions to a given problem. The IT simply displays the current status of knowledge about possible interpretations and provides no assistance in determining sensing strategies.

One draw-back of the interpretation tree approach (as it has been used up until now) is that it places no ordering upon the admissible interpretations. Pairings of sensed points and object edges are judged either admissible or they are culled, and no information is provided regarding the relative likelihood of different interpretations until only one remains.

The approach taken in this thesis is to maintain probabilistic estimates of confidence in different interpretations. In the presence of uncertain information, it may never be possible to cull the interpretation tree down to only one branch, however the use of probabilistic measures and sequential analysis enables a conclusion to be achieved within stated confidence levels. As more sensory data becomes available the prior information function can be updated using Bayes' rule, to provide the current probabilities for the admissible interpretations, and also to act as the prior information for the next sensory stage. This gives rise to the concept of a probabilistic interpretation tree, which is discussed further in Chapter 6.

Another limitation of the IT (as applied by Grimson and Lozano-Pérez) is that it makes no provision for the failure to detect the object with a given sensory action, although such an outcome can provide equally strong constraints on the possible interpretations. The only data which is interpreted corresponds to point detection of edges. The approach taken in this thesis is that all available data should be integrated (whether it be from non-detection, edges, corners, or other features) and utilized maximally in the recognition task.

2.3 "Intelligent" sensor placement

All systems that have been proposed to perform object recognition or localization using sparse sensory data have acknowledged that data must be accumulated from several locations by repositioning the sensor. Few, however, have proposed strategies for selecting the sensing locations or directions. Although it is clear that judicious placement of the sensor will reduce the number of sensing operations required and the quantity of data to be processed, most systems have settled for random sensor placement. This is despite the fact that many researchers have indicated the desirability of guided-sensing [11,20,26].

The need to relocate the sensor also arises in other sensing applications where the characteristics of the sensing medium are such that the problem of reconstructing object information from the sensory data is under-constrained. Examples of this are the photoelastic tactile sensor developed by the author [13,14] and two-dimensional robot vision. It is envisaged that guided sensing would also find applications in these areas.

The attempts that have been made at selecting optimal sensing locations (including those by Grimson [23,22,24], Schneiter [41,42] and Ellis [17]) rely on random sensing to remove all but a finite number of admissible interpretations and locations, and then use guided sensing to help discriminate between the remaining cases. No attempts have been made to guide the sensor during the early sensing stages when uncertainties with regard to localization within a continuous range may result in infinitely many interpretations of the sensory data. Benefits of increased speed and reduced data storage stand to be gained by guiding the sensing from the first sensed point rather than only guiding the latter stages.

This is the problem that is specifically addressed in this thesis: How to choose the optimal sensing action to solve a recognition, localization or navigation problem, making proper use of any previous sensory data, knowledge of object models, and prior expectations of the result.

It is anticipated that the techniques chosen should be equally applicable at any stage of the sensing process. As this problem basically represents a choice over a set of sensing actions, the natural manner in which to consider this choice is as a decision problem. In particular, a statistical decision theoretic approach is advocated and described in this work, the fundamentals of which are described in the next section.

Chapter 3

Statistical Decision Theory

3.1 Introduction

Statistical decision theory formalizes the problem of taking optimal actions with uncertain information. This problem is encountered not only in science, but even in routine decisions faced by everyone in their daily lives. Tasks such as choosing the quickest route when commuting to work or choosing the best savings investment, involve uncertainty in factors such as traffic or future economic conditions, which cannot be known at the time of the decision, and yet which play an important role in determining the merit of the decision taken. The application of statistics to decision problems has enabled the uncertainty to be assigned a measure, and alternative actions to be compared quantitatively. This has led to the study of statistical decision theory which has found wide application in the past fifty years, particularly in the study of economics, and which has led to the development of new and related fields, including game theory.

The terminology used in this chapter, and throughout this thesis, has been adapted from that used by Berger [6], however, some aspects are also drawn form other texts on statistical decision theory including Blackwell and Girshick [7], De Groot [15] and Wald [50].

There are four basic elements of statistical decision theory which must be defined in order to analyze any given problem. These are:

1. the state of the environment, or the state of nature, termed θ;

2. the decision to be made, or action, termed a;

3. a utility function, dependent on θ and a, termed $U(\theta, a)$;

4. a prior information function on θ, termed $\pi(\theta)$.

Uncertainty in the state of nature leads to uncertainty as to the optimal action. This is due to the dependence of the outcome of the action on the (only partly known) state. If the state of nature were known exactly, the decision problem would be trivial, as the merits of alternative actions could be compared for the known state. It is because the action must be chosen without complete knowledge of the state, and hence without complete knowledge of the outcome of the action, that a decision problem arises.

The action is the control that can be exercised upon the outcome of the event. Different actions lead to different outcomes for a given state of nature. The aim of statistical decision theory is to choose the action which optimizes (in some statistical sense) the outcome, when the actual state of nature is only partly known. In some cases, the required action may be simply to nominate a value for the state θ. This class of problems is known as estimation problems.

A key element of statistical decision theory is the utility function $U(\theta, a)$ which assigns a value to the outcome of the action, a, when the state of nature is θ. These utility functions will often be chosen subjectively, but aim to reflect the merit of the chosen action for the given state of nature. The most desirable action is that which maximizes $U(\theta, a)$ when the state of nature is θ. If θ were known the optimal action could be simply determined from the utility function for the known state.

Any information that is possessed about the state can be represented by a prior information function, $\pi(\theta)$. Information contained in $\pi(\theta)$ is seldom precise and may even be subjective. The function $\pi(\theta)$ is most commonly a probability density on θ. Thus, if a problem situation can assume only a finite number of discrete states, then the probability of state θ_i occurring is $\pi(\theta_i)$. If, on the other hand, there is some continuum of possible states, then the probability of the actual state being within a subset A of the admissible states is

$$P(\theta \in A) = \int_A \pi(\theta)\, d\theta$$

Often in statistical decision theory there is also data (often termed x) which is acquired before a decision is made. This data is dependent in some manner on the state θ and hence is used to ascertain some indication of the value of θ to reduce uncertainty in the decision. The class of problems in which no data is acquired, and the action is chosen purely on the basis of the prior information about θ (contained in $\pi(\theta)$) are termed "no data" problems.

The method by which the action is chosen is called the decision principle. Different decision principles determine the manner in which information about the state of nature is applied in attempting to maximize the utility function over the possible actions. The decision principle pursued in this thesis is the Bayes principle, which is described and justified in the following section.

3.2 Bayes' Decision Principle

The two most important principles in decision theory are the Bayes' principle and the minimax principle.

The Bayes' approach attempts to make maximum use of any prior information about the state of nature, as provided by the function, $\pi(\theta)$. This is achieved by weighting the utility function by $\pi(\theta)$ to obtain the expectation of the utility over θ. Thus the Bayes decision is the action a which maximizes the Bayes' risk given by[1]

$$r(\pi, a) = E^{\pi}[U(\theta, a)] = \int_{\text{all } \theta} U(\theta, a)\, \pi(\theta)\, d\theta \tag{3.1}$$

As the state is unknown, it is impossible to claim that any decision principle will nominate the optimal action (ie. that action which maximizes the decision function) at any given stage. The aim of the Bayes approach is to nominate

[1] It is common to define the Bayes risk, r, as the expectation of the loss function (as opposed to utility). When dealing with utility functions, the term payoff function, denoted ρ, has been proposed by Hager [27] as the expectation of the utility over the prior distribution of states. In this thesis, however, we will refer to the risk as the expectation of the utility as defined in Equation 3.1.

actions that will "on average" result in the highest utility. Thus if a sequence of actions were to be taken over a period of time, it could be anticipated the Bayes approach would result in a higher sum of utilities than any other principle.

The minimax approach attempts to maximize the utility, assuming the least favourable state of nature. Minimax techniques have their basis in the field of game theory, where the aim is to outwit an intelligent adversary. It is expected that the opponent will have pursued actions that lead to the least favourable state for the player (as a loss for the player corresponds to a gain for the opponent). Similarly, the optimal action must be chosen by the player with a view to minimizing the utility of the opponent's subsequent actions. Applying these concepts leads to the formulation of the minimax action as the action a which maximizes the utility, when the state θ takes the value that minimizes the function $U(\theta, a)$ for each a. Thus the value of the utility will be $max_a min_\theta[U(\theta, a)]$.[2] The minimax principle attempts to protect against the worst case outcomes, represented by very low utilities. These can result using the Bayes approach (although they should be statistically rare), however such actions are eliminated using the minimax approach as having low values of $min_\theta[U(\theta, a)]$. This leads to a very conservative strategy, which neglects potentially lucrative actions, if there is a small chance of a significant loss.

When dealing with uncertainty in nature, where there is no intelligent adversary, there is less argument for the minimax approach. The loss associated with

[2] The name minimax has its origins in analysis using loss functions, where the optimal action is the one which minimizes the maximum loss given by $min_a max_\theta[L(\theta, a)]$, and hence the name minimax.

worst case outcomes can be represented appropriately by a well-chosen utility function, no matter how devastating the loss may be. In this case, to adopt a minimax approach, and hence assume the worst possible state of nature, is not only pessimistic, but pays little regard to any prior information.

Critics of the Bayesian approach cite its dependence on subjective measures of prior information. Clearly choosing a different prior information function will lead to a different risk function in Equation (3.1) and hence result in an alternative "optimal" action. Any decision principle will, however, make some assumption about the unknown state of nature. The minimax principle assumes the least favourable state, and various non-information prior distributions (see Berger [6]) assume equal likelihood of all states. If the prior information function provides a more accurate measure of the perceived distribution of states (which it should by definition) then it would seem to be sub-optimal not to use it. The minimax approach does guard against the worst case, however, it will be out-performed on average by the Bayesian approach.

For the sensing problems addressed in this thesis, there is no intelligent adversary, but only uncertainty due to incomplete information, which can be described quantitatively. For this reason, the Bayes' principle is pursued exclusively in this thesis, and for these applications it can be relied upon, on average, to return higher utilities.

3.3 Application to "intelligent" sensor placement

The sensor placement problem involves uncertainty in the identity and/or location of the sensed object. The decision that must be made is to choose the optimal sensing action to remove or reduce these uncertainties. To apply the concepts of statistical decision theory it is necessary to define the basic elements: the state, θ; the action, a; the utility function, $U(\theta, a)$; and the prior information function, $\pi(\theta)$.

The action, a, to be taken is the specification of the sensing action, which will involve nominating a location at which to sense or selecting a sensing path. Thus, when choosing the optimal action, the choice that must be made is at which location or along which path to sense. (Whether sensing is at a location or along a path will depend on the nature of the sensor, discussed in later chapters.)

The outcome of the action will depend on the state of nature. For the sensing problem addressed here, the outcome will depend on the identity and location of the object in the workspace. If the object occupies the region of space that lies within the scope of the sensor, then (assuming a perfect sensor) it will be detected. A simple representation of the state of nature, θ, is a two-valued membership (or characteristic) function on the sensing space, corresponding to the presence or absence of the object. If the sensor has the capability to detect and identify specific object features, it will be shown in Chapter 4 that more sophisticated state functions can be generated indicating the presence of given identifiable object features.

Note that the approach taken here is not to consider the decision problem as an estimation problem, as the action modelled by a is not an estimation of the state, but a choice of sensing action. (An estimation problem approach is proposed by Hager [27].)

In the sensor placement problem, the optimal action aims to reduce uncertainty in the object's identity or location, and hence to constrain maximally the object model. Thus the utility function, $U(\theta, a)$, must embody the constraint that derives from detecting the state θ when performing the sensing action a. For the problem of object recognition, the detection of (or failure to detect) the object at a location which allows its identity to be uniquely determined will correspond to a maximum value of the utility function. Conversely, detection of the sensed object along a path which provides no new information (such as sensing where detection is assured for all admissible objects) will have a minimum utility. The value of the utility function may be subjective, but will depend on the specific sensing problem, and on the particular set of objects to be recognized.

The prior information function, $\pi(\theta)$, must contain any knowledge that is possessed about the state. For the sensor placement problem, this consists of any current knowledge of the likelihood of sensing the object at a particular location, based on any knowledge of the object shapes, the location of previously-sensed points, and any preconceptions about the likely identity or location of the object. Given that the state can be described by a two-valued membership function, a probabilistic membership function (PMF) is proposed in this thesis to provide an appropriate formulation for the prior information function. The

PMF indicates the estimated likelihood of the sensed object occupying a given location within the sensing space, by assuming values in the range $[0, 1]$, with a value of zero corresponding to certainty that the object is not present, and a value of one indicating certain presence. Any uncertainties due to sensor uncertainty can also be incorporated into the PMF. It will be shown in Chapter 4 that more sophisticated PMFs can be generated corresponding to the likelihood of detecting given identifiable object features.

The approach taken in this thesis is to consider the decision problem as a "no data" problem. This is despite the fact that data collected at any stage of the sensing process does influence the selection of the next sensing action. Instead of considering the sensory data to be data in the decision theoretic terminology, it is used to update the prior information function at each stage, and the updated prior information function is used solely to represent the state, and hence determine the next sensing action. This difference is only representational, as the problem could be formulated other than as a "no data" problem, however this approach was chosen as it was desired to update the prior function at each stage in any case, and these methods led to fewer computations at each stage.

In Chapter 4, a probabilistic membership function is developed which describes the likelihood of sensing the object in response to a given sensing action. In Chapter 5, utility functions are discussed and various examples proposed for particular sensing tasks. As described in this section, the functions proposed are functions of the sensing action (the action a), and the presence or absence of the object (or a given feature) at that location (the two-valued state θ). The functions are chosen with an aim to assigning high utility to those combinations of action and state which result in a reduction in uncertainty for the task at hand.

Chapter 4

Probabilistic Membership Functions

4.1 Introduction

When a sensor contacts an object, the position and/or pose of the object is constrained. If a recognizable feature, such as an edge or corner, is detected, a tighter constraint is obtained, and this can be tightened further by considering the sensory data in conjunction with data from other locations. Most importantly, once the first sensed point has been obtained, the knowledge of the shape of the object can be used to guide the subsequent search for further contact points. In this chapter, it is shown how knowledge of the object shape together with the location of a sensed point (and any other information from the point, including local surface normal) can be used to determine the probability of sensing the object with each sensing action. The function describing these probabilities will be termed the probabilistic membership function (or PMF).

The analyses presented in this chapter deal with two dimensional functions. The generalization of these methods to three dimensions is discussed in Chapter 8.

The use of a characteristic function to represent the two-dimensional model of an object is described by Horn [35]. The model is assigned a "home" location in the Cartesian coordinate system, and values of one or zero are assigned to each coordinate position depending on whether or not the model occupies that location.

In this thesis, more general membership functions are used which are not necessarily defined in Cartesian space. The coordinate system chosen is dependent on the sensing mechanism of the sensor used. If, for example, the sensor is capable of detecting the presence or absence of the object at a given location in Cartesian space, then this would be the natural coordinate system to use for the membership function. Consider, however, a sonar system which can be directed to sense along a path described by polar coordinates (r, θ) in the horizontal plane. What is of interest here is the probability of sensing the object with this sensor action, and hence it is natural to express the membership function in polar coordinates (ie. as a function of the sensor action). Note that in this case it is necessary to know not whether the object is located at a given point, but whether it intersects a sensing ray described by the given parameters. The probability of this event can also be readily described by an appropriate definition of the PMF.

This chapter deals initially with the probabilistic membership functions in Cartesian coordinates, as would be useful for evaluating the likelihood of detecting an object at a particular location. Methods of generating and updating these functions are explained together with examples. These methods assume that it is possible to sense the workspace at point locations. In practice, the sensor will almost always be of finite extent (even if it returns a single value), and the effects of finite sensor size on the generation and application of the PMFs are discussed in Section 4.4.

Some sensor systems may be able to extract more detailed information from the sensed location than simply the presence or absence of the object. If the sensor can, for example, detect and measure edges and corners, then it will obviously make for more powerful sensing strategies to take account of these abilities of the sensor. To do so it will be necessary to generate expectations for detecting given features at each point in the coordinate space. These functions will be termed feature PMFs and are discussed in Section 4.5.

Finally, the alternative sensing capability of sensing along a path instead of at a location is addressed, and its effect upon the probabilistic membership functions that will be appropriate. Methods of generating and updating PMFs in these situations are described in Section 4.6.

4.2 Generating PMFs

In a two-dimensional domain, the model of an object o may be described by a membership function $O(x, y)$ (associated with a "home" placement for the object, usually at the origin of the coordinate system). If the shape of o is completely known, $O(x,y)$ is defined by

$$O(x,y) = \begin{cases} 1 & \text{if } (x,y) \in o \\ 0 & \text{otherwise} \end{cases}$$

If the object shape is not completely known, a value can be assigned to $O(x,y)$ that represents the confidence that $(x,y) \in o$, subject to the conditions

$$0 \leq O(x,y) \leq 1 \;\; \forall (x,y)$$

and

$$\iint O(x,y)\, dx\, dy = \text{ area of } o \text{ (or its expected value)}$$

The accuracy of the data supplied by the sensor can also be modelled [16,38]. The function $S(x,y)$ will be used to represent the uncertainty in any data returned by the sensor. $S(x,y)$ is a probability density function and satisfies the conditions

$$S(x,y) \geq 0 \;\; \forall (x,y)$$

and

$$\iint S(x,y)\, dx\, dy = 1$$

For a perfect sensor (one which returns data with no uncertainty), $S(x,y) = \delta(x,y)$. All data returned by the sensor should be convolved with $S(x,y)$ to take correct account of the inherent limitations of the sensor.

In addition to uncertainties introduced by the sensor, any attempts to determine accurately the position and pose of the sensed object are also affected by uncertainty in the relative position of the sensor and object, which in turn depend on the shape of the sensed feature. For example, if a corner is sensed, the location of the object may be completely determined. However, if an edge is detected without one of its endpoints, there will be uncertainty in the object's position in the direction parallel to the edge. If a point within the boundaries of the object is sensed without detecting an edge, there will be uncertainty in three degrees of freedom (two translational and one rotational). This "feature uncertainty" can be represented by a distribution function $F(x, y)$ satisfying

$$F(x,y) \geq 0 \;\; \forall(x,y)$$

and

$$\iint F(x,y)\,dx\,dy = 1$$

For the cases of corner and edge mentioned above, $F(x, y)$ will, in general, be represented by a delta function and a uniform distribution in one variable respectively.

Once contact has been established with a particular feature of the object, the object model can be transformed to align the corresponding feature on the model with the sensed point. This gives rise to a new function $O'(x, y)$ describing the transformed model. To incorporate the uncertainty in the object's location, the model must be convolved with the sensor and feature uncertainty functions. Thus a new "blurred" object model, $\Pi(x, y)$, is obtained, representing the probability of sensing the object at each location in the workspace.[1]

$$\Pi(x,y) = O'(x,y) * F(x,y) * S(x,y)$$

[1] Throughout this thesis the symbol * will be used to represent the convolution integral

$$a(x,y) * b(x,y) = \iint a(x-\mu, y-\eta)b(\mu,\eta)\,d\mu\,d\eta$$

This function will be termed the probabilistic membership function (PMF) and has the same properties as the characteristic function, $O(x, y)$, namely (see Appendix A):

$$0 \leq \Pi(x, y) \leq 1 \ \ \forall (x, y)$$

and

$$\iint \Pi \, dx \, dy = \text{ area of } o$$

$\Pi(x, y)$ represents the probability of the location (x, y) lying within the boundary of the object, taking into account uncertainties in the sensed data and object model.

Some examples follow, showing how the PMF can be generated from the membership function of an object.

4.2.1 Example 1: A symmetric object

Consider a square of side length l as shown in Figure 4.1. The characteristic function for the square is[2]

$$O(x, y) = \begin{cases} 1 & \text{if } 0 \leq x \leq l \text{ and } 0 \leq y \leq l \\ 0 & \text{otherwise} \end{cases}$$

It will be assumed that the shape is sensed by a sensor which has no uncertainty, so

$$S(x, y) = \delta(x, y)$$

[2] The specification of $O(x, y)$ is unique only up to arbitrary translations and rotations of the model or coordinate system.

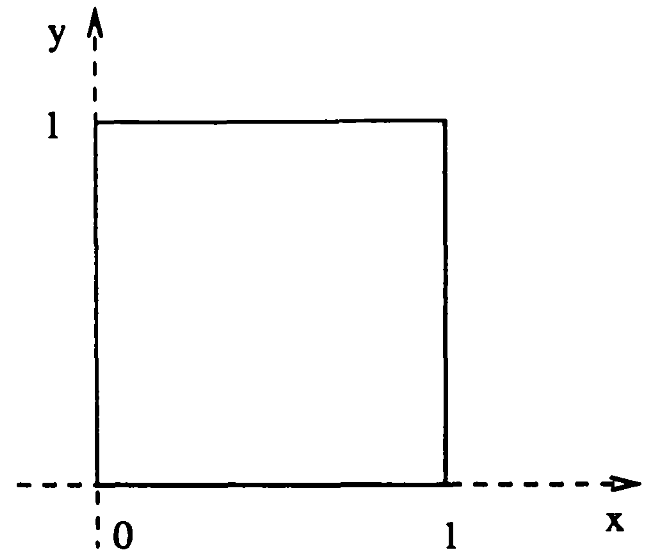

Figure 4.1: Object characteristic function.

Now assume that contact is made with an edge of the square at the point (x_0, y_0) with the edge lying parallel to the x-axis of the coordinate system. As the object is completely symmetric, the sensed edge can be associated with any model edge. For simplicity, it will be associated with the model edge whose endpoints are $(0, 0)$ and $(l, 0)$.

The sensed point could be any point lying on the chosen model edge. This uncertainty in relative position of the sensed point to the feature is described by the feature uncertainty function. Assume the sensed point (x_0, y_0) has been associated with the midpoint of the edge in the model, the point $(l/2, 0)$. The resulting transformed model function would be[3]

$$O'(x, y) = O(x - x_0 + l/2, y - y_0)$$

The actual position of the sensed point could with equal probability be located anywhere within a range of values $\pm l/2$ of this location. In this case, an appropriate formulation for this function is a uniform distribution in the x-dimension,

[3] Any arbitrariness in the choice of the model location is removed by the transformation matching the characteristic function to the sensory data. The resultant transformed model O' is independent of the coordinate system chosen to define O.

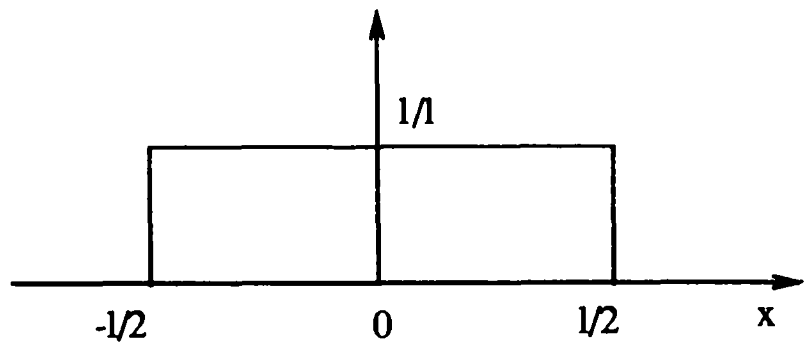

Figure 4.2: Feature uncertainty function.

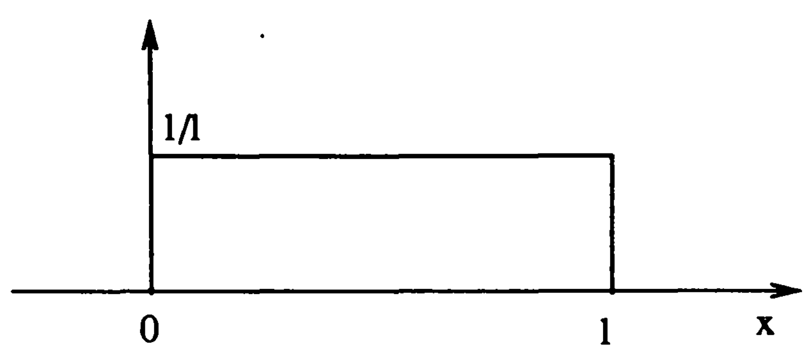

Figure 4.3: Alternative feature uncertainty function.

$U_{(-l/2,l/2)}(x)$, as shown in Figure 4.2.

Alternatively, the sensed point may be associated with one of the endpoints of the modelled edge, say the point $(0,0)$. This would give rise to a different transformed model function

$$O'(x,y) = O(x - x_0, y - y_0)$$

In this case, the uncertainty in the x-value is within the range 0 to $+l$ of the chosen value. Again this distribution is uniform and the resulting feature uncertainty function is shown in Figure 4.3.

Irrespective of which point is used, the resulting function $\Pi(x,y)$ is not affected.

$$\Pi = O'(x,y) * F(x,y) * S(x,y)$$

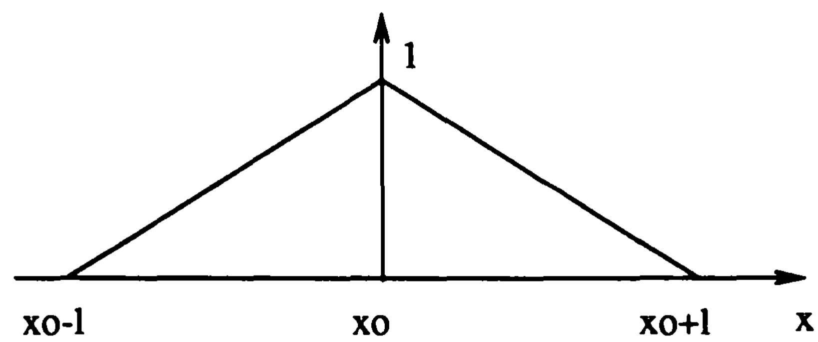

Figure 4.4: Cross-section of probabilistic membership function.

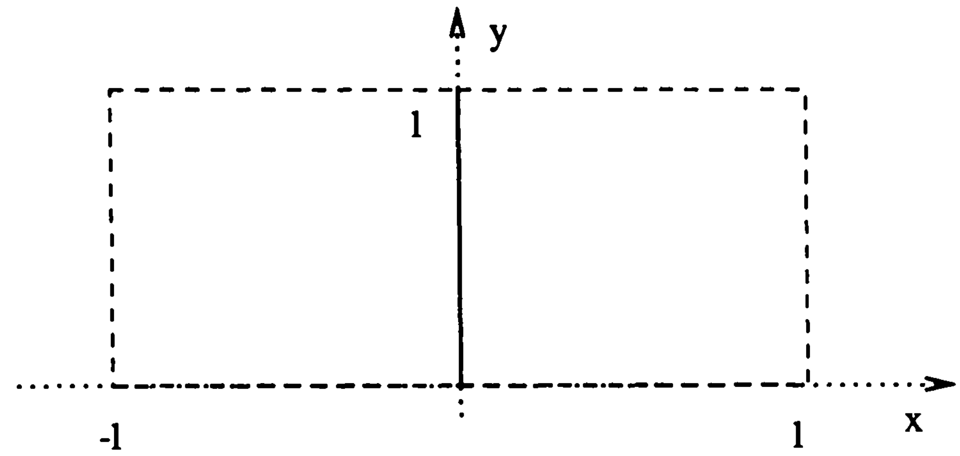

Figure 4.5: Plan view of probabilistic membership function.

$$
\begin{aligned}
&= O'(x,y) * F(x,y) \\
&= \iint O'(x-\mu, y-\eta) F(\mu,\eta)\, d\mu\, d\eta \\
&= \int O'(x-\mu, y) F(\mu, 0)\, d\mu \\
&= 1/l \int_0^l O(x - x_0 - \mu, y - y_0)\, d\mu \\
&\quad \left(\text{or } 1/l \int_{-l/2}^{l/2} O(x - x_0 + l/2 - \mu, y - y_0)\, d\mu\right) \\
&= \begin{cases} 0 & \text{if } y < y_0, \text{ or } y > y_0 + l, \text{ or } x < x_0 - l, \text{ or } x > x_0 + l \\ 1 - \frac{x_0 - x}{l} & \text{if } y_0 \le y \le y_0 + l \text{ and } x_0 - l \le x \le x_0 \\ 1 - \frac{x - x_0}{l} & \text{if } y_0 \le y \le y_0 + l \text{ and } x_0 \le x \le x_0 + l \end{cases}
\end{aligned}
$$

A cross-section of $\Pi(x, y_1)$ along the x-dimension for $y_0 \le y_1 \le y_0 + l$ is given in Figure 4.4. A plan view of the regions of this function is given in Figure 4.5.

As would be expected, the probability of sensing the object again at (x_0, y_1)

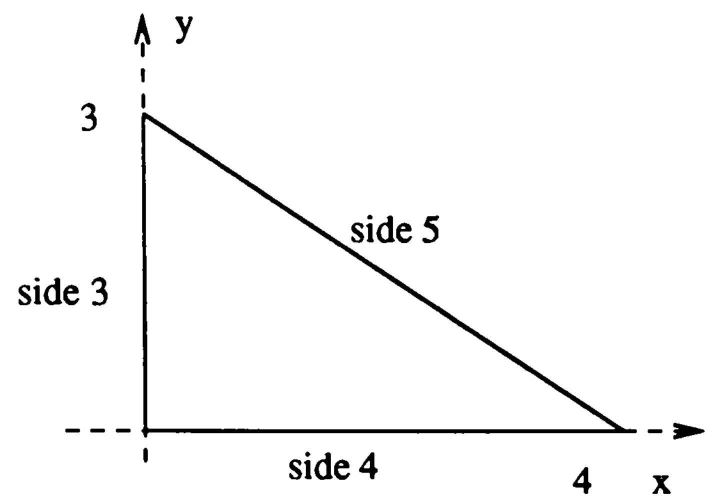

Figure 4.6: Object model (or membership function).

is one (assuming a perfect sensor). The probability decreases in the x-direction with distance from x_0 until it becomes zero, when the distance from the sensed point is greater than the side-length, l. Thus the results obtained are consistent with reasonable expectations.

4.2.2 Example 2: An asymmetric object

Consider a right-triangle with sides of length 3, 4 and 5 units respectively. A possible "home" location is shown in Figure 4.6 for the object giving rise to the characteristic function

$$O(x,y) = \begin{cases} 1 & \text{if } 0 \le y \le 3 \text{ and } 0 \le x \le 4(1 - y/3) \\ 0 & \text{otherwise} \end{cases}$$

As in the previous example it is assumed that the sensor is perfect, and that detection of an edge occurs at the point (x_0, y_0), with the detected edge lying parallel to the x-axis of the coordinate system. The form of $O'(x,y)$ and $F(x,y)$ will depend on which edge is associated with the sensed point, and hence the three edges must be treated as separate cases.

Consider first associating the sensed point with the side of length 4 (termed *side* 4). For the following calculations, the sensed point shall be associated with the left-hand-most point of the edge (although it was seen in the previous example that any point can be chosen by an appropriate formulation of the feature uncertainty function).

Hence $O'(x, y)$ can be calculated.

$$O'(x,y) = \begin{cases} 1 & \text{if } y_0 \le y \le y_0 + 3 \text{ and } x_0 \le x \le x_0 + 4(1 - \frac{y-y_0}{3}) \\ 0 & \text{otherwise} \end{cases}$$

The feature uncertainty function is given by

$$F_{side\ 4}(x,y) = \begin{cases} U_{(0,4)}(x) & \text{if } y = 0 \\ 0 & \text{otherwise} \end{cases}$$

Now $\Pi_{side\ 4}(x, y)$ can be calculated:

$$\begin{aligned} \Pi_{side\ 4}(x,y) &= O'(x,y) * F_{side\ 4}(x,y) * S(x,y) \\ &= O'(x,y) * F_{side\ 4}(x,y) \\ &= \iint O'(x-\mu, y-\eta) F(\mu,\eta)\, d\mu\, d\eta \\ &= \int_0^4 1/4\, O'(x-\mu, y)\, d\mu \\ &= \begin{cases} 0 & \text{if } y \le y_0, \text{ or } y \ge y_0 + 3, \\ & \text{or } x \le x_0 - 4, \text{ or } x \ge x_0 + 4(1 - \frac{y-y_0}{3}) \\ 1 + \frac{x-x_0}{4} & \text{if } y_0 \le y \le y_0 + 3 \text{ and } x_0 - 4 \le x \le x_0 - 4\frac{y-y_0}{3} \\ 1 - \frac{y-y_0}{3} & \text{if } y_0 \le y \le y_0 + 3 \text{ and } x_0 - 4\frac{y-y_0}{3} \le x \le x_0 \\ 1 - \frac{y-y_0}{3} - \frac{x-x_0}{4} & \text{if } y_0 \le y \le y_0 + 3 \text{ and } x_0 \le x \le x_0 + 4 - 4\frac{y-y_0}{3} \end{cases} \end{aligned}$$

A cross-section along the x-dimension of this function is plotted in Figure 4.7, and a plan view of the various regions is shown in Figure 4.8.

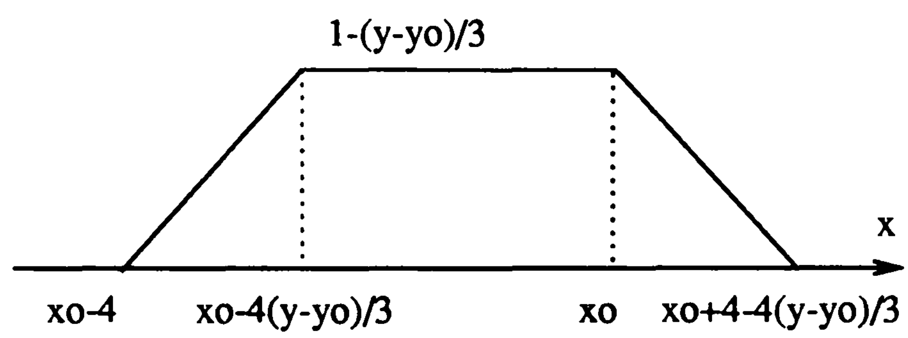

Figure 4.7: Cross-section of PMF for *side* 4.

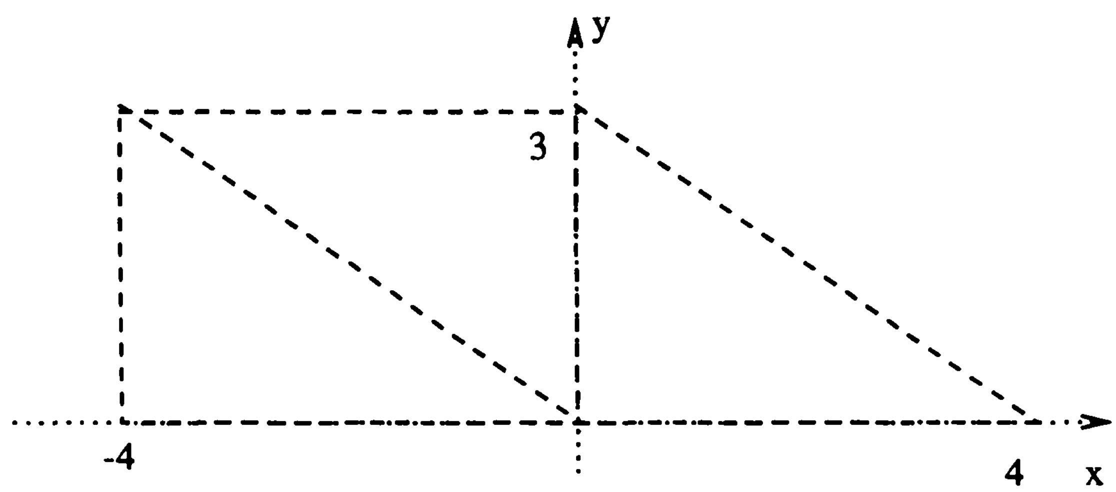

Figure 4.8: Plan view of PMF for *side* 4.

Similar calculations associating the sensed point with the other two sides can be performed. For the side of length 3 (*side* 3), the transformed model is shown in Figure 4.9 and the resulting transformed object model is

$$O'(x,y) = \begin{cases} 1 & \text{if } y_0 \leq y \leq y_0 + 4 \text{ and } x_0 + 3\frac{y-y_0}{4} \leq x \leq x_0 + 3 \\ 0 & \text{otherwise} \end{cases}$$

The feature uncertainty function is

$$F_{side\ 3}(x,y) = \begin{cases} U_{(0,3)}(x) & \text{if } y = 0 \\ 0 & \text{otherwise} \end{cases}$$

Now, $\Pi_{side\ 3}(x,y)$ can be calculated:

$$\Pi_{side\ 3}(x,y) \quad = \quad \int_0^3 1/3\, O'(x-\mu, y)\, d\mu$$

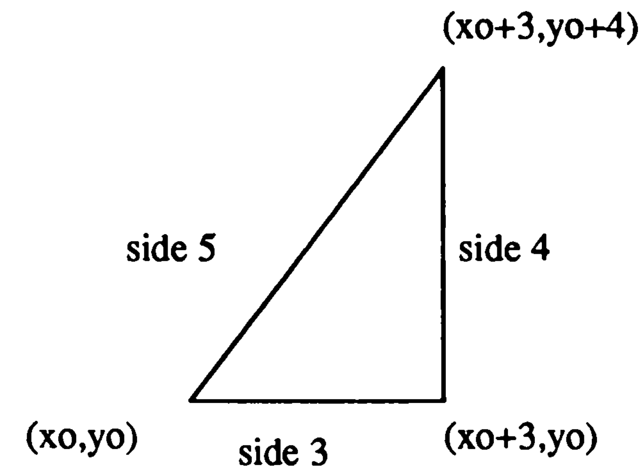

Figure 4.9: Object model for contact on *side* 3.

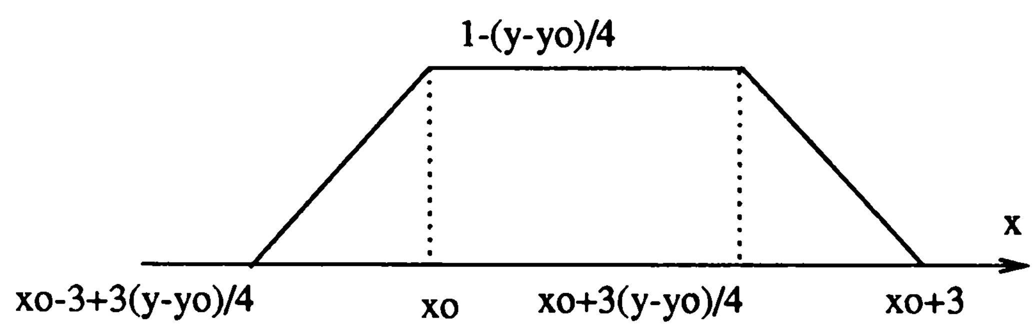

Figure 4.10: Cross-section of PMF for *side* 3.

$$= \begin{cases} 0 & \text{if } y \le y_0, \text{ or } y \ge y_0 + 4, \\ & \text{or } x \le x_0 - 3(1 - \frac{y-y_0}{4}), \text{ or } x \ge x_0 + 3 \\ 1 - \frac{y-y_0}{4} + \frac{x-x_0}{3} & \text{if } y_0 \le y \le y_0 + 4 \text{ and } x_0 - 3(1 - \frac{y-y_0}{4}) \le x \le x_0 \\ 1 - \frac{y-y_0}{4} & \text{if } y_0 \le y \le y_0 + 4 \text{ and } x_0 \le x \le x_0 + 3\frac{y-y_0}{4} \\ 1 - \frac{x-x_0}{3} & \text{if } y_0 \le y \le y_0 + 4 \text{ and } x_0 + 3\frac{y-y_0}{4} \le x \le x_0 + 3 \end{cases}$$

The x-cross-section of this function is plotted in Figure 4.10.

For the side of length 5 (*side* 5) the transformed object model is shown in Figure 4.11. For this case

$$O'(x,y) = \begin{cases} 1 & \text{if } y_0 \le y \le y_0 + 2.4 \text{ and } x_0 + 3.2\frac{y-y_0}{2.4} \le x \le x_0 + 5 - 1.8\frac{y-y_0}{2.4} \\ 0 & \text{otherwise} \end{cases}$$

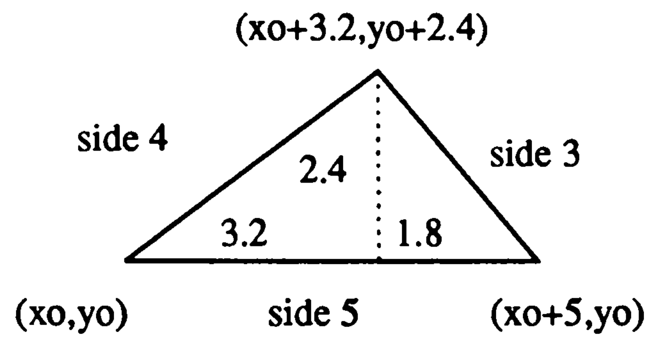

Figure 4.11: Object model for contact on *side* 5.

The feature uncertainty function is

$$F_{side\ 5}(x,y) = \begin{cases} U_{(0,5)}(x) & \text{if } y = 0 \\ 0 & \text{otherwise} \end{cases}$$

Now $\Pi_{side\ 5}(x,y)$ can be calculated:

$$\begin{aligned} \Pi_{side\ 5}(x,y) &= \int_0^5 1/5\, O'(x-\mu,y)\, d\mu \\ &= \begin{cases} 0 & \text{if } y \leq y_0, \text{ or } y \geq y_0 + 2.4, \\ & \text{or } x \leq x_0 - 5 + 3.2\frac{y-y_0}{2.4} \text{ or } x \geq x_0 + 5 - 1.8\frac{y-y_0}{2.4} \\ 1 + \frac{x-x_0}{5} - 3.2\frac{y-y_0}{12} & \text{if } y_0 \leq y \leq y_0 + 2.4 \\ & \text{and } x_0 - 5 + 3.2\frac{y-y_0}{2.4} \leq x \leq x_0 - 1.8\frac{y-y_0}{2.4} \\ 1 - \frac{y-y_0}{2.4} & \text{if } y_0 \leq y \leq y_0 + 2.4 \\ & \text{and } x_0 - 1.8\frac{y-y_0}{2.4} \leq x \leq x_0 + 3.2\frac{y-y_0}{2.4} \\ 1 - \frac{x-x_0}{5} - 1.8\frac{y-y_0}{12} & \text{if } y_0 \leq y \leq y_0 + 2.4 \\ & \text{and } x_0 + 3.2\frac{y-y_0}{2.4} \leq x \leq x_0 + 5 - 1.8\frac{y-y_0}{2.4} \end{cases} \end{aligned}$$

The x-cross-section of this function is plotted in Figure 4.12.

To obtain the total probability of the object being present at a particular location in the workspace, without any knowledge of which edge has been contacted, the functions obtained for the three sides must be combined. The total

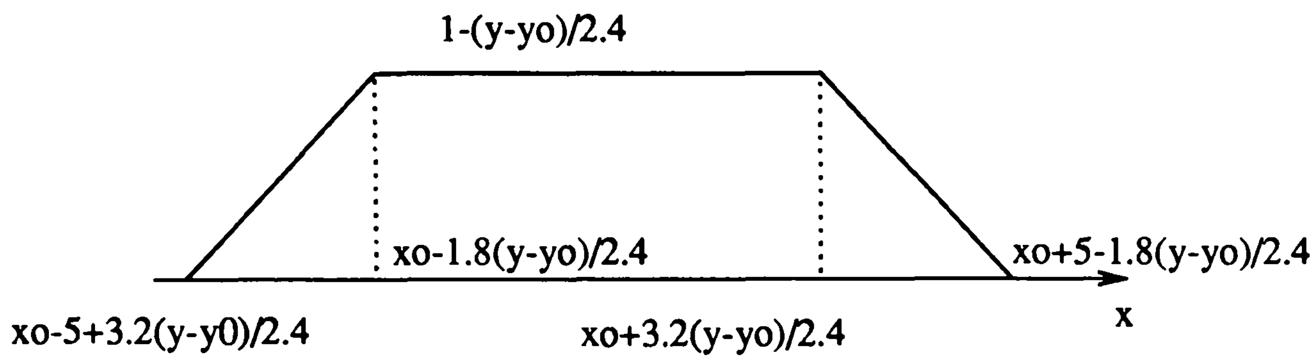

Figure 4.12: Probabilistic membership function for *side* 5.

probability is evaluated as follows:

$$\Pi_{total}(x,y) = P(side\ 3)\,\Pi_{side\ 3}(x,y) + P(side\ 4)\,\Pi_{side\ 4}(x,y) + P(side\ 5)\,\Pi_{side\ 5}(x,y)$$

To evaluate the probability of the sensed side having a particular identity (in the absence of any other information), we assume that the probability, $P(side\ n)$, is proportional to the length, n, of the side.

Thus the probability for each side is the ratio of the side length to the perimeter of the object. For the triangle considered in this example

$$P(side\ 3) = 1/4\,,\ P(side\ 4) = 1/3\,,\ P(side\ 5) = 5/12$$

This leads to the more complex PMF whose regions are plotted in Figure 4.13. Taking a sample cross-section along the line $y = 1$ (as indicated in Figure 4.13) results in the function plotted in Figure 4.14.

Using these methods, the likelihood of detecting the object at any point in space the be determined, without any knowledge of which edge was initially sensed. Clearly, knowledge of which edge had in fact been sensed would lead to a simpler formulation of the PMF.

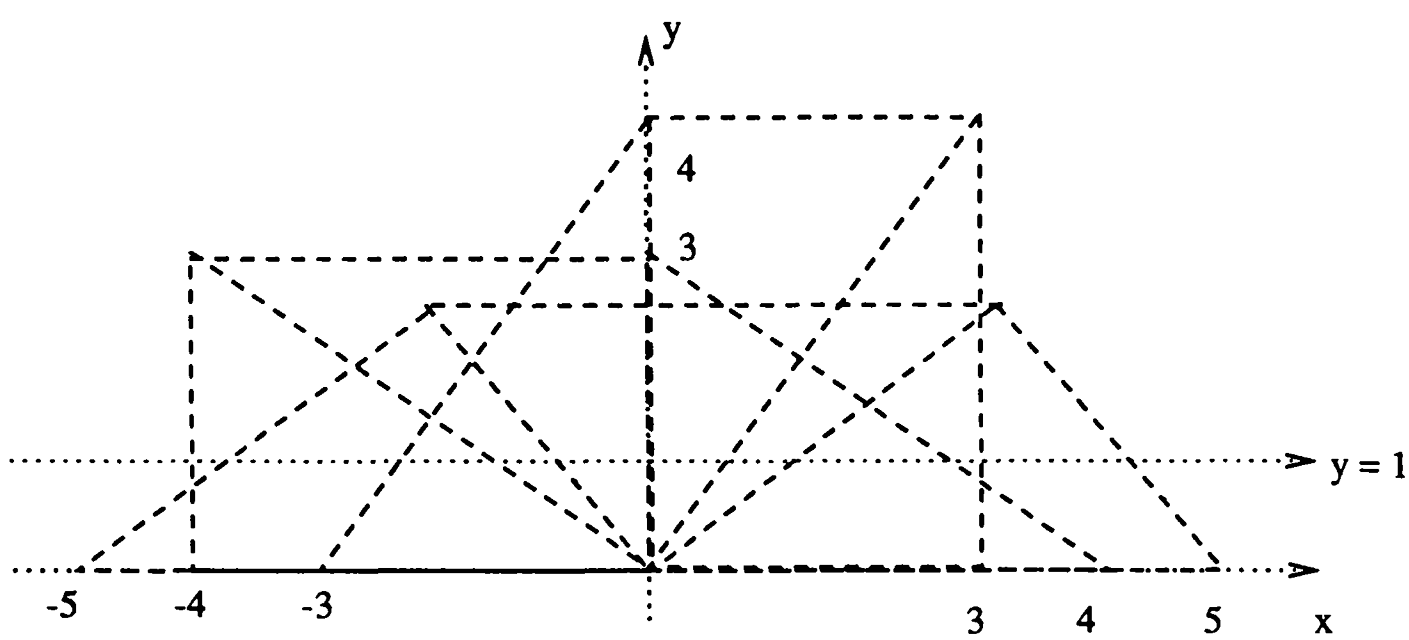

Figure 4.13: Plan view of resultant PMF.

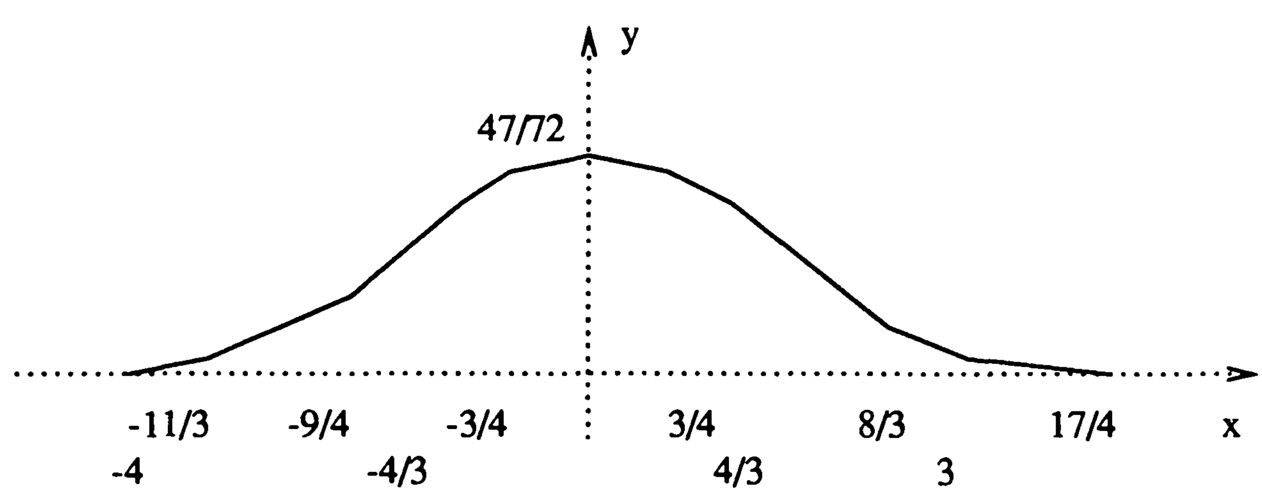

Figure 4.14: Cross-section of resultant PMF.

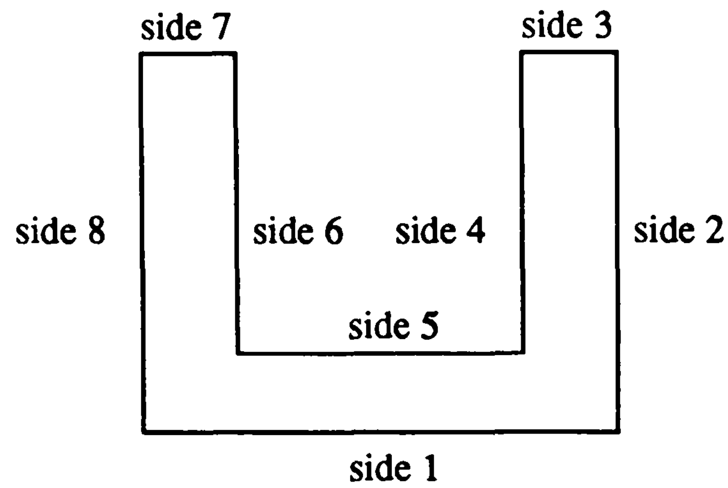

Figure 4.15: Object characteristic function.

4.2.3 Example 3: Automatic Generation for a Concave Object

This section considers the PMF for a more complex object. Rather than deriving analytically an expression for this function, automatically generated plots are displayed showing the PMFs for each contact side.

Consider the concave object shown in Figure 4.15. It has eight sides, and as there is no rotational symmetry for the object, each must be considered separately in generating the PMF.

The PMFs generated automatically for each side are shown in Figures 4.16 to 4.23. The plots use a grey-scale to indicate likelihood of detection, with black indicating certain detection (PMF value of one), and white indicating no prospect of detection (PMF value of zero). The overall PMF (obtained by obtaining a sum of the side PMFs each weighted according to the length of the corresponding side), is shown in Figure 4.24.

The figures demonstrate how the PMFs explicitly represent the shape of the

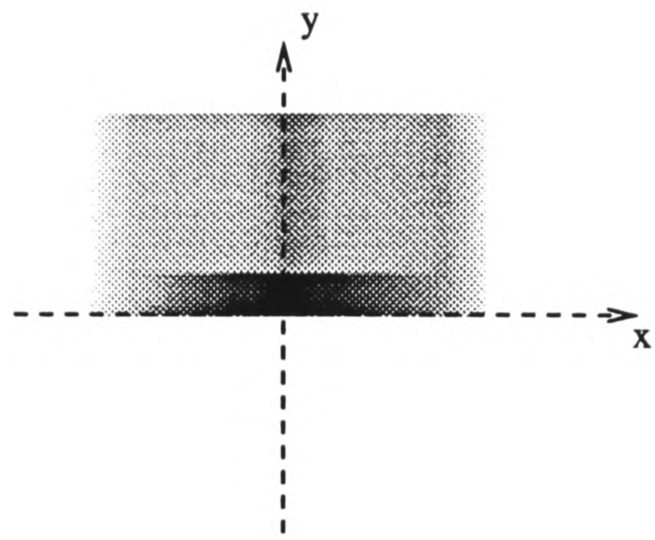

Figure 4.16: PMF for *side* 1.

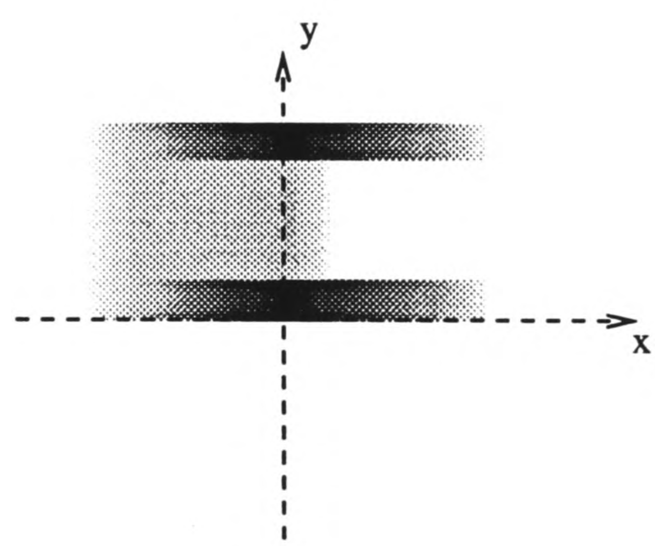

Figure 4.17: PMF for *side* 2.

object, blurred by uncertainty along the designated contact edge.

The automatic generation of the PMF is readily performed from a list of vertex coordinates representing the object model. As the objects become more complex, the amount of processing and data storage required to compute the PMF increase linearly with the number of sides. (ie. The algorithm used to generate the PMF autonmatically is only $O(n)$ complex with respect to the number of sides n.) It is shown in Section 7.4 that it is, in fact, not necessary to evaluate the PMF at all points in the sensing space, but only at a finite number of locations. This leads

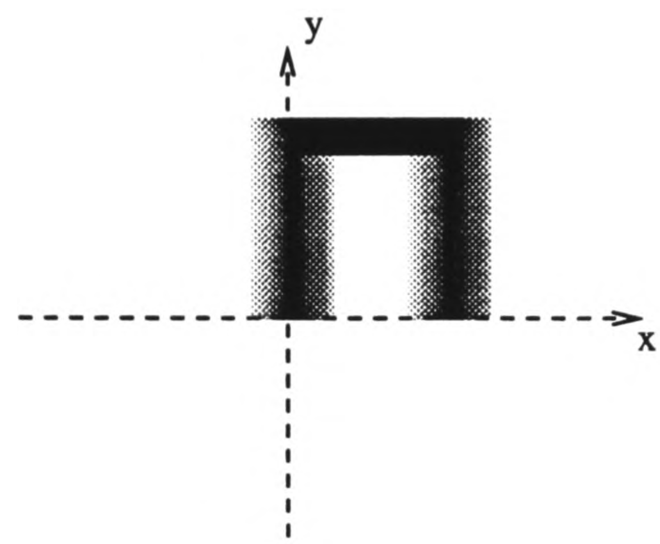

Figure 4.18: PMF for *side* 3.

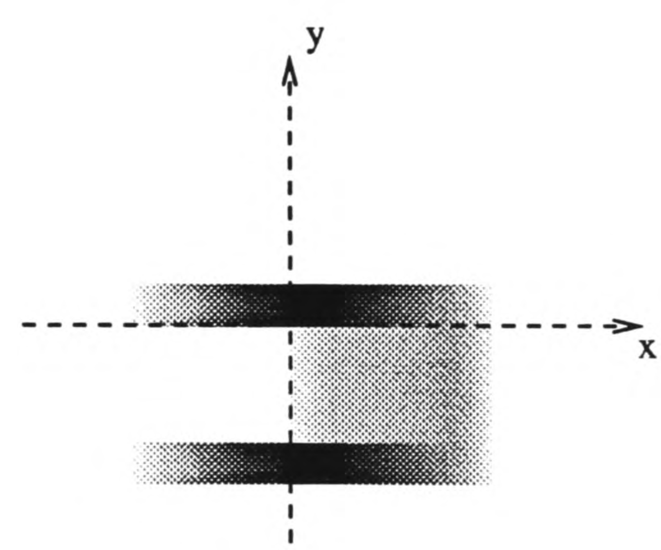

Figure 4.19: PMF for *side* 4.

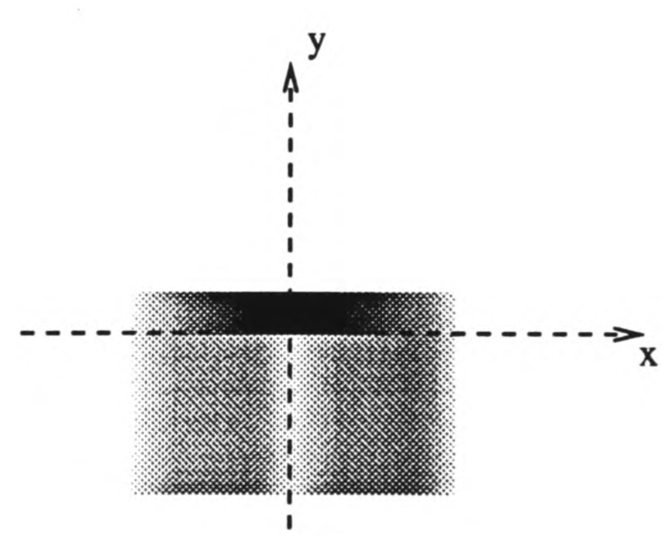

Figure 4.20: PMF for *side* 5.

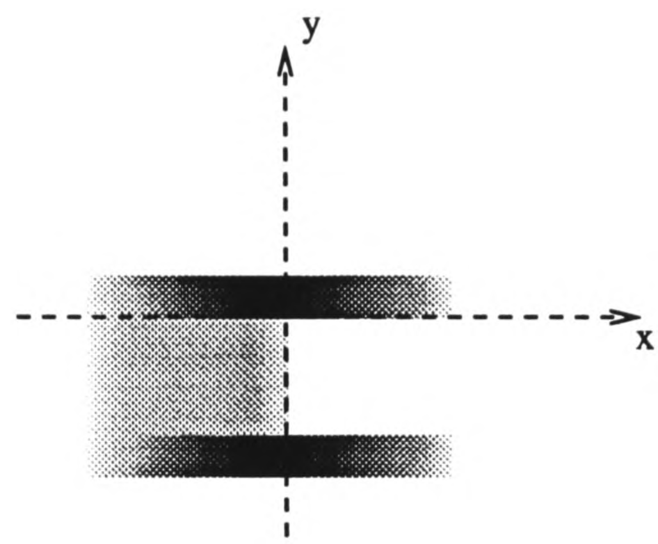

Figure 4.21: PMF for *side* 6.

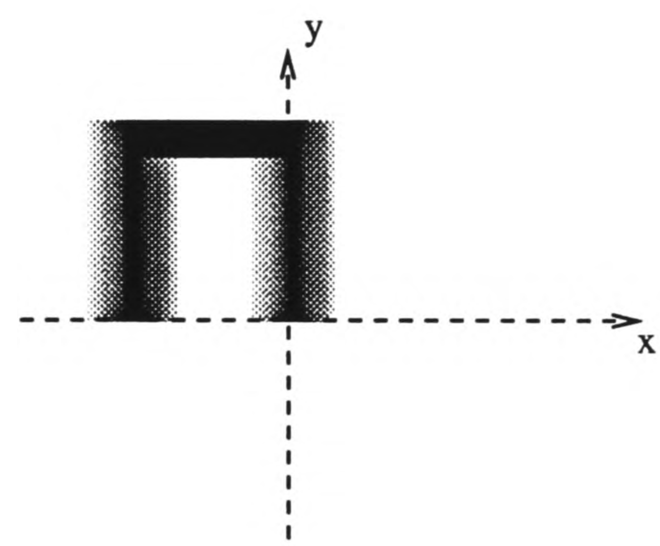

Figure 4.22: PMF for *side* 7.

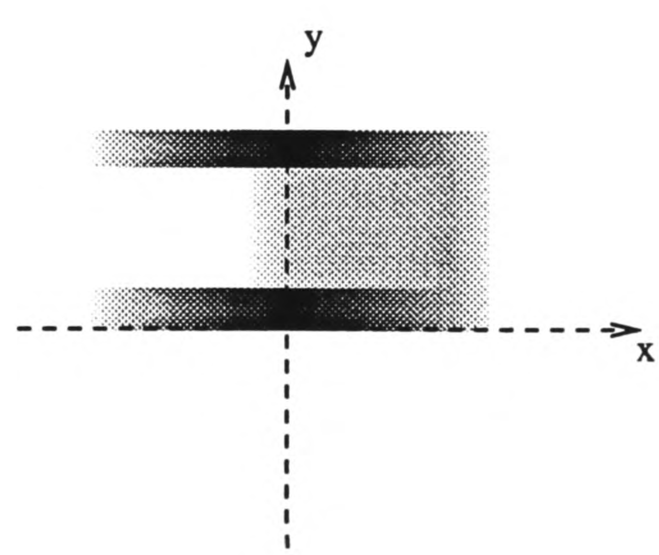

Figure 4.23: PMF for *side* 8.

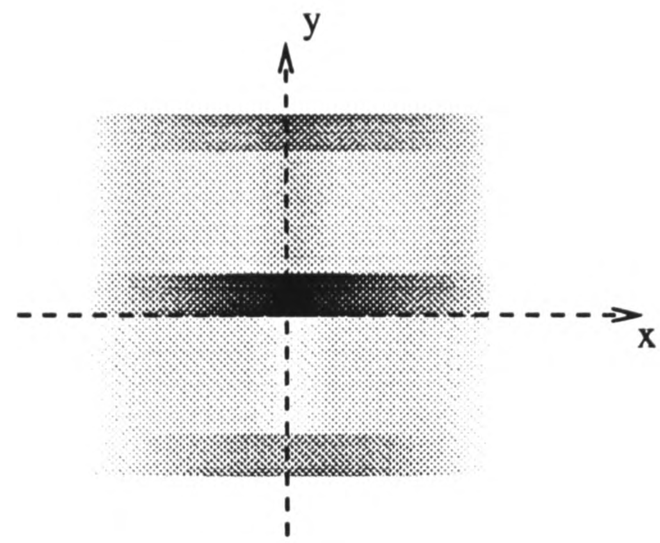

Figure 4.24: Resultant PMF.

to a significant reduction in processing time and data storage requirements, as is discussed in Chapter 7.

4.3 Updating PMFs

As has been shown in the previous section, the only data required to construct a PMF is a model of the sensed object and the location of a sensed object feature. However, as data from more sensed points becomes available, it must be used to update the PMF, so that the function accurately represents the knowledge of the workspace.

One appealing solution to this problem would be to generate a PMF for each sensed point and combine them. In practice however, this proves to be non-trivial, and requires knowledge of the object's features.

Consider the case of a square of side-length l whose edge has been sensed at two locations, separated by a distance d. It was seen in Subsection 4.2.1 that the resulting PMFs can be represented graphically as shown in Figure 4.25.

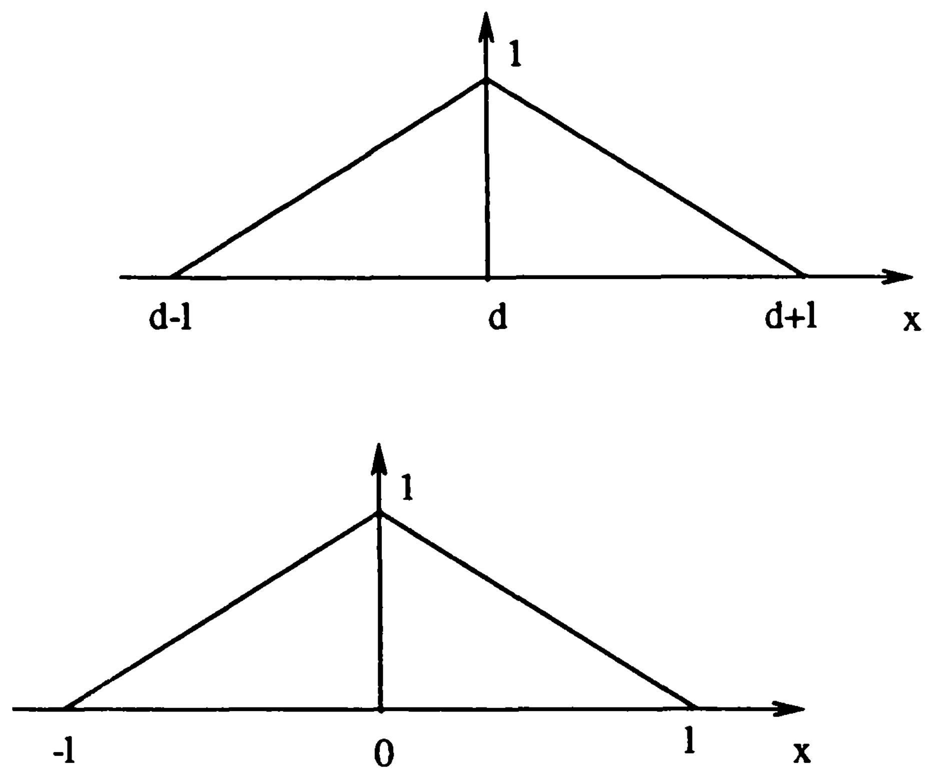

Figure 4.25: Probabilistic membership functions.

To determine the appropriate PMF which represents the information provided by both sensed points, the feature uncertainty functions for the two locations relative to the sensed point at (x_0, y_0) must be considered. (Figure 4.26.)

Interpreting these functions, it can be deduced that the combined uncertainty in the feature's position is given by the intersection of the two uncertainty ranges and can be represented by the uniform distribution $U_{(d,l)}(x)$, shown in Figure 4.27.

Convolving this function with the object model function gives the PMF shown in Figure 4.28.

This function cannot be obtained directly from the two single-point PMFs shown earlier. It is also necessary to reconstruct the feature uncertainty functions to take account of locations at which no feature is detected (and hence have no associated PMF although they constrain the object's position) and also observations of different features, such as interiors (which do not provide strong constraints on the object location by themselves, but do so in combination with

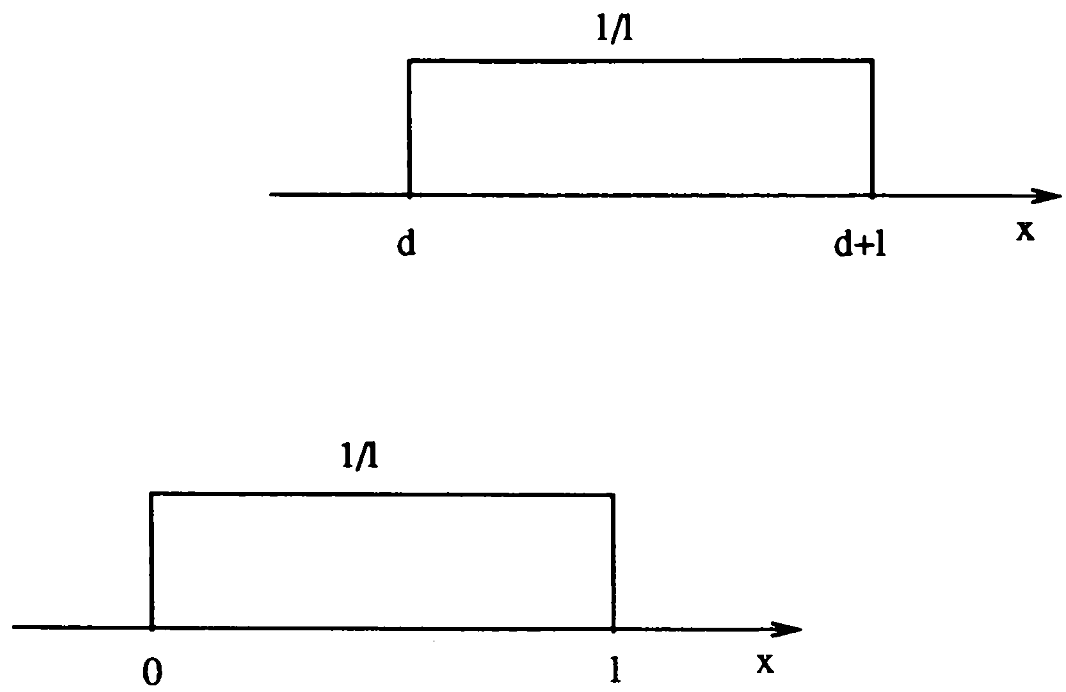

Figure 4.26: Feature uncertainty functions.

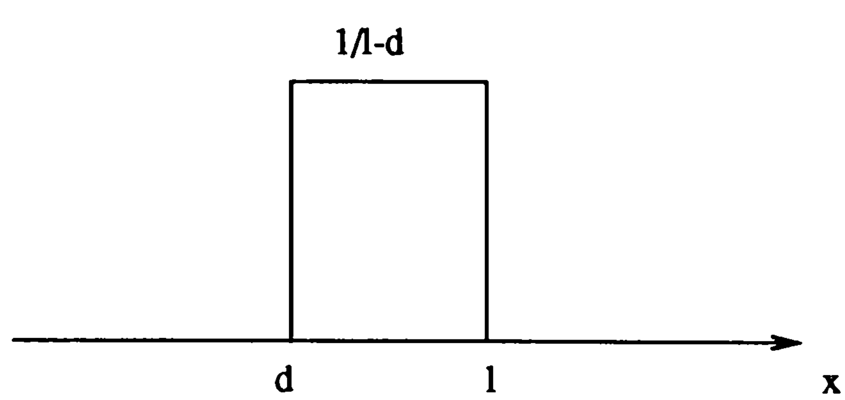

Figure 4.27: Updated feature uncertainty function.

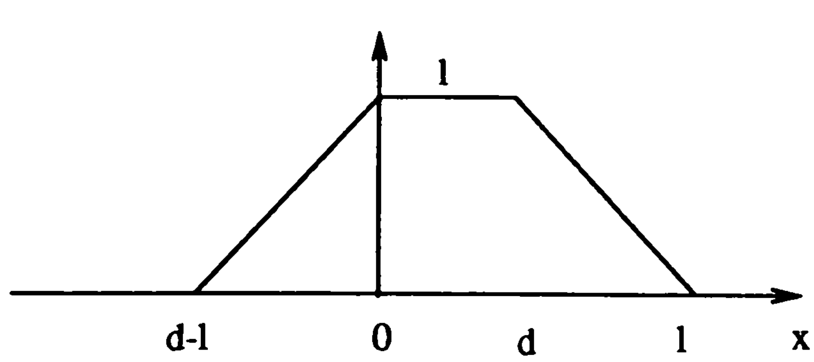

Figure 4.28: Updated probabilistic membership function.

a sensed edge).

A robust method is required for integrating information provided by further sensed points, to update the feature uncertainty function.

Consider again the example of sensing a square where the initial sensed point was an edge contact. Thus the feature uncertainty function is as shown in Figure 4.3. In being convolved with this function, the object model $O'(x-\mu, y)$ is integrated with respect to μ as μ varies in the (admissible) range $(0, l)$. Consider an arbitrary location in the sensing space (x_1, y_1). Now, unless $\Pi(x_1, y_1)$ is zero or one, then there is uncertainty as to whether or not the object is present at this location. What this means is that some admissible interpretations assigning the sensory data to the object model give rise to the object being present at (x_1, y_1) whilst other admissible interpretations do not. This is represented by the function $O'(x_1-\mu, y_1)$ taking the value zero for some subset of values of μ in $(0, l)$, and the value one for the remainder.

Once sensor data has been acquired at (x_1, y_1) and the actual value of the state at (x_1, y_1) determined, those values of μ which predicted the wrong state can be rejected. This leads to a smaller range of μ corresponding to reduced uncertainty as more sensory data is acquired, as would be expected. The admissible range of μ consists only of those values that are consistent with all of the sensory data obtained thus far. A uniform distribution on the remaining acceptable values of μ can be assumed, producing the new feature uncertainty function.

4.4 Effect of finite sensor size

All the analysis carried out thus far, has assumed that it is possible to obtain sensory information at a given point in the workspace. Due to the finite sensing area of all sensors, this is never true in practice, nor is it always desirable. Often by sensing over a finite area in a single action, more information can be acquired, enabling characteristic point features, such as corners, to be detected, which would not be possible with a point sensor. As corners are often salient features of an object set (as discussed in Section 5.4), they may play an important role in the sensor placement strategy. It will be clear that in these instances, uncertainty is reduced (and hence sensing power increased) as the sensor scope increases.

By sensing over a finite area, the sensing is effectively being performed at many adjacent points simultaneously. This gives rise to a different PMF than is obtained by sensing at a single point. It was seen in the previous section that sensor data from different sensor locations combine to alter the feature uncertainty function, F. It is to be expected that the feature uncertainty function due to sensing over a finite area is the same as that due to sensing at point locations spread over the same sensing area. No effect on the object model, O, or the sensor uncertainty function, S, arise due to the finite size of the sensor.[4] Hence the probabilistic

[4] It may be expected that altering a property of the sensor, such as its size, would have an effect on the function S. S, however, is the sensor uncertainty function, and models the accuracy of (or the confidence in) the sensor data. F, on the other hand, is the feature uncertainty function, and represents the uncertainty in the location of the model due to the properties of the sensed feature and the manner in which it is sensed.

membership function, Π, can be computed as before

$$\Pi = O' * F * S$$

where F is the only quantity that has altered. This is demonstrated in the example given at the end of this section.

Finite sensor size also affects the manner in which PMFs can be interpreted for subsequent sensing actions. As the PMF represents the likelihood of detecting the object at each point in the sensing space, a sensor of finite extent will effectively sense at several points concurrently, each of which may have a different value. Careful consideration must be given to the true expectation of sensing the object at a given location, given that the sensor will actually return data from a finite area surrounding the point. Often the true expectation will simply be the maximum of the PMF values within the sensor's scope however this may not always be the case.

To address this issue thoroughly, it is again necessary to go back to the feature uncertainty functions to determine which range of uncertainty values gives rise to detection at each point within the scope of the sensor. The union of these ranges should then be convolved with the object model to give rise to a new probabilistic sensing function representing the likelihood of sensing the object when the sensor is placed at a given location, and hence its scope takes in a known region.

Often the PMF will be linearly increasing (or decreasing) over a region with distance from a sensed point. (See, for example, Figure 4.31.) In this case, the range of uncertainty values giving rise to the point of highest probability

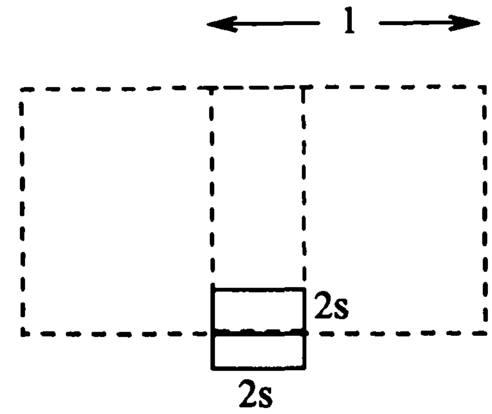

Figure 4.29: Sensor data and possible object locations.

within the sensed region, contains the uncertainty range for all the points of lower probability. In this case, the likelihood of sensing the object within the scope of the sensor, is indeed equal to the maximum value of the PMF within this region.

Example

Consider a sensor with a square sensing area of small, but finite, extent.

It is assumed that the edge (without an endpoint) of an object of length l has been sensed by a sensor of dimensions $2s$ by $2s$. The sensor output is as shown in Figure 4.29, together with possible locations of the object. The range of possible locations of the sensor along the edge is only $(s, l-s)$, as if the middle of the sensor was located within s of an endpoint of the edge, a corner would have been detected. This leads to the feature uncertainty function along the dimension parallel to the sensed edge as shown in Figure 4.30. Assuming no sensor uncertainty ($S(x,y) = \delta(x,y)$), the resulting PMF along this dimension is as plotted in Figure 4.31.

Note that the PMF obtained is equivalent to that obtained by sensing with a

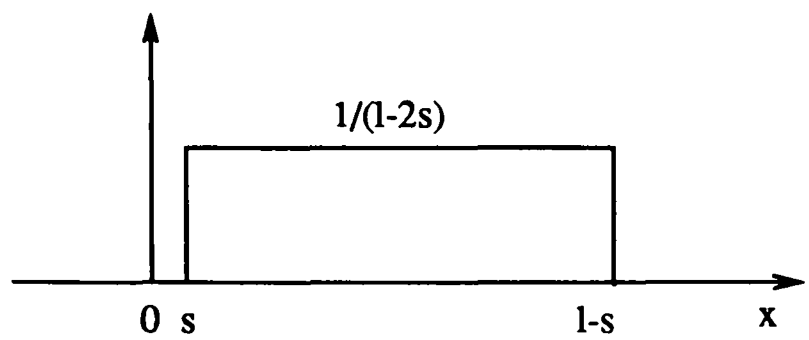

Figure 4.30: Feature uncertainty function.

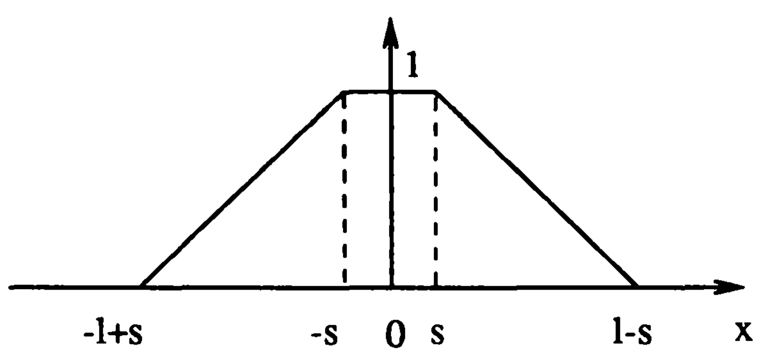

Figure 4.31: Probabilistic membership function.

point sensor at points displaced by $\pm s$ from the midpoint of the sensor.

4.5 Feature PMFs

The PMFs discussed thus far have represented the probability of detecting the sensed object at a given location in the workspace. If the sensor is capable of detecting features (such as edges or corners) then it may be useful to represent the probability of detecting a given identifiable feature at a given location in the workspace. In this way it is possible to generate PMFs for each feature of interest and use this information in evaluating potential sensing locations.

Because features such as edges and corners represent point discontinuities of the object's characteristic function, they exist only at point locations in space and have no finite extent. The possibility of detecting such "point" features using

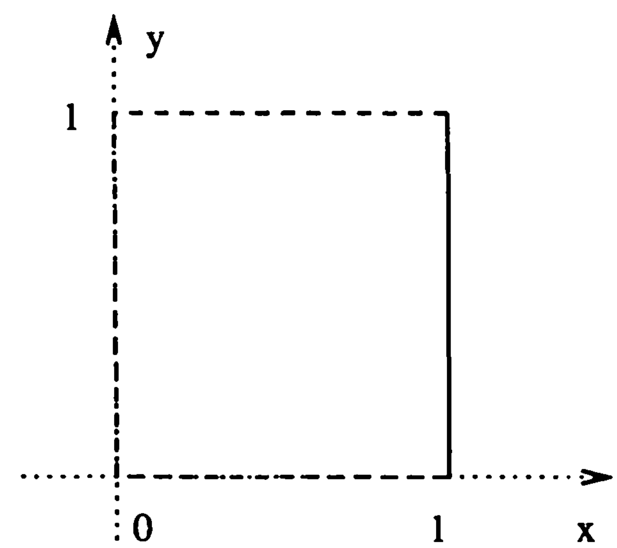

Figure 4.32: Characteristic function of an edge.

a sensor which can only sense at points in space is zero everywhere. In practice, however, most sensors have a finite field of view, and further, the ability to detect edge or corner features implies a finite scope. Hence, although the PMF for an edge or corner may be zero at all points the probability of detecting these features over a given range (in this case, the scope of the sensor) may be non-zero (much like a pdf). It is clear that the probability of detecting these features will vary with the scope of the sensor used. Their effect on sensing strategy will be more significant for sensors of relatively large scope, and less significant for more "point-like" sensors.

The method of generation of PMFs for each feature of interest is as for the object PMFs described in Section 4.2. A model of each feature is developed to fill the role of the object characteristic function used previously. In this case the functions may have zero area, taking non-zero values at only a finite number of locations.

Consider, for example, the characteristic function for the edge of a square shown in Figure 4.32.

The feature characteristic function can be described as

$$O(x,y) = \begin{cases} 1 & \text{if } x = l \text{ and } 0 < y < l \\ 0 & \text{otherwise} \end{cases}$$

The PMF can be obtained in the standard manner.

$$\Pi(x,y) = O'(x,y) * F(x,y) * S(x,y)$$

If this edge is not parallel to the direction of uncertainty (as defined by the slope of the first sensed edge) then the PMF will be zero everywhere, as previously explained. However, because of the capabilities of a sensor of finite scope to sense over a region (and hence an infinite number of points) concurrently, the likelihood of sensing the object with a given sensing action, need not be zero.

Consider a sensor of finite scope, with dimension s in the direction of uncertainty, attempting to detect an edge which is not parallel to the direction of uncertainty. If the uncertainty range is $[u, v]$, then the location of the edge is unknown within this range. If sensing for the edge is performed at a location such that the whole scope of the sensor includes only points which are feasible locations for the edge (ie. the scope of the sensor lies completely within the uncertainty range of the edge), the probability of detection will be $\frac{s}{v-u}$. As anticipated, the probability is proportional to the sensor scope, and hence would be zero for a point sensor ($s = 0$).

On the other hand, if the direction of uncertainty is parallel to the edge of interest, as would be the case if we were interested in the probability of detecting again the first sensed edge, then the PMF may take non-zero values. For the case of the first edge, the PMF would simply correspond to the cross-section of the object PMF along the line $y = 0$. The effects of finite sensor size in this case would be as discussed in Section 4.4.

4.6 Path sensing

All the PMFs considered so far have measured the likelihood of the presence of the object at a given location in the sensing space. This information is most desirable if a choice must be made between alternative locations at which to sense. Some sensors, however, do not sense at a specified location, but can only be directed to sense along a specified path. This is true of many non-contact sensors including sonar, radar, vision, and infra-red light. It may also be true of contact sensors, including tactile sensing, which may be able to track along a given path until an object is encountered.

The major difference between this situation and that described previously, is the relative position of the sensor to the objects of interest. To enable "location" sensing or "probing", it is implicitly assumed that the path along which the sensing occurs is not parallel to the plane in which the objects lie. Hence there is only one point of intersection between the sensing ray and the workspace which enables sensing to be performed at this location. With path sensing, it is assumed that the sensing ray lies in the same plane as the objects to be sensed. Hence, the intersection between the ray and the workspace includes the whole path of the ray, and all points along this path are sensed concurrently.

In practice data will usually be returned from the point of intersection between ray and object that is closest to the ray's source. For this reason, it is impossible to sense within the interior of an object, and only boundary features can be detected. Thus the whole of the sensing strategy will need to be formulated on the basis of detecting object edges and corners. The advantage of sensing along a path is that there will be non-zero probabilities for detecting point features (such as edges and corners) as sensing is actually being performed over a range of locations concurrently (just as if using a location sensor with a finite scope). Hence sensing can be performed for point features lying within a certain uncertainty range.

In three dimensional problems, it is almost always only possible to detect the boundaries of a sensed object. The most natural way to model these situations is by considering the path sensing approach. For this reason, the approach outlined in this section, may have greater application to the generalization of the active sensing methods to three dimensional problems. (See Section 8.2.)

In generating probabilistic membership functions for path sensing, it is necessary to define carefully the sensing action – in this case the ray along which the sensing will be performed. The sensing ray will be defined by the polar coordinates, (r, θ), describing the perpendicular distance of the ray from the origin of the workspace, and the angle of inclination of the ray relative to the workspace Cartesian coordinate system respectively. (See Figure 4.33.)

It is necessary to define the direction of the ray to avoid ambiguity, as we will be interested in determining the point of intersection of the ray with the sensed object, closest to the ray's source. For this purpose r is defined to have

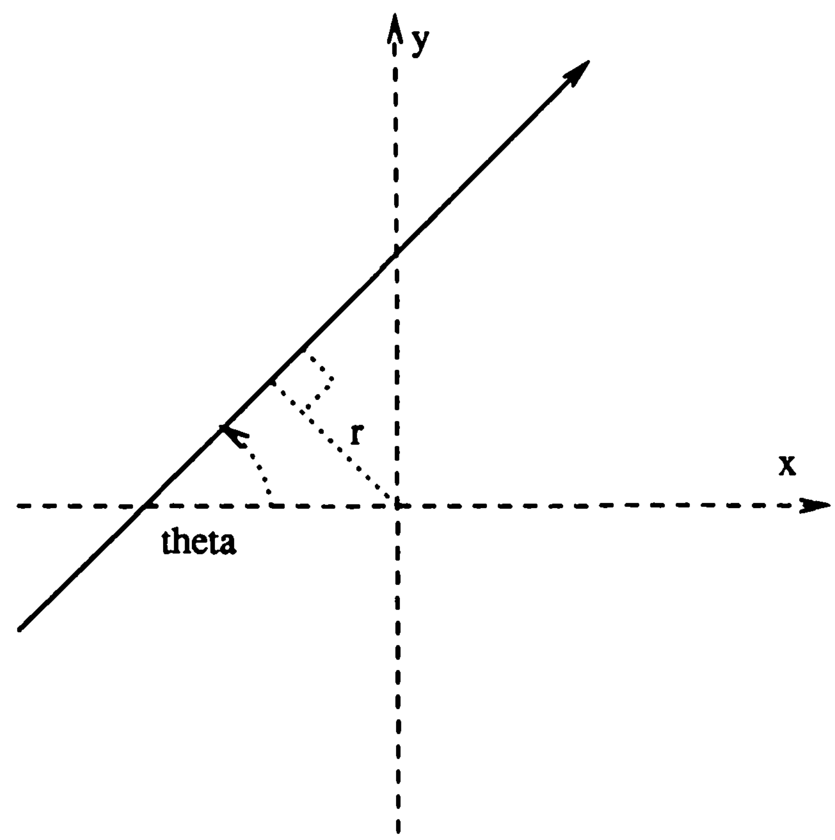

Figure 4.33: Definition of polar coordinates.

a positive or negative sign depending on whether the origin lies to the right or left of the directed ray respectively when viewed along the direction of the ray, and θ is defined to lie in the range from 0 to 2π indicating the ray's direction. Thus the rays defined by (r, θ) and $(-r, \theta + \pi)$ will be collinear but have different orientations.

In determining the likelihood of sensing an object along the path defined by (r, θ), it is natural to begin with a model of the object and a range of uncertainties, as used previously for the location sensing case. In this case feature characteristic functions for edges and corners will be relevant in the place of object characteristic functions, as it will be these features that may be sensed. The feature characteristic functions and uncertainty ranges may take the same form as was developed previously, however, there is no benefit to be gained by convolving them to obtain the likelihood of the object being present at each point in space. Combining such point likelihoods along the sensing path proves to be a non-trivial task, and requires knowledge of the geometry of the sensed object. Instead, it is possible to work directly with the feature models and uncertainty ranges to determine the PMF in polar coordinates. An algorithm for performing these calculations is presented in Appendix C. This algorithm was used in the location sensing implementation described in Section 7.3. As the object models become more complex, the processing and data storage required to compute the PMF using this method increase with the square of the number of model sides. (ie. Given n sides, the complexity of the computation is $O(n^2)$. This is in comparison with the algorithm used to determine the PMF for location sensing which is only $O(n)$ complex.)

Once a measure of the likelihood of sensing each identifiable feature has been obtained as a function of the ray parameters (r, θ), this PMF can be used in the same manner as those generated for location sensing. Considerations of the effects of finite sensor size on path sensing PMFs are again relevant for the same reasons as discussed in Section 4.4.

Chapter 5

Utility and Discrimination Functions

5.1 Nature of utility functions

As was discussed in Chapter 3, one of the vital elements of decision theory is a utility function which enables the merits of different actions to be compared for a particular state of the environment. Because of their importance to decision theory, utility functions have been studied extensively. This has led to the development of techniques which can be applied to specific situations to develop appropriate utility functions. Some of these methods are explained in Berger [6], De Groot [15], Blackwell and Girshick [7], Wald [50] and Jones [36]. Despite the existence of such methods, there is still considerable subjectivity involved in the selection of a utility function.

For the tasks of object recognition and localization, there is no obvious metric that can be applied to the outcome of a particular sensing action. To enable a quantitative analysis to be performed, it is necessary to develop an ordering on the desirability of particular outcomes, and attempt to discern a relationship between the action and state and their desirability. This relationship can then be used to determine the utility of arbitrary combinations of action and state. This is the approach adopted in this chapter.

Our approach is based on the observation that a good sensing location is, or rather was in retrospect, one which enabled the object to be recognized or localized. A poor location was one which (in retrospect) provided no new information. Thus utility functions are proposed which provide some measure of the ability of the sensing location to discriminate between possible object identities or locations. What is really required is some measure of the distance remaining (perhaps measured in the minimum number of sensing points required) to the completion of the task. Attempts at developing this measure are described in the next section.

Once the utility functions have been obtained, the Bayes decision principle requires that they be weighted according to the prior distribution of the states to determine the optimal action. In this section, the weighted utility value (usually termed the Bayes risk when dealing with loss as opposed to utility) will be termed the discrimination function.[1] This function depends only on the alternative sensing actions and can be used to discriminate between them (hence its name). The optimal sensing action is that action which maximizes the discrimination function.

[1] Hager [27] has termed this function the payoff function.

It will be assumed throughout this chapter that the state of the environment can be described by a characteristic function on the workspace, representing the presence or absence of the object at each location. The prior knowledge of the characteristic function is the probabilistic membership function described in Chapter 4.

The next section presents several examples of the generation of utility functions (and the resulting discrimination functions) to demonstrate the proposed approach. We argue that the proposed utility criteria lead to good sensing strategies, as demonstrated by the implementations described in Chapter 7.

In Section 5.4 salient features and their role in object recognition are discussed. It is shown that well-chosen utility functions will implicitly favour the detection of salient features, and hence that by employing a decision theoretic approach to object recognition, it is not necessary to deal with salient features as a special class.

Finally Section 5.5 considers the Artificial Intelligence technique of "look-ahead search" and its application to the problem of determining the optimal sensing action.

5.2 Examples

Several examples of utility functions are presented in this section, demonstrating how they may be constructed in practice.

The first five examples deal with the object recognition problem in increasing levels of complexity. It is assumed in Examples 5.2.1 and 5.2.2 that the object recognition task is to be performed between two objects on the basis of binary contact data from a sensor. In Examples 5.2.3 and 5.2.4, the object set is generalized to contain n objects. Finally in Example 5.2.5, a utility function is proposed for the situation in which a sensor can detect and identify a set of k features, and the system can utilize this data to recognize the sensed object from a known set of n objects.

In Example 5.2.6, the localization problem is addressed for a known object which must be localized on the basis of binary contact data.

All the utility functions proposed here are applicable irrespective of the sensing action available or the coordinate system in which the probabilistic membership functions are described. They are suitable for both location sensing and path sensing systems.

5.2.1 The two-object set (Case 1)

Consider the task of identifying the sensed object from a set of two objects, $\{o_1, o_2\}$. The utility function is a function of two variables: the sensing location, (x, y) (the action); and whether or not the object is present at the sensing location (the state or outcome). It will be assumed that the sensor used returns only binary contact data from the sensed object and hence that the state can be described by a two-valued function: $detection = 0$ (object not present at the sensing location) and $detection = 1$ (object present).

If a location (x_0, y_0) exists at which detection of the object will enable its identity to be deduced as o_1, then the utility function should have a high value for (x_0, y_0) and $detection = 1$. Similarly, if a location (x_1, y_1) exists at which detection of the object implies its identity is not o_1 then the utility function should also be high for (x_1, y_1) and $detection = 1$. These two situations represent extreme values (ie. zero and one) of the function

$$P(object = o_1 | detection = 1)$$

As it may be assumed (in the absence of any other information) that

$$P(object = o_1) = P(object = o_2) = 1/2$$

detection of the object at a location where

$$P(object = o_1 | detection = 1) = 1/2$$

suggests, in retrospect, a poor choice of sensing location.

Failure to detect an object can be just as important and useful as detecting the object. If the object is not detected at a location where $P(object = o_1 | detection = 0)$ has an extreme value, then this information can be used to deduce the object's identity.

In the two object case, it is only necessary to test the hypothesis that the object's identity is o_1, as rejection of this hypothesis implies that the object is o_2. As

$$P(object = o_1 | state) = 1 - P(object = o_2 | state)$$

then extreme values of $P(object = o_2 | state)$ correspond to extreme values of $P(object = o_1 | state)$.

Consider the following as a useful utility function.

$$U(state,(x,y)) = |2\,P(o_1|state)(x,y) - 1| \tag{5.1}$$

This function attains its maximum value of 1 when $P(o_1|state) = 0$ or 1 and attains its minimum value of 0 when $P(o_1|state) = 1/2$ (no information).

To evaluate the function $P(o_1|state)$ we use Bayes' Theorem.

$$\begin{aligned} P(o_1|state) &= \frac{P(state|o_1)\,P(o_1)}{P(state)}. \\ &= \frac{P(state|o_1)\,P(o_1)}{P(state|o_1)\,P(o_1) + P(state|o_2)\,P(o_2)} \end{aligned}$$

The functions $P(detection = 1|o_1)$ and $P(detection = 1|o_2)$ are simply the PMFs (Π) for the objects o_1 and o_2 as discussed in Chapter 4. As

$$P(detection = 0) = 1 - P(detection = 1)$$

the functions $P(detection = 0|o_1)$ and $P(detection = 0|o_2)$ are given by the functions $(1 - \Pi)$ for o_1 and o_2 respectively.

Thus the value of the utility function, $U(x,y)$, can be readily calculated for each point in the workspace using the probabilistic membership function techniques described in Chapter 4.

The utility function by itself, however, does not contain sufficient information to enable the optimal sensing location to be determined. Applying the Bayes decision principle, it is also necessary to take into account any prior information regarding the distribution of states. The desired location is the one which maximizes the discrimination function,

$$D(x,y) = U(detection = 1,(x,y))\,P(detection = 1)$$

$$+ U(detection = 0, (x,y))\, P(detection = 0)$$

By expanding $P(state)$ in terms of conditional probabilities and using Equation (5.1), the discrimination function can be expanded and simplified (see Appendix B.1) to obtain the result

$$D(x,y) = |\Pi_{o_1} P(o_1) - \Pi_{o_2} P(o_2)| + |(1 - \Pi_{o_1}) P(o_1) - (1 - \Pi_{o_2}) P(o_2)| \quad (5.2)$$

When $P(o_1) = P(o_2) = 1/2$, as will be the case for the first sensed point (in the absence of any other information)

$$D = |\Pi_{o_1}(x,y) - \Pi_{o_2}(x,y)|$$

Hence for this task, the first guided sensing location should be the location at which the difference between the probabilistic membership functions is greatest. Subsequent sensing points can be determined by evaluating and maximizing Equation (5.2) over all potential sensing locations, for updated values of $P(o_1)$ and $P(o_2)$.

5.2.2 The two-object set (Case 2)

This example deals with the same problem described in the previous subsection but proposes an alternative utility function. This is intended to emphasize the subjectivity of the choice involved in selecting an appropriate utility function, as both will be seen to be based on "reasonable" criteria, however they will give rise to different discrimination functions, and hence advise different sensing strategies.

Again the task will be to identify the sensed object from a set of two objects, $\{o_1, o_2\}$, on the basis of binary contact data from the sensed object. The state can again be described by a two-valued function: *detection* $= 0$ (object not present at the sensing location) and *detection* $= 1$ (object present).

Consider the following function as a utility function.

$$\begin{aligned} U(state,(x,y)) &= |P(o_1|state)(x,y) - P(o_1)| + |P(o_2|state)(x,y) - P(o_2)| \\ &= 2\,|P(o_1|state)(x,y) - P(o_1)| \qquad (5.3) \end{aligned}$$

This function attains its maximum value when the *a posteriori* expectation of the object's identity being o_1 ($P(o_1|state)$) differs greatest from the *a priori* expectation ($P(o_1)$), and attains its minimum value of zero when the expectations are the same (no new information).

By expanding Equation (5.3) using Bayes' Rule and weighting the function over the prior expectation on the states, the discrimination function can be expanded and simplified (see Appendix B.2) to obtain the result

$$D(x,y) \propto |\Pi_{o_1}(x,y) - \Pi_{o_2}(x,y)| \qquad (5.4)$$

Hence for this task, the optimal sensing location is the location at which the difference between the probabilistic membership functions is greatest. (Hence the same first sensing location is proposed by this utility function as for the function discussed in the previous section.) This location is determined easily and corresponds with an intuitive sense of the characteristics of the optimal sensing location.

If we sense at a location where there is a high probability of detecting the object if it is of a given identity, but a low probability if it is of the other identity, we can expect the outcome to influence our expectations significantly. Clearly, if the object is detected, our expectation of it being the first object will be strengthened at the expense of the second object. Conversely, non-detection will favour our expectation of the second object.

On the other hand, we can expect to acquire little information at locations where we have a similar expectation of detection for both objects. There is no point in sensing at a location where detection is assured for both objects, as the expectations will not be altered by the sensory data. The same will be true at locations where non-detection is assured.

5.2.3 The n-object set (Case 1)

Consider the task of identifying a sensed object from a set of n candidates $\{o_1, o_2, ..., o_n\}$. The utility function that is sought will be a function of the prospective sensing location and the (unknown) state at that location. An appropriate function is one which will return a high value for those values of location and state, where the detection of the given state at the given location is sufficient to deduce the identity of the sensed object. More formally, high values of $U(state, (x, y))$ would occur at locations where $P(o_k|state)$ was high for one of the objects, o_k.

Pursuing the same reasoning as was adopted in Example 5.2.1, a worthwhile sensing location (in retrospect) is one which gives rise to *a posteriori* expectations which differ significantly from the initial "no information" probability values. For the n-object identification problem, the initial expectations of the objects may be assumed (in the absence of any other information) to be

$$P(o_1) = P(o_2) = ... = P(o_n) = 1/n$$

This suggests the following utility function

$$U(state,(x,y)) = \sum_i |n\,P(o_i|state) - 1|$$

This function attains its maximum value of $2n-2$ when $P(o_i) = 1$ for some object o_i, and (hence) $P(o_j) = 0$ for all the other $n-1$ objects in the set. It attains its minimum value of zero, when $P(o_1) = P(o_2) = ... = P(o_n) = 1/n$

Evaluating the discrimination function for this utility function by considering its expectation over all the possible states leads to the result (see Appendix B.3)

$$\begin{aligned} D &= \sum_i |\sum_j \left(\Pi_{o_i} P(o_i) - \Pi_{o_j} P(o_j)\right)| \\ &\quad + \sum_i |\sum_j \left((1-\Pi_{o_i}) P(o_i) - (1-\Pi_{o_j}) P(o_j)\right)| \end{aligned} \tag{5.5}$$

This result is very similar to that obtained in Equation (5.2) for the two object case. Indeed the form of Equation (5.2) can be readily obtained from this result by setting $n = 2$. (See Appendix B.3.)

For the case when $P(o_1) = P(o_2) = ... = P(o_n) = 1/n$ (as may occur when the first decision is to be made), it is shown in Appendix B.3 that Equation (5.5) simplifies to

$$D = 2/n \sum_i |\sum_j \left(\Pi_{o_i} - \Pi_{o_j}\right)|$$

Thus the optimal sensing location in this case is the location at which a sum of the differences between object PMFs is maximized. This result has the same intuitive appeal as those obtained in the previous examples.

5.2.4 The n-object set (Case 2)

This example deals with the same problem described in the previous subsection but proposes an alternative utility function.

The proposed function is

$$U(state, (x, y)) = \sum_i |P(o_i|state) - P(o_i)| \tag{5.6}$$

where $P(o_i)$ represents the current expectation of the sensed object's identity being o_i. This function assigns a high value to sensing situations which drastically alter the expectations of the object's identity, and no value to those situations which do not alter the expectations (as such locations provide no new information). This is the same reasoning which led to the utility function proposed in Example 5.2.2 for the two-object problem.

Repeating the analysis performed for the previous examples to determine the discrimination function corresponding to Equation (5.6) gives (see Appendix B.4)

$$D = 2 \sum_i \left(P(o_i) | \sum_j \left(P(o_j) \left(\Pi_{o_i} - \Pi_{o_j} \right) \right) | \right) \tag{5.7}$$

As all the probabilities necessary to evaluate this function are well-known, the discrimination function can be evaluated at all possible sensing locations and the location which maximizes its value chosen as the best next sensing location.

For the case when $n = 2$ this result simplifies to that obtained in Equation (5.4) for the two object case. (See Appendix B.4.)

5.2.5 Multiple object states

Consider the n-object recognition problem described by the previous example, with the added complexity that there are more than two states which may be returned by the sensor. It will be assumed that there are k different features that the sensor can identify uniquely, each of which representing a possible sensing outcome at each location. These will in turn be represented by a set of k possible sensing states.

The same utility function can be employed in this case as in Example 5.2.4:

$$U(state,(x,y)) = \sum_i |P(o_i|state) - P(o_i)| \tag{5.8}$$

In this case however, there are several states over which to average the utility function in order to determine the discrimination function. In this event

$$D(x,y) = \sum_k (U(state_k,(x,y))\,P(state_k)) \tag{5.9}$$

Substituting Equation (5.8) into Equation (5.9) gives (see Appendix B.5)

$$D = \sum_k \sum_i \left(P(o_i) | \sum_j (P(o_j)\,(P(state_k|o_i) - P(state_k|o_j)))\,| \right) \tag{5.10}$$

The conditional probabilities $P(state_k|o_i)$ are functions of the sensing action, and represent the likelihood of detecting the feature k at any location (or along any path) given that the identity of the object is o_i. These functions are the feature probabilistic membership functions for feature k and object o_i described in Section 4.5. Hence the discrimination function is readily evaluated, and is a function of the difference between feature PMFs, in a similar manner to the discrimination functions obtained in the previous examples.

It is interesting to consider this situation, when the task is to discriminate between only two objects on the basis of k possible sensing states. Setting $n = 2$ in Equation (5.10) gives (see Appendix B.5)

$$D = 2\,P(o_1)\,P(o_2)\sum_k |P(state_k|o_1) - P(state_k|o_2)| \tag{5.11}$$

This result is exactly the same as that obtained in Example 5.2.2, except that the sum is performed over the k possible sensing states, rather than over two possible states. Setting $k = 2$ in Equation (5.11) gives rise to Equation (5.4).

Again the optimal sensing location is the location at which the difference between the probabilistic membership functions is greatest, satisfying the intuitive arguments presented earlier.

5.2.6 Localizing an object

Consider the problem of localizing a single object, whose shape is known. This requires constraining the feature uncertainty function to be a delta function.

Hence detection of (or failure to detect) the object at a location which significantly reduces the spread of the feature uncertainty function should give a high value for the utility function. Sensing at locations which have no effect on the position uncertainty should give low values.

Generally, if the object is detected at a location at which there was an *a priori* low probability of detection, there will be considerable reduction in the uncertainty in its position. This is because locations with a low *a priori* probability correspond to a small portion of the uncertainty range giving rise to detection at that point. If the object is detected at such a point, its uncertainty range can be constrained to that small subset of values which supported the observed sensory outcome. Such a situation will result in a high utility value. This suggests the utility function

$$\begin{aligned} U(detection = 1, (x,y)) &= 1 - P(detection = 1)(x,y) \\ &= 1 - \Pi(x,y) \end{aligned}$$

Similarly, failure to detect the object at a location at which there is a high probability of detection will result in a high utility value. This suggests the following utility function.

$$\begin{aligned} U(detection = 0, (x,y)) &= P(detection = 1)(x,y) \\ &= \Pi(x,y) \end{aligned}$$

Combining these results leads to the discrimination function,

$$\begin{aligned} D &= U(detection = 0)\, P(detection = 0) \\ &\quad + U(detection = 1)\, P(detection = 1) \\ &= \Pi\,(1 - \Pi) \;+\; (1 - \Pi)\,\Pi \\ &= 2\,\Pi\,(1 - \Pi) \end{aligned}$$

The form of this quadratic function is well-known and attains its maximum value when $\Pi = 1/2$. Hence, for the example of the the square of side-length l, whose PMF was found in Subsection 3.1, the optimal location for sensing to localize the object, is to sense at a location removed a distance $l/2$ along the direction of the edge from the first sensed point. This is the expected result for constraining the position of the object with the minimum number of sensing locations.

5.3 Discussion of Examples

The two utility functions proposed for each of the object recognition tasks underline the notion that several functions may be appropriate for a given task. In implementing object recognition systems using these functions (see Chapter 7) several features were observed which led us to favour one method over the other.

The problems arising with the methods described in Case 1 (Examples 5.2.1 and 5.2.3) are probably best identified by way of an example.

Consider the problem of discriminating between three objects on the basis of binary contact data. Assume also that sufficient data has been collected to eliminate one object as a possible interpretation of the object and that the probabilities for the remaining interpretations are equal. Therefore

$$P(o_1) = 1/2, \quad P(o_2) = 1/2, \quad P(o_3) = 0$$

According to the Case 1 reasoning, in evaluating potential sensing locations, the utility function is constructed so that it favours combinations of location and sensing outcome which lead to *a posteriori* expectations for the objects which differ most from the initial "no information" state

$$P(o_1) = 1/3, \quad P(o_2) = 1/3, \quad P(o_3) = 1/3$$

Consider two locations one of which is certain to give rise to the *a posteriori* expectations

$$P(o_1) = 0.35, \quad P(o_2) = 0.65, \quad P(o_3) = 0$$

and the other which is certain to produce the following a posterior expectations

$$P(o_1) = 1/2, \quad P(o_2) = 1/2, \quad P(o_3) = 0$$

Given that the eventual aim is to obtain the situation where $P(o_i) = 1$ for object o_1 or o_2, the first of these outcomes represents a significant step forward towards identifying the sensed object, whilst the second location represents no improvement on the current knowledge. In fact the second set of expectations would result if we sensed at a point at which the state was already known, which would clearly be a pointless exercise. There is no doubt that the first alternative is the preferable location at which to sense.

In spite of this, the utility of both situations is the same using the methods of Case 1. For the first location

$$\begin{aligned} U &= \sum_i |3\,P(o_i|state) - 1| \\ &= |3 \times 0.65 - 1| + |3 \times 0.35 - 1| + |3 \times 0 - 1| \\ &= 0.95 + 0.05 + 1 \\ &= 2 \end{aligned}$$

For the other location

$$\begin{aligned} U &= \sum_i |3\,P(o_i|state) - 1| \\ &= |3 \times 0.5 - 1| + |3 \times 0.5 - 1| + |3 \times 0 - 1| \end{aligned}$$

$$= 0.5 + 0.5 + 1$$
$$= 2$$

In the light of this example, this utility function was deemed inappropriate in achieving the goals of this task, as it was possible for the system to continue to choose the same location indefinitely, and hence reach a point at which it no longer collected useful information. For this reason, it was seen as desirable that utility function should assign zero utility to any outcome which does not alter the current expectations. This was the motivation for the utility functions described in Case 2 (Examples 5.2.2 and 5.2.4).

The limitation of the Case 1 approach, described in this section, does not occur for the two-object recognition problem, but only for problems involving larger sets of objects. In spite of this the Case 2 approach was preferred in the implementations, even if the object set contained only two models, because it occurs as a simplification of the more general n-object utility function which was employed (see Example 5.2.4).

The other advantage of the Case 2 approach is the simpler calculations involved in computing the optimal sensing location. This can be seen by comparing the discrimination functions given by Equations (5.5) and (5.7) for the Case 1 and Case 2 approaches respectively. In that the utility functions chosen may not be "exact" in any measurable sense, there is some argument for choosing a utility function which gives rise to a simple discrimination function, in preference to a more complicated function, provided that the simple function results in a

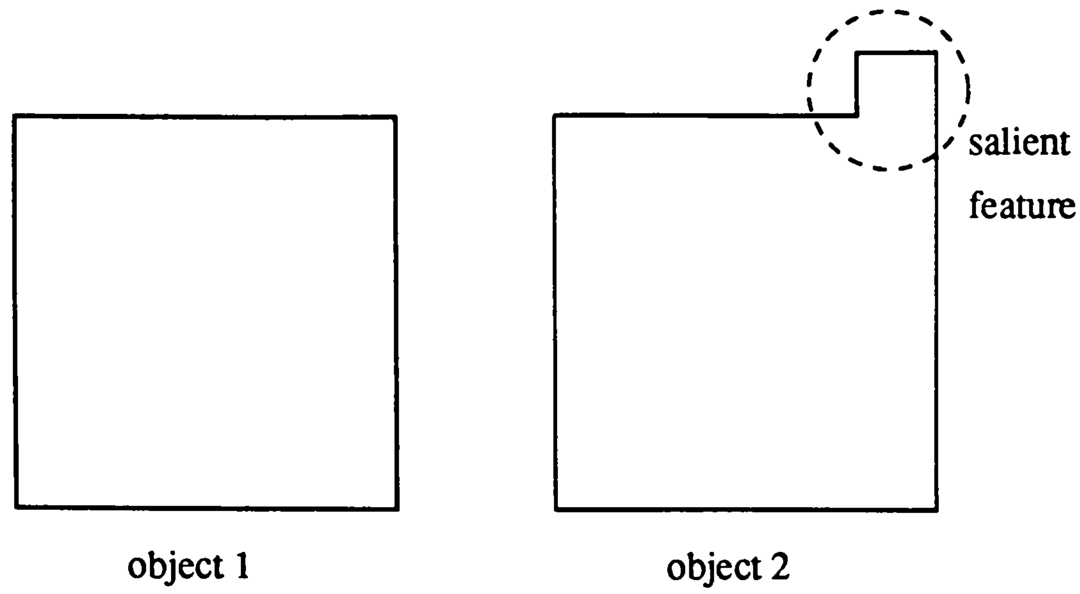

Figure 5.1: Object set with a salient feature.

"reasonable" sensing strategy.

5.4 Salient features

It has been recognized by several researchers that specific object features can play an important role in discriminating between particular sets of objects. The extent to which a specific feature constrains object interpretation has been termed its saliency and has been utilized in recognizing visually-occluded objects by Turney, Mudge & Volz [48]. Such features have much in common with the "seeds of perception" proposed by Brady [10] to describe the constraining power of data from particular locations upon visual processes. The work of Bolles & Cain [8] also uses certain object features as a focal point for recognizing and locating partially-visible objects.

As an illustration of a salient feature, consider the two object models shown in Figure 5.1.

The two models are identical except for the small protrusion on *object* 2. Given an object from this set, it is necessary either to detect this protrusion, or

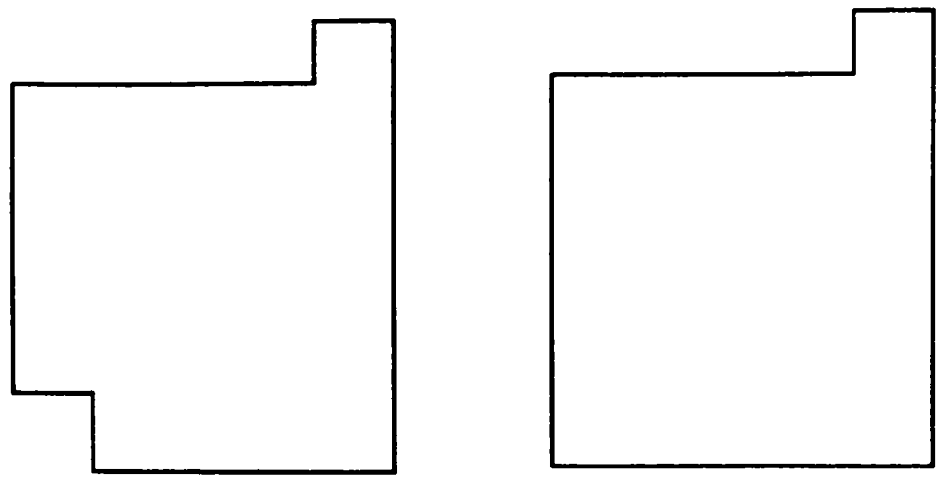

Figure 5.2: Object set with a different salient feature.

else prove that it is not present, in order to identify the object with certainty. In that it is the distinguishing feature between the two objects. the protrusion is said to have high saliency, or to be a salient feature for the set.

Salient features are a function of the set, not the individual object on which they appear. In a different object set, the protrusion may not be a salient feature, as, for example, in the set shown in Figure 5.2.

Clearly, salient features must be detected (or eliminated) to identify objects, and hence should play an important role in a guided sensing strategy. We claim that it is not necessary to make special provision for salient features in the utility function in order to ensure that they are effectively utilized in the sensing strategy. Instead, it is only necessary to ensure that the salient feature is adequately represented by a feature PMF, and a well-chosen utility function will automatically utilize the saliency of the feature. It is not necessary to identify the feature to the system as having high saliency, as this will be discerned by the system.

Consider again the object set represented by Figure 5.1. An appropriate representation for the salient feature in this case need not be the whole protrusion, but may be, for example, the concave vertex, of which there are two for *object* 2 but none for *object* 1. If the feature PMFs are generated for this vertex for *object* 1 and *object* 2, they will be zero everywhere for *object* 1, but non-zero at some locations for *object* 2. Recall the discrimination function obtained in Example 5.2.5 for the problem of discriminating between two objects on the basis of k features.

$$D = 2\,P(o_1)\,P(o_2)\sum_k |P(state_k|o_1) - P(state_k|o_2)|$$

The optimal sensing location is the one at which the sum of differences between the feature PMFs is greatest. Consider the contribution of the PMFs for the concave vertex. This will be greatest at that location where the expectation of detecting the feature (for *object* 2) is greatest, as this will represent the greatest difference from the *object* 1 expectation for this feature, which is zero everywhere. Thus the system will attempt implicitly to seek out the salient feature to determine its presence, without having received any information as to the saliency of particular features. The system effectively identifies salient features on the basis of their PMFs.

5.5 Look-ahead search

A common technique in Artificial Intelligence and Game Theory for determining the optimal action in the presence of uncertainty is to consider the long term effects of the alternative moves between which a choice must be made, by considering the accumulated effect of several moves. This technique is termed "look-ahead search" [40,51]. It would appear that this approach has application to the sensing problem.

To demonstrate the application of these techniques, consider the problem of traversing a maze. every time a junction is encountered a choice must be made as to which path to take. Assuming that the various paths are not signposted to indicate the shortest distance along them to the goal, some heuristic must be applied to aid in the choice – perhaps the path will be chosen that heads in the perceived direction of the goal. This short-sighted guess as to the correct path may be proven wrong by dead ends or changes in path direction, which are just out of sight. If it were possible to look further ahead, a better choice may be made. Indeed, if a map were available, it would be possible to look ahead for each path to determine the optimal one.

Assume that it were possible to see beyond the choice at hand to the next junction along each path. Applying the same heuristic to each second junction to determine their worth is more likely to provide an accurate measure of which path to choose at the first junction. Clearly, the path should be chosen at the first junction which leads to the second junction from which emerges what appears to be the best path at this stage. This is called a two-ply search [40]. Looking ahead to a further junction would result in a three-ply search, and so on.

Attempting to apply these techniques to the sensor placement problem, it is clear that the decision to be made in this case is which sensory action to pursue. The goal in this case is the recognition of the sensed object. It may be at some stage of the recognition process that it is not possible to recognize the sensed object on the basis of data from one further sensing location. It may however be possible to recognize the object on the basis of two points, if the first is chosen appropriately. This suggests a strong argument for a look ahead search. Considering the outcome of combinations of two (or more) sensing actions should lead to a more efficient sensing strategy. If it were possible to perform an unlimited stage look ahead search, then the optimal strategy could be exactly determined.

A significant problem arises, however, in attempting a look ahead search over the space of possible sensing action, due to the size of the branching factor at each stage. As the space is continuous, there is a choice between an infinite number of sensing actions at the first stage, each of which leads to an infinite choice at the second stage, and so on. It is clearly not possible to evaluate the utility of the combination of all of these alternative sensing actions. Unless the number of choices can be reduced to a finite number, it will not be possible to perform a look ahead analysis.

The only alternative to this problem, is to provide an accurate measure of the distance form the goal at each stage of the search process. Considering again the example of traversing a maze, it is clear that a look ahead search will not be necessary if each of the alternative paths is accurately signposted indicating the distance to the goal.

The same will be true of the sensing problem. If an accurate measure exists of the amount of sensing remaining to identify the object, then there will be no need to look ahead. The role of the utility function is to provide this measure. The more accurate the utility function in evaluating the merit of each combination of sensing action and (unknown) state, the more efficient will be the generated sensing strategy. If the utility function provides an exact measure, then the resulting strategy will be justifiably optimal.

Chapter 6

Representing uncertainty

6.1 Sequential analysis

In previous chapters, analysis has been presented showing how sequential sensing actions can be chosen (on the basis of a statistical decision theoretic analysis), to perform a desired task. At the same time, it is necessary to monitor the progress towards accomplishing the task, and to determine when a solution has been obtained. The decision of when to cease gathering more sensory data, and propose a result, is arrived at using the sequential techniques first proposed by Wald [49]. Statisticians often refer to these techniques as experimental design, as the choice is essentially how much data to collect (ie. what size of statistical experiment should be performed), on the basis of which a decision shall be made.

Sequential analysis ties in closely with the approach to object recognition developed in this thesis thus far. The experimental design of acquiring data and then updating expectations and choosing the next sensing action represents a typical sequential design, and hence the problem of determining the endpoint for the experiment provides a natural application for sequential techniques.

The terminology adopted in this chapter follows closely that used by Wald. The aim of a statistical experiment is considered to be the testing of a set of hypotheses. After each sample is taken, there is a choice between three alternative responses for each hypothesis. These are either to accept the hypothesis, reject the hypothesis, or defer judgement pending further samples.

In general, it may not be possible to state with certainty whether or not a given hypothesis is true from a finite amount of data. However, using statistical techniques, it will be possible to specify at each stage the perceived likelihood of the truth of each hypothesis, and thus to specify the probability that a chosen decision is incorrect.

Associated with any decision, there are two possible types of error: rejection of the hypothesis when it is true (called an error of the first kind, whose probability is denoted by α); and acceptance of the hypothesis when it is not true (called an error of the second kind, whose probability is denoted by β). It is common in sequential analysis to specify the acceptable errors beforehand, and to continue acquiring more data, until a decision can be made for which α and β fall within the stated bounds.

In the present case, the task of identifying an object can be considered as the testing of a hypothesis regarding the object's identity. For the problem of discriminating between n objects known to the system, there may be n hypotheses, each stating: "the identity of the object is *object* n". Clearly these hypotheses are not independent as the acceptance of one hypothesis will imply the rejection of all the others. The approach taken in this thesis is to maintain a probability measure for each object in the model set representing the likelihood that it corresponds to the sensed object. As the aim is to propose the identity of the sensed object, there is more concern with errors of the second kind, β, (ie. accepting a hypothesis as to the object's identity when it is not true), although it is clear that such an error implies in turn an error of the first kind for one of the other hypotheses. Once a value of β has been specified, then as soon as the probability of one of the models exceeds $1 - \beta$, that model can be accepted as the identity of the object within the stipulated error bounds. It is possible to incorporate the error bound α by removing from the set of admissible objects, those whose probabilities fall below α. As the termination of the experiment only occurs with the acceptance of an identity for the object, errors of the first kind (as measured by α) are not applied in the implementations described in Chapter 7. This approach seems reasonable as the expectation of unlikely objects may rise above the α threshold with the incorporation of further sensory data (which is being gathered in any event if the object has not yet been identified).

The only reason for proposing different values for α and β would be if it were more damaging to commit one type of error than the other. Consider the problem of recognizing an object from the set $\{o_1, o_2\}$ and assume that there is a large penalty associated with identifying the object as o_1 when it is actually o_2, but only a small penalty associated with identifying the object as o_2 when it is really o_1. In such a situation, it would be advisable to set a very tight bound on errors of the second kind (ie. a small value for β), but a slacker bound on errors of the first kind (ie. not such a small value for α). In this case, larger errors in rejecting H_1 (where H_1 is the hypothesis: "the object is o_1") incorrectly will be more readily tolerated than errors in accepting H_1.

An important component of sequential analysis which is not treated in this thesis is the cost of experimentation, or for our applications, the cost of sensing. (See Berger [6] for a thorough treatment of this topic.) To reduce the uncertainty in some sensing situations, it may be possible to continue acquiring data indefinitely to further refine the accuracy of the object models. However, it is generally desirable to reach a conclusion within constraints of time and computational demands. The cost of experimentation is a method of weighing the additional time and computational resources required to extract and process an additional set of sensory data, against the expected information content of the data that will be obtained, which is described by the utility function. This leads to a more general "overall" utility function, composed of the difference between the utility and the cost functions at each stage of sensing. The general sequential analysis strategy is to continue sensing whilst the expectation of the utility function exceeds the cost of experimentation (ie. the "overall" utility function is positive.)

Because the implementations dealt with in this thesis required strategies with very few sensing actions to achieve the experimental goal, the cost of experimentation was never felt to be significant in comparison to the utility of the information that stood to be gained by acquiring more data, prior to the experimental goal being realized. For this reason, the cost of experimentation was not treated explicitly in the decision theoretic analysis.

Although, no explicit allowance was made for cost of sensing, by utilizing the admissible error bounds (α and β) there is a recognition that a stage will be reached at which sufficient accuracy has been obtained, and there is no justification in obtaining further data. If there was no cost associated with sensing, then there would be no reason to conclude an experiment before error bounds of zero could guaranteed, as there is nothing to be lost by obtaining more data. Thus, the application of error bounds, implicitly recognizes the cost of sensing, and plays a similar role in terminating the experiment.

The application of these principles to the sensing problem are demonstrated in an example at the end of this chapter.

6.2 The probabilistic interpretation tree

The application of an interpretation tree as a tool in model-based object recognition was proposed by Gaston and Lozano-Pérez [20] and Grimson and Lozano-Pérez [26] and has since found widespread application for this task. (The interpretation tree is described and discussed in Section 2.2.)

One of the limitations of this approach to object recognition (discussed in Section 2.2) is that it places no ordering on the admissible interpretations of sensory data to object models at each stage of the sensing process. On the basis of each set of sensory data, potential assignments of object models to sensory data are either retained or culled for the interpretation tree, until only one interpretation remains. No measure is proposed to gauge the likelihood of the remaining models, prior to identification of the object.

In using the techniques of sequential analysis discussed in the previous section, it is assumed that it is possible to provide a probabilistic measure of the relative likelihood of different models after the processing of each sensory point. These *a posteriori* probabilities can be associated with each branch of the interpretation tree and updated as more data is acquired to provide the desired ordering on alternative interpretations.

The method by which the *a posteriori* probabilities are obtained is by the application of Bayes Theorem to the conditional probabilities of detection (the probabilistic membership functions) utilized in determining the sensing strategy. Because the techniques proposed to determine the sensing action at each stage enable the conditional probabilities of sensing specific features to be calculated, these probabilities can be used after sensing has taken place to assign an expectation to each interpretation. The probabilistic membership functions measure the conditional likelihood of sensing a given feature given the identity of the object. Applying Bayes theorem gives:

$$P(object_i|feature) = \frac{P(feature|object_i)\,P(object_i)}{\sum_j P(feature|object_j)\,P(object_j)} \tag{6.1}$$

Thus the probability estimate for each object can be updated to the conditional probability by inserting the conditional probabilities of detection in the above formula (for $P(feature|object_j)$).

One advantage of this method is that it is possible to utilize the constraints imposed by failing to sense an object at a given location, which cannot be easily integrated into the standard interpretation tree approach. Using probabilistic techniques, these constraints are reflected both in the updated feature uncertainty function (described in Section 4.3) and in the *a posteriori* probabilities for each object. Just as it is possible to state the likelihood of detecting any given feature (conditional on the object's identity) at each stage of the sensing process, it is also possible to determine the likelihood of not detecting the object. These probabilistic values for non-detection can be used in the same manner as PMFs for detection (effectively by treating non-detection as a feature) to update the probabilities of each object. In this manner the weightings applied to each branch in the interpretation tree can be adjusted although the object has not been contacted.

Most importantly, at each stage of the process, the probabilistic method provides information regarding the current probabilities of each interpretation from the admissible set. Thus if for some reason it is not possible to acquire more sensory information, it is still possible to make some statement as to current expectations of alternative interpretations. Using the method of Grimson and Lozano-Pérez, no information is offered as to the relative likelihoods of different interpretations – either an interpretation is admissible or it has been pruned from the tree. However, by introducing expectations, the possible interpretations can be ranked according to their likelihood at any stage of the sensing process, and a hypothesis accepted or rejected with some known probability of error.

To compare the two approaches, consider the task of discriminating between the two objects shown in Figure 6.1, assuming the edge of the object has been

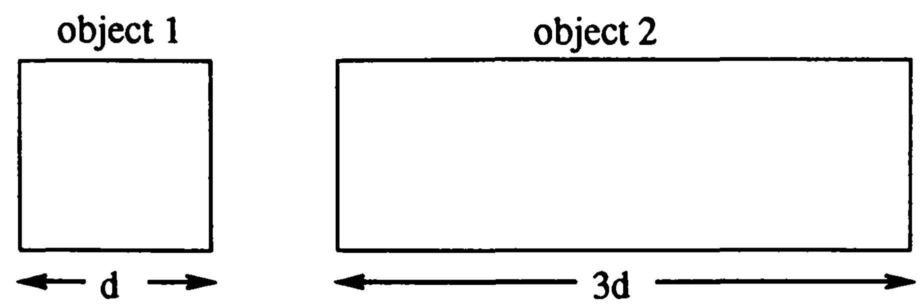

Figure 6.1: Object shapes.

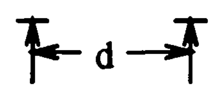

Figure 6.2: Sensory data.

sensed at two locations as shown in Figure 6.2.

The possible interpretations for the two objects are shown in Figure 6.3 superimposed on the sensory data.

It is clear that a range of locations of *object* 2 could give rise to detection at the two points as shown. However, to detect *object* 1 at both locations would require contacting the object very close to its vertices. On the basis of the first sensed point (which it would be initially assumed could be lying anywhere along the edge of either object), it would be considered a very unlikely event to sense *object* 1 again at a distance d removed from this point. Detection would however be quite likely for *object* 2, as a large range of possible interpretations of this object would give rise to detection at the second point.

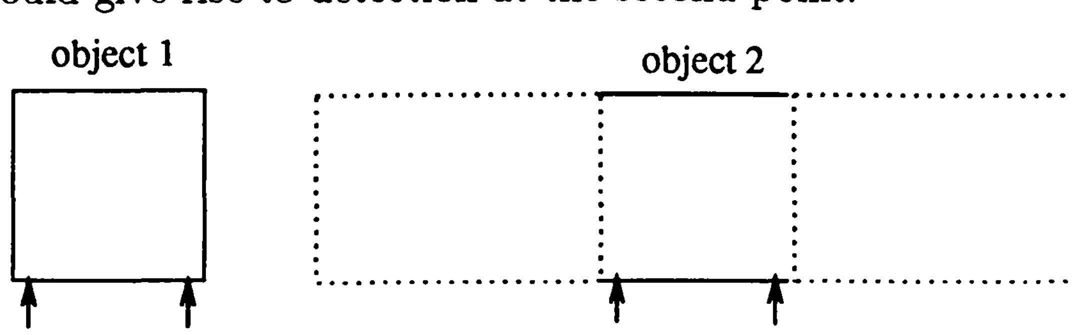

Figure 6.3: Possible object interpretations.

On the evidence of these two sensed points, it is most likely that the sensed object is *object* 2. This would be reflected in the *a posteriori* probabilities of the two objects (obtained using Equation(6.1)). However, as both objects do give rise to possible interpretations, neither would be culled from the interpretation tree using the approach of Grimson and Lozano-Pérez, and it would be impossible to reach any conclusion with respect to the object's probable identity on the basis of these two points.

In the following example, an application of these techniques is demonstrated for the problem of recognizing an object from a two-object set.

6.3 Example: The two object set

Consider the problem of recognizing the sensed object from a set of two objects $\{o_1, o_2\}$, described in Subsection 5.2.2. Applying Wald's terminology, an appropriate hypothesis for this task would be H_1: "the sensed object is o_1". In this case, it is not necessary to propose a second hypothesis, H_2: "the sensed object is o_2" as this simply corresponds to the negation of H_1. It has already been seen in Subsection 5.2.2 that the proposed sensing location at each stage of the sensing process is the location which maximizes the difference in the probabilistic membership functions:

$$D(x, y) = |\Pi_{o_1}(x, y) - \Pi_{o_2}(x, y)|$$

After sensing it will have been determined whether or not the object is present at the chosen location. The *a posteriori* probability can then be calculated.[1]

$$\begin{aligned} P'(o_1) &= P(o_1|state) \\ &= \frac{P(state|o_1)\,P(o_1)}{P(state|o_1)\,P(o_1) + P(state|o_2)\,P(o_2)} \end{aligned}$$

If the *a priori* probabilities for the two objects are equal

$$P(o_1) = P(o_2) = 1/2$$

then this equation simplifies to

$$\begin{aligned} P'(o_1) &= \frac{P(state|o_1)}{P(state|o_1) + P(state|o_2)} \\ &= \begin{cases} \frac{\Pi_{o_1}}{\Pi_{o_1} + \Pi_{o_2}} & \text{if detection} = 1 \\ \frac{(1-\Pi_{o_1})}{(1-\Pi_{o_1}) + (1-\Pi_{o_2})} & \text{if detection} = 0 \end{cases} \end{aligned}$$

Assume that the error bounds α and β are considered acceptable for errors of the first and second kinds respectively. Then if

$$P'(o_1) > 1 - \beta$$

the hypothesis can be accepted, and the object can be identified as o_1.

Alternatively, if

$$P'(o_1) < \alpha$$

then H_1 can be rejected and the object can be identified as o_2.

If

$$\alpha < P'(o_1) < 1 - \beta$$

[1] The symbol P' is used to denote the *a posteriori* probabilities in order to distinguish them from the *a priori* probabilities P.

no decision can be reached and more data should be obtained.

If the object cannot be identified (within the stipulated error bounds) on the basis of the latest set of data, another cycle of data collection and analysis must be performed. Firstly the PMF must be updated using the techniques described in Section 4.3, and the discrimination function recalculated. Sensing can again be performed at the optimal location, and the *a posteriori* probabilities calculated using Equation (6.1). This procedure can be repeated until the *a posteriori* probabilities fall within the acceptable bounds, when a decision can be made.

Chapter 7

Implementations

7.1 Introduction

Several experiments were conducted to test the principles described in this thesis. The task chosen to investigate the performance of the sensor placement strategies was a "sparse data" object recognition system. It is important to note that the scope of the implementations was to test the active sensing strategies and not the sensing and object recognition systems. Hence, in the implementations, the sensing tasks were constrained so that the limitations of the sensing system did not obscure the behaviour of the sensor placement strategies.

As the motivation for the work described in this thesis was the "sparse data" problems encountered in tactile sensing, the initial intention was to implement an "intelligent" tactile sensing system. Limitations in currently available tactile sensors and the need for fine force control (not present on the robots available to perform the implementations) to implement effectively a tactile sensing system, led to the abandonment of this plan. Notwithstanding the unavailability of the necessary hardware, it was felt that to test effectively a sensor placement system, it was desirable to have a robust sensing system. The problems addressed by the implementations were those concerned with sensing strategy, not sensory data acquisition and low level data processing. Hence an attempt was made to design implementations for which reliable sensory data was readily available and presented no difficulties to process. Two-dimensional vision was chosen as a modality which satisfied these requirements, and for which the necessary hardware was available. To simplify the low level processing requirements, the objects to be sensed were chosen as black shapes, silhouetted against a white background, providing a binary vision scenario.

For all the experiments, the sensing system consisted of a CCD array camera mounted on a PUMA 560 robot arm. The camera was directed downwards onto a horizontal workspace, and restricted to move in the plane parallel to the workspace, under the control of the PUMA robot controller. The camera's field of view was restricted so that it could only sense a small segment of the shape at any instant and had to be repositioned several times to accumulate sufficient data to uniquely identify the sensed shape. (This was to achieve the characteristic of "sparse" sensory data.) The configuration of the robot, camera and workspace were as shown in Figure 7.1.

The task addressed was to identify a two-dimensional polygonal shape in the sensing workspace from a set of models known to the system. The shapes could be either convex or concave, and of arbitrary complexity. The nature of the problem

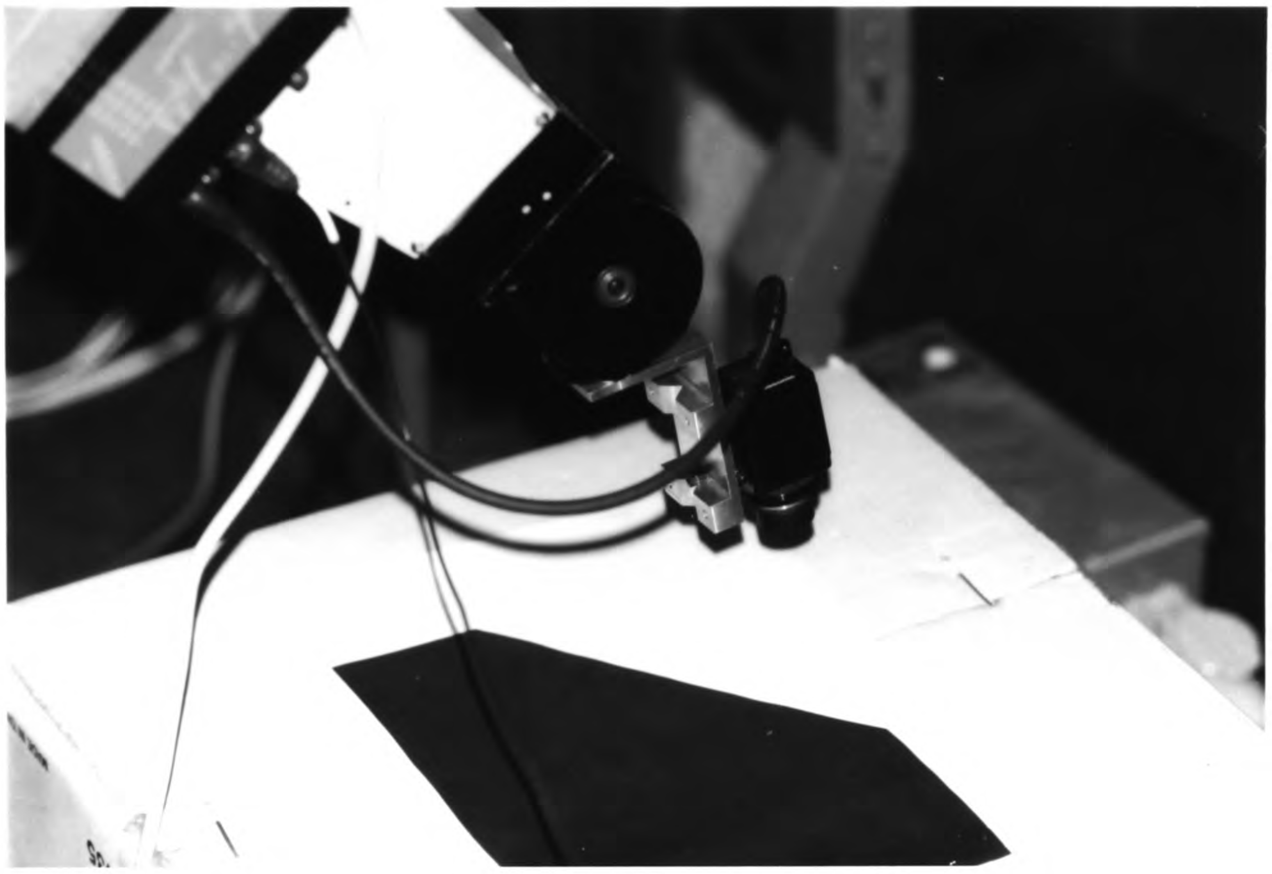

Figure 7.1: Configuration of the experimental equipment.

was strictly two-dimensional, and no three-dimensional issues were addressed. This was due mainly to the nature of the available hardware.

The images obtained by the camera were stored and low level image processing performed by a Datacube vision processing system mounted on the VME bus of a Sun Microsystems workstation. Higher level image interpretation and active sensing strategy planning were performed on the Sun workstation. All algorithms were implemented in the C programming language, and executed under the UNIX operating system. Motion commands were communicated to the PUMA controller in the VAL-I programming language by the Sun over a serial interface. The connections between the various components of hardware are shown in Figure 7.2.

The method of operation was that the robot would move to a specified loca-

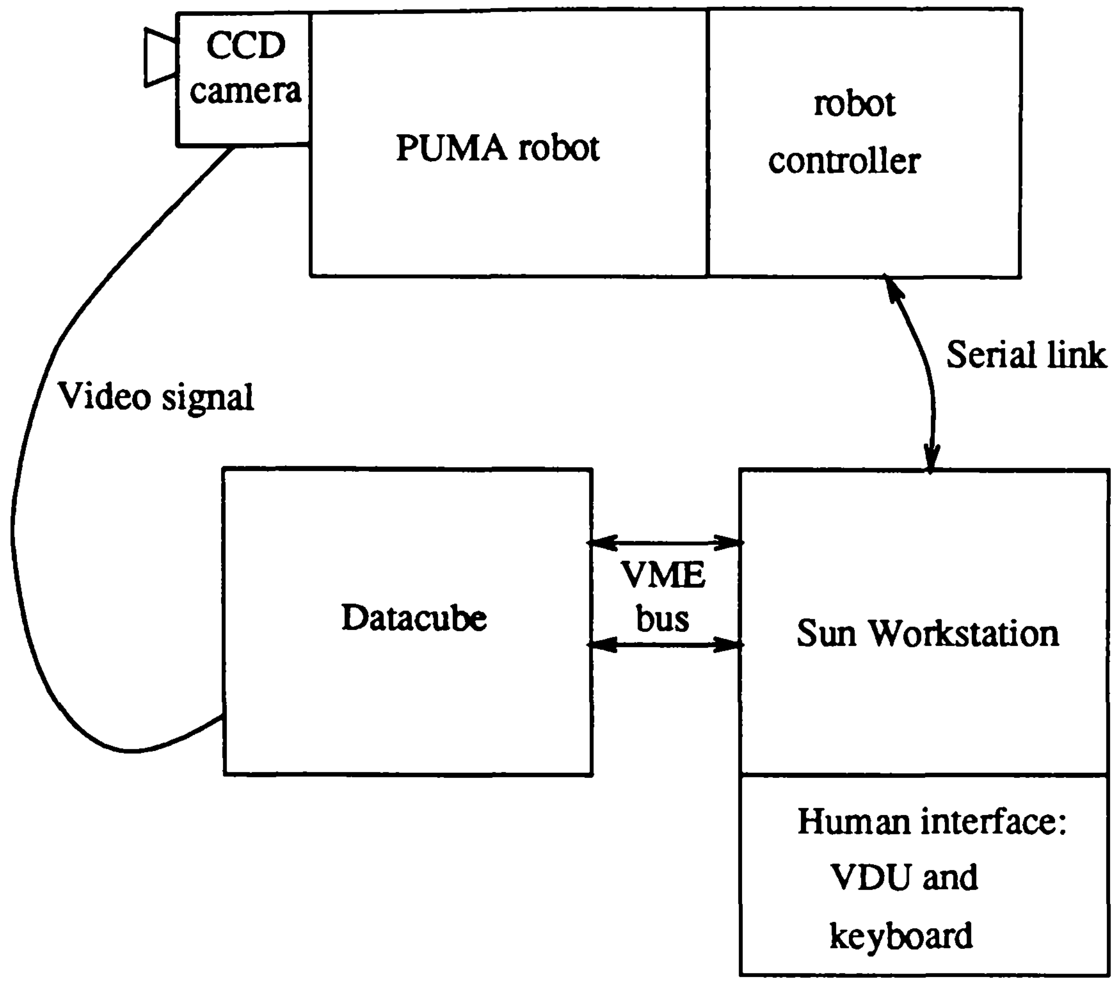

Figure 7.2: Connections between the hardware components.

tion; the camera would acquire an image at that location; the image would be stored and processed by the Datacube; the Sun workstation would interpret the processed image and identify the sensed object or determine the optimal sensing location; a command would be sent by the Sun to the PUMA controller initiating a move to the chosen location; and the cycle would continue. Hence, the feedback loop was completely closed and the system could perform active sensing and recognition tasks independently of human intervention. Because of the nature of the operating systems used by the workstation and the robot controller, and because of significant delays introduced by the serial link between these devices, the feedback control was not performed continuously in real-time, resulting in a static period while image processing and strategy planning were performed, between each discrete robot motion.

Two simplifications were applied to all the experiments. Firstly, any uncertainty in the data returned by the camera was not considered. Thus the sensor data was assumed to be perfect, corresponding to a delta function as the sensor uncertainty function ($S(x,y) = \delta(x,y)$). Secondly, the camera's field of view was always kept sufficiently small (relative to the size of the objects) so that the effects of finite sensor size (as described in Section 4.4) were not significant.

7.2 Location sensing

The four implementations described in this chapter are distinguished by:

- the manner in which data is acquired from the workspace;
- the type of information extracted from the sensory data;
- the method of modelling used to describe the shapes and workspace (grid-based maps or analytic representations);
- whether the sensing strategy is computed on- or off-line.

For the first implementation, the sensor could be positioned at a chosen co-ordinate location and could obtain data from a small, but finite, area at that location. Thus the sensor is capable of "probing" the workspace to detect the object, and may return data from inside or outside the object's boundaries.

The information provided by the sensor was processed to provide a single bit, indicating whether or not the object has been sensed by the probe. (An exception is the first sensed point, at which the data was processed as described below.) This binary detection capability is weaker than the ability to identify and measure object features (such as edges and corners) at each sensed region.

The first task that must be performed is to obtain a first contact location with the object in the workspace. This is achieved by incrementally tracking the sensor in from an extremity of the workspace and acquiring images until the boundary of the object is detected. The data from this location is processed to determine the slope of the edge or the presence (and included angle) of a corner. Once contact has been made, powerful constraints limit the possible placements of the shape, and the probabilistic membership functions and utility functions described in Chapters 4 and 5 can be readily evaluated. If a corner has been detected only a finite number of admissible interpretations exist, which match the sensory data to the object models. If an edge is detected, there is uncertainty in only one direction (parallel to the edge) as to the object's location.

The method of modelling used in this experiment was to describe the workspace by an occupancy grid. Similarly, grid-based models were generated to describe the shapes and the uncertainty ranges (or feature uncertainty functions). To determine the best next sensing location, it is necessary to evaluate the discrimination function at every point in the workspace grid, and identify the location at which the function attains its maximum value. The convolutions necessary to evaluate the PMF (as described in Chapter 4) are easily and rapidly performed using the Datacube once the workspace, object models and uncertainty ranges have been tessellated.

After sensory data has been obtained at a given location, the uncertainty ranges are updated and the discrimination function is re-evaluated at each point in the workspace grid. The sensing strategy is thus determined on-line, as the next sensing location is calculated sequentially with the collection of data at a new sensing point.

The algorithm for this implementation is given in Algorithm 1. The first four steps of the pseudo-code describe the initialization routines followed by the system, including reading the object set from a database, intializing the robot, and searching for the first contact point. (The algorithms describing the search for and processing of the first contact point are described later.) Once the first contact has been obtained, the system enters a loop which it follows until the object is identified. This loop includes: updating the uncertainty range for each admissible contact side; evaluating the PMFs and determining the next sensing location; and obtaining data from the proposed location. (Steps 7 to 11 of the pseudo-code.) Each time a new set of data is obtained, the object probabilities are updated, and, when the object can be identified within the required confidence bounds (ie. one of the updated probabilities is within some margin of one) , the loop is exited. Otherwise, the uncertainty ranges are again updated and the cycle continues.

The algorithm for the initial contact search is given in Algorithm 2. The search involves moving the sensor to an extremity of the workspace and then commencing a loop involving alternately stepping the sensor along a chosen trajectory and acquiring images. From each image a pixel histogram is obtained. If the image

```
Identify:

 1. Read in object models, translate and rotate models to align
    with robot-centred coordinate system, check for symmetry, fill
    object model grids, and calculate the side length and angle
    sizes.

 2. Initialize and calibrate the robot.

 3. Search for and process the first contact point.

 4. Calculate the object and contact side probabilities.

 5. If the object is not identified:

 6.     Repeat:

 7.         Update the uncertainty ranges.

 8.         Convolve the object model with the corresponding
            uncertainty range to obtain PMFs.

 9.         Evaluate the discrimination function at all grid
            locations, and determine the maximum point as the
            optimal sensing location.

10.         Move to and acquire data from the optimal location.

11.         Update the object and contact side probabilities.

12.     Until the object is identified.

13. Return identity of object.
```

Algorithm 1: Location sensing for object identification.

consists of a significant number of "black' and "white" pixels, it is processed to determine the presence of an edge or corner. If such a feature is detected, the loop is exited, and the describing qualities of the feature are returned to the recognition algorithm.

The algorithm used to determine the presence of an edge or corner in an image is listed in Algorithm 3. The algorithm uses a Sobel operator to determine points of local discontinuity in the image, and forms a "chain" of connected points of high discontinuity. A long chain is taken to indicate a section of the object's boundary. The chain is broken into a set of connected segments using the algorithm of Pavlidis [39]. This approach initially considers the straight segment defined by the endpoints of the chain. If the chain deviates significantly from the segment, the chain is split at the point which deviates most, and the attention of the algorithm is then focussed on the two sub-chains thus formed. If the chain consists of only one straight segment, the feature is interpreted as an edge, and if more than one segment is present, a corner is returned. For the case of an edge, the slope is measured, and for a corner the included angle is determined.

An example of the information supplied by the implemented system whilst operating is shown in Figure 7.3.

This system was successful in identifying the shapes present in the workspace, in what was felt to be an efficient manner with regard to the number of sensing actions. The number of sensing locations required to discriminate between a set of objects clearly depends on the number of objects and their similarity. The increase in the average number of sensing locations required to identify the object with

```
Search(starting location, trajectory):

 1. Move to starting location.

 2. Repeat:

 3.      Acquire image.

 4.      Obtain binary histogram of pixels.

 5.      If there are greater than some threshold number of both
         ``black'' and ``white'' pixels in the image:

 6.           Process image to extract edge or corner features.

 7.           If edge or corner is detected of greater than some
              predetermined length:

 8.                Break.

 9.      Step one frame dimension along trajectory.

10. Until sensor is outside predetermined workspace bounds.

11. If edge or corner has been detected of required length:

12.      Return location of sensor, type of contact (edge or
         corner) and describing features (edge slope or corner
         angle).

13. Else:  (if no object has edge or corner has been detected
    along chosen trajectory)

14.      Call search again with a different starting location and
         trajectory.
```

Algorithm 2: Search for the first contact point.

```
Process(image):

 1. Convolve image with Sobel operator in x- and y-directions.

 2. For (i = 0; i < number of rows; ++i):

 3.      For (j = 0; j < number of columns; ++j):

 4.           If x-convolution or y-convolution at pixel(i,j)
              exceeds a predetermined threshold value:

 5.                Start a chain.

 6.                Repeat:

 7.                     Add pixel(i,j) to chain.

 8.                     Set pixel(i,j) = 0.

 9.                     Consider adjacent pixels (in tangent
                        direction of edge) and set (i,j) to the
                        pixel with the greatest convolution
                        value (in direction consistent with
                        previous data).

10.                Until convolution value at pixel(i,j) is less
                   than the threshold.

11.                If chain is longest obtained thus far:

12.                     Set longest chain = current chain.

13. If length of longest chain is less than predetermined length:

14.      Return null value.

15. Else:

16.      Apply Pavlidis algorithm to break chain into
         straight line segments.

17.      If there is only one segment in the chain:

18.           Return value for a line, and measure of slope.

19.      If two segments:

20.           Return value for a corner, and measure of included
              angle.
```

Algorithm 3: Processing to determine edge or corner contact.

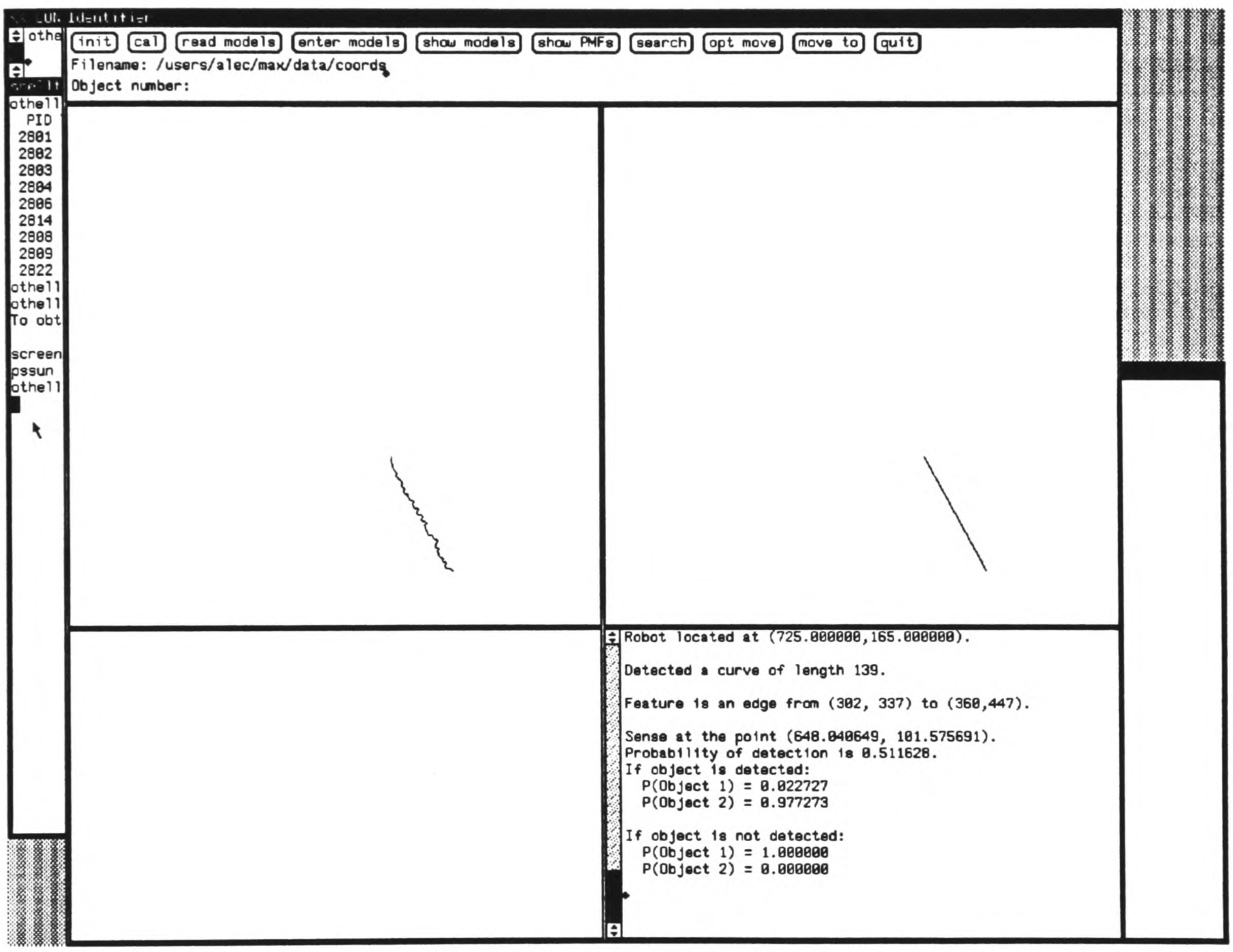

Figure 7.3: Screendump from the system in operation.

increasing number of admissible objects in the set, was observed to be less than a linear relationship. Although it was rarely possible to discriminate between two objects on the basis of two sensory points, it was usually possible to discriminate between n objects on the basis of n points, once n was of the order of five or greater. Clearly more points are required to identify between objects which are of very similar size and shape. An example of a typical strategy generated by the system for this problem is analyzed in Section 7.6.

The main problem with this implementation of the system was that it was felt to be slow in determining each sequential action, particularly if many objects or complex objects were present in the model set. There is also an implicit trade-off between speed and accuracy for a system such as this, which utilizes occupancy grids, as coarser tessellations of the workspace result in faster determination of the sensing strategy at a cost of reduced spatial resolution. An analytic solution to maximizing the discrimination function would overcome this limitation, however such a solution is not readily available. The difficulties involved in finding such a solution and an attempt to overcome these limitations are described in Section 7.4.

7.3 Path sensing

In the second implementation, it was assumed that data could be acquired by tracking the sensor along a given trajectory from an extremity of the workspace, until contact is made with the object (or until another extremity of the workspace is reached). This approach was intended to model "beam-type" sensors such as sonar, radar or infra-red light which can only sense along a given direction and not at a given point in Cartesian space. Tactile sensors may also operate in this manner if they are used to provide feedback in controlling the closing grasp of an actuator on an object.

Because of the nature of the data acquisition, the sensor will only ever obtain data from the boundary of the sensed object – it is impossible to sense within the object's interior. We assumed that we could compute the presence of object edges or corners within the sensor's scope, and measure their slopes or angles of inclusion respectively. This provides stronger constraints than the simpler sensor, modelled in the previous implementation, which can only detect the presence or absence of the object at each location. Clearly, to return features such as the slopes of edges, it is necessary to sense over a finite field of view. The ability to identify given features (corners and edges) enables them to be used in active sensing and recognition. From the object models and sensory data, PMFs can be generated for each identifiable object feature; in this case, edges with slopes within certain ranges and corners with certain ranges of interior angle. The PMF for each feature can then be included in the utility function. (The advantages of this capability were described in Section 5.4.) The saliency, or constraining power of each feature is implicitly utilized by the system, and salient features are sought by the sensing strategy.

Grid-based models were again used to describe the space of possible actions. In this case, however, the space does not correspond to the Cartesian space describing location in the workspace. Instead, each possible sensing path is described by the polar coordinates as defined previously in Section 4.6. As we are interested in determining the probabilities of sensing given object features in response to each sensory action, and evaluating the merit of each alternative action, both the PMFs and the utility functions need to be described in this coordinate system. This is again done exhaustively, with the discrimination function being evaluated at each point in the sensing-space grid. In this case, the convolution capabilities of the Datacube are of no advantage, as the PMFs in the sensing space coordinate system do not correspond to simple convolutions of object models and uncertainty ranges. The calculation of the PMFs in this coordinate system is described in Appendix C.

The algorithms for this implementation are essentially the same as those shown in Algorithms 1 to 3. The main differences are that there is no initial need to fill an object model grid, as the PMFs cannot be evaluated by convolving object models with the uncertainty functions, but must be determined point-by-point from the corner coordinates of the object models (as provided by the database of object models). Also, there is no need to translate and rotate the object models at this stage of the computation. In this case the coordinate system is referenced to fixed axes in space (which define the sensing coordinates (r, θ)). Once the first sensed point is obtained it is necessary to rotate and translate the object models to align with the sensory data. These differences are revealed in the algorithm for the path sensing implementation, which is given in Algorithm 4. There are no changes in the search and processing algorithms (Algorithms 2 and 3).

As with the previous implementation, the sensing strategy was successful in

```
Identify:

 1. Read in object models and determine the a priori object
    probabilities.

 2. Initialize and calibrate the robot.

 3. Search for and process the first contact point, and translate
    and rotate models to align with the sensory data.

 4. Update the object and contact side probabilities.

 5. If the object is not identified:

 6.      Repeat:

 7.           Update the uncertainty ranges.

 8.           Evaluate the PMFs at all points in the sensing space.

 9.           Evaluate the discrimination function at all grid
              locations, and determine the maximum point as the
              optimal sensing location.

10.           Move to and acquire data from the optimal location.

11.           Update the object and contact side probabilities.

12.      Until the object is identified.

13. Return identity of object.
```

Algorithm 4: Path sensing for object identification.

identifying the object in the workspace. It was, however, significantly slower than the first experiment in determining each sequential action, because of the more complicated calculations that had to be performed for each possible sensing action, with no assistance from the parallel processing capabilities of the Datacube.

The added capability engendered by the multiple PMFs (one for each object edge) was apparent in the sensing strategies chosen, and led to more efficient strategies with regard to the number of sensing actions. Fewer sensing actions were required to discriminate between given sets of objects using this system than using the system described in the previous implementation. In general, fewer sensing actions were required to identify the object than the number of objects present in the set, with significantly fewer required as the number of objects increased (ie. the relationship between objects and sensing actions was markedly less than linear.)

As an example of the strategies made available through the ability to discriminate specific features, consider the trivial case of discriminating between the square and triangle shown in Figure 7.4. If it is possible to discriminate the slope of the edge contacted, and generate PMFs for each edge, then *path* 1 can be guaranteed to provide sufficient data to identify the object. In this case

$$P(slope = 120|object = object\ 1)(path\ 1) = 1$$

$$P(slope = 90|object = object\ 2)(path\ 2) = 1$$

and *path* 1 is an optimal sensing path for the recognition task.

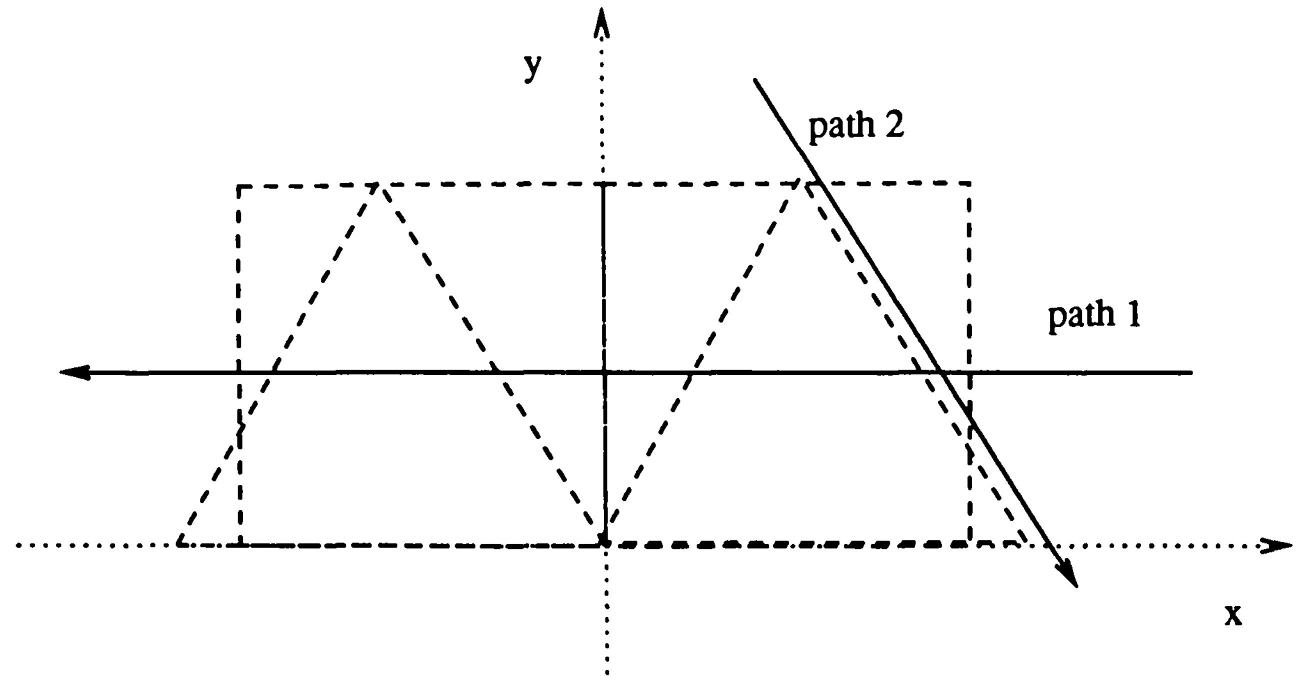

Figure 7.4: Alternative sensing strategies.

If, however, only binary contact information is available

$$P(contact|object = object\,1)(path\,1) = 1$$
$$P(contact|object = object\,2)(path\,2) = 1$$

and *path* 1 is a poor strategy. In this case, a better option would be *path* 2, which has different probabilities of detection for each of the objects, and hence will give rise to different *a posteriori* probabilities for the two objects. With only binary contact data available, identification cannot be guaranteed on the basis of one further sensing action.

7.4 Efficient strategy generation

One limitation of grid-based modelling approaches is the implicit tradeoff between accuracy and speed of processing, as was discussed in Section 7.2.

To determine the best sensing action to pursue, it is necessary to compare the merits of all the alternative actions, of which there will be infinitely many.

Dividing the space of sensing actions (which is of finite dimensions) into a grid, and evaluating the discrimination function at all points in the the grid, is one attempt to overcome the problem of comparing an infinite number of options.

A preferable solution would be to obtain an analytic expression for the maximum of the discrimination function. This would provide an exact solution for the best sensing action for the cost of only one computation, as opposed to the inexact solution provided currently by evaluating the discrimination function at every point in a grid. Problems arise in finding an exact solution because of the nature of the discrimination function.

The discrimination function is a combination of the PMFs, which are themselves convolutions of the object models and feature uncertainty functions, neither of which are continuous functions. Determining the maximum value involves differentiating and solving the discrimination function, which, considering the integral functions involved with the convolutions, proves to be a non-trivial problem to implement on a digital computer.

Some insight into a possible solution is gained by observing the typical cross-sectional form of a PMF for sensing in Cartesian space as shown in Figure 7.5.

The important characteristic is that the PMFs consist of straight-line segments, and the same is true if a y-cross-section of the PMF is considered. In fact, for this problem, the PMF in two dimensions consists of planar segments in space, and, as the discrimination function is a linear combination of such PMFs, it becomes apparent that the maximum for the discrimination function will occur at a corner point of the PMF. This suggests that the only points at which it is

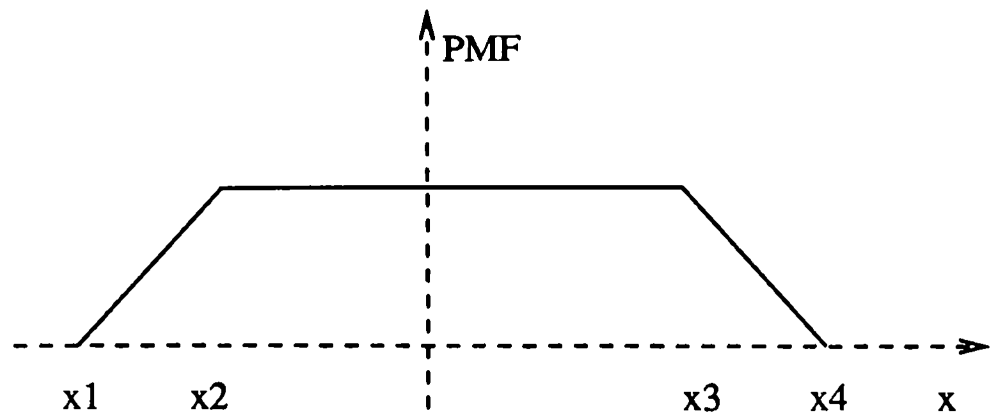

Figure 7.5: Typical cross-sectional form of a PMF.

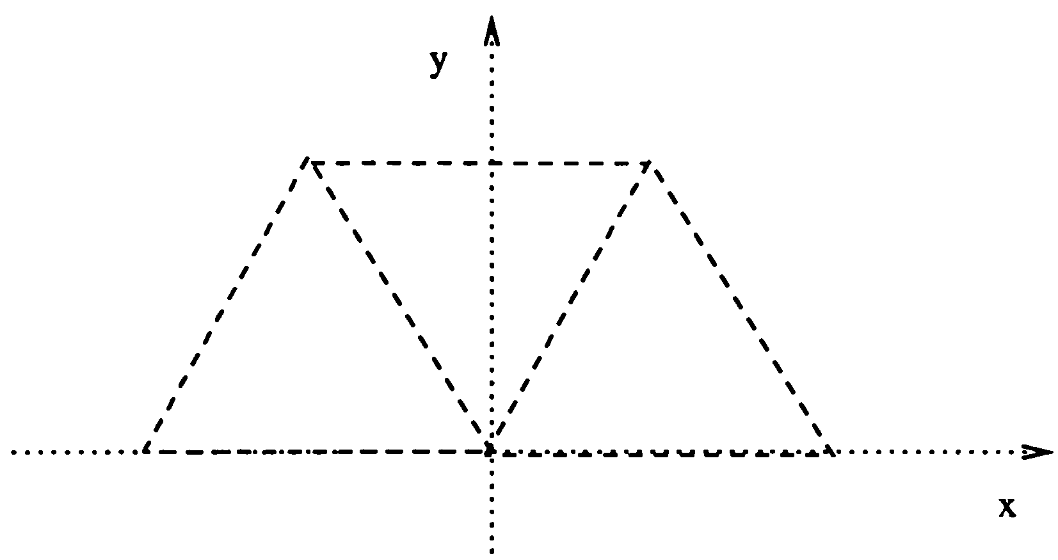

Figure 7.6: Two-dimensional plot of a PMF.

necessary to evaluate the discrimination function is at the corner points of all of the PMFs – of which there are only a finite number. A plot showing the lines of discontinuous slope for a typical PMF is shown in Figure 7.6, corresponding to an equilateral triangle of side length d, whose edge has been sensed at the origin of the coordinate system.

Noting that the function varies linearly along the lines of discontinuous slope, it can be further deduced that the maximum will occur at one of the corner points of the plot. These corner points in turn correspond to the vertices of the object model transformed by the limit values of the uncertainty range. For example, in the plot shown, the uncertainty range in the direction of the x-axis is $[-d, 0]$ and the corner points of the model are $(0,0)$, $(0,d)$ and $(d/2, \sqrt{3}d/2)$. This gives rise to the six corner points in the PMF (two of which are concurrent): $(-d,0)$, $(0,0)$, $(-d/2, \sqrt{3}d/2)$, $(0,0)$, $(0,d)$ and $(d/2, \sqrt{3}d/2)$.

The problem of determining the optimal location at which to sense has been simplified from an infinite search, to comparing the merits of a finite set of locations, readily calculable from the object models and uncertainty ranges. This enables an exact solution to be found on a digital computer in a finite time.

The algorithm used to implement this strategy is given in Algorithm 5. In the same manner as Algorithms 1 and 4, this algorithm undertakes an initialization routine consisting of reading the object models from a database, initializing the robot, and searching for and processing the first contact point (steps 1 to 4 of the pseudo-code). In this case, it is not necessary to obtain grid occupancy models for the objects, uncertainty ranges or PMFs. A loop is then entered in which the discrimination function is evaluated at potential maximal points (ie. those points corresponding to the transformed vertices of the object model), and further data is obtained from the resulting maximum. This algorithm has complexity of $O(n^2)$ in determining the optimal sensing location for an object set consisting of n sides. It is, however, still considerably faster than the previous algorithms (which were $O(n)$) providing n is not extremely large, as it is not necessary to evaluate the discrimination function at every point is a grid.

These techniques were applied to the binary information, location sensor de-

```
Identify:

  1. Read in object models, translate and rotate models to
     robot-based coordinate system, evaluate side-lengths and angle
     sizes and check for symmetries.

  2. Initialize and calibrate the robot.

  3. Search for and process the first contact point.

  4. Evaluate the object and contact side probabilities, and the
     uncertainty ranges.

  5. If the object is not identified:

  6.      Repeat:

  7.           Update the uncertainty ranges and contact side
               probabilities.

  8.           Evaluate the discrimination function at all possible
               maximum locations, and determine the maximum point as
               the optimal sensing location.

  9.           Move to and acquire data from the optimal location.

 10.           Update the object probabilities.

 11.      Until the object is identified.

 12. Return identity of object.
```

Algorithm 5: Efficient location sensing for object identification.

scribed in the first implementation. The strategies were again generated on-line in this implementation, but the speed improvements were significant. Actual savings over the first implementation are dependent on the specific model set and the coarseness of the grid used in the first implementation. However, for some implementations (such as that described in Section 7.6) that had required a processing time between each robot motion of over thirty seconds using the grid-based approach, the typical delay using the techniques described in this section was less than five seconds.

7.5 Off-line strategy generation

The significant reductions in processing time and data storage requirements provided by the methods described in the previous section suggested the possibility of generating the complete sensing strategy off-line, before any sensing has been performed. As generation of a complete strategy involves determining optimal sensing locations for all possible sensing outcomes (almost all of which would not be required for a specific identification task), off-line computation had been previously considered too computationally time-consuming and too demanding on the amount of data storage required. Using the techniques of Section 7.4 presented the opportunity to generate and investigate complete strategies.

A suitable representation for the complete strategy is a strategy tree, where each node represents a sensing location and each branch leading from a node corresponds to a different sensing outcome at that location. Once the tree has been generated, sensing can be performed and the next sensing location can be "looked-up" from the tree by following the appropriate branch corresponding to the sensed data.

The programme flow of the system for this implementation is shown in Algorithms 6 and 7. The algorithms generate the strategy tree recursively, by expanding each node, for each possible sensing outcome. Algorithm 6 consists of two parts: the off-line strategy generation; and the on-line sensing and recognition. The on-line recognition is similar to that presented in Algorithm 5, except that, in this case, it is only necessary to look-up the next sensing location from the strategy tree once sensory data has been obtained.

Algorithm 7 shows the data structures and recursive routines used to generate each node of the strategy tree. Each node has associated with it a sensing location, uncertainty ranges and object probabilities corresponding to that stage of the sensing process. If an object probability falls within some margin of one (so that the object can be identified), a flag is set to prevent the expansion of this node. Hence this node represents an end point for the strategy corresponding to recognition of the object. Expansion of the tree continues until all branches are terminated by nodes corresponding to end states.

Off-line generation also makes it possible to display and analyze complete sensing strategies. This provides an insight into the moves pursued by the robot and also the number of locations necessary to identify an object, both on average and in the worst case. To provide an understanding of the automatically generated strategy, and the reasoning behind the choice of each sensing location, a sample

```
Off-line strategy generation:

  1. Read in object models, and determine the a priori object
     probabilities.

  2. Generate strategy tree.

Identify:

  1. Initialize and calibrate the robot.

  2. Search for the first contact point.

  3. Obtain optimal first move from the node of the strategy tree.

  4. Repeat:

  5.       Move to and acquire data from the optimal location.

  6.       Look up new optimal location in strategy tree (by tracing
           branch from current location corresponding to acquired
           data.)

  7. Until object is identified (ie. a node of the tree is
     encountered).
```

Algorithm 6: Object identification with off-line strategy generation.

```
Generate tree:

struct node {
int node_number;
boolean stop;
float object_probabilities;
float contact_side_probabilities (for each object);
float uncertainty_ranges (for each object);
float sensing_location (x,y);
struct node *det, *ndet (branches);
} *root;
```

1. Assume edge contact, determine object_probabilities, contact_side_probabilities, uncertainty range and assign these values to root->object_probabilities root->contact_side_probabilities and root->uncertainty_ranges.

2. Call generate_next_layer(root).

3. Call recursively_grow_tree(root)

Algorithm 7 (continued over): Generation of strategy tree.

strategy generated by the system to discriminate between a set of shapes, is described in Section 7.6.

As there was no processing performed on-line, this system was the fastest of the four implementations, (not considering the time required for off-line processing). The strategies generated by this system were the same as those generated by the first and third implementations.

7.6 Analysis of a sensing strategy

Consider the task of sensing to discriminate between the shapes shown in Figure 7.7. The strategy tree generated automatically for this task by the system is shown in Figure 7.8. The coordinate locations referred to in the tree are relative

```
generate_next_layer(leaf):
struct node *leaf;
```

1. Evaluate discrimination function at potential maximum points (using leaf->object_probabilities, leaf->contact_side_probabilities, and leaf->uncertainty ranges to determine the optimal next sensing location and set leaf->sensing_location to the optimal location.

2. Assume contact at optimal sensing location, determine a posteriori object probabilities and assign these to leaf->det->object_probabilities. If the object is identified at this stage (ie. one of the object probabilities is within some error margin of 1) set leaf->det->stop. If (!leaf->det->stop) determine and assign values to leaf->det->contact_side_probabilities and leaf->det->uncertainty_ranges.

3. Assume no contact at optimal sensing location, determine a posteriori object probabilities and assign these to leaf->ndet->object_probabilities. If the object is identified at this stage set leaf->ndet->stop. If (!leaf->ndet->stop) determine and assign values to leaf->det->contact_side_probabilities and leaf->det->uncertainty_ranges.

```
recursively_grow_tree(leaf):
struct node *leaf;

1. If (!leaf->det->stop):
2.     generate_next_layer(leaf->det).
3.     recursively_grow_tree(leaf->det).
4. If (!leaf->ndet->stop):
5.     generate_next_layer(leaf->ndet).
6.     recursively_grow_tree(leaf->ndet).
```

Algorithm 7 (cont): Generation of strategy tree.

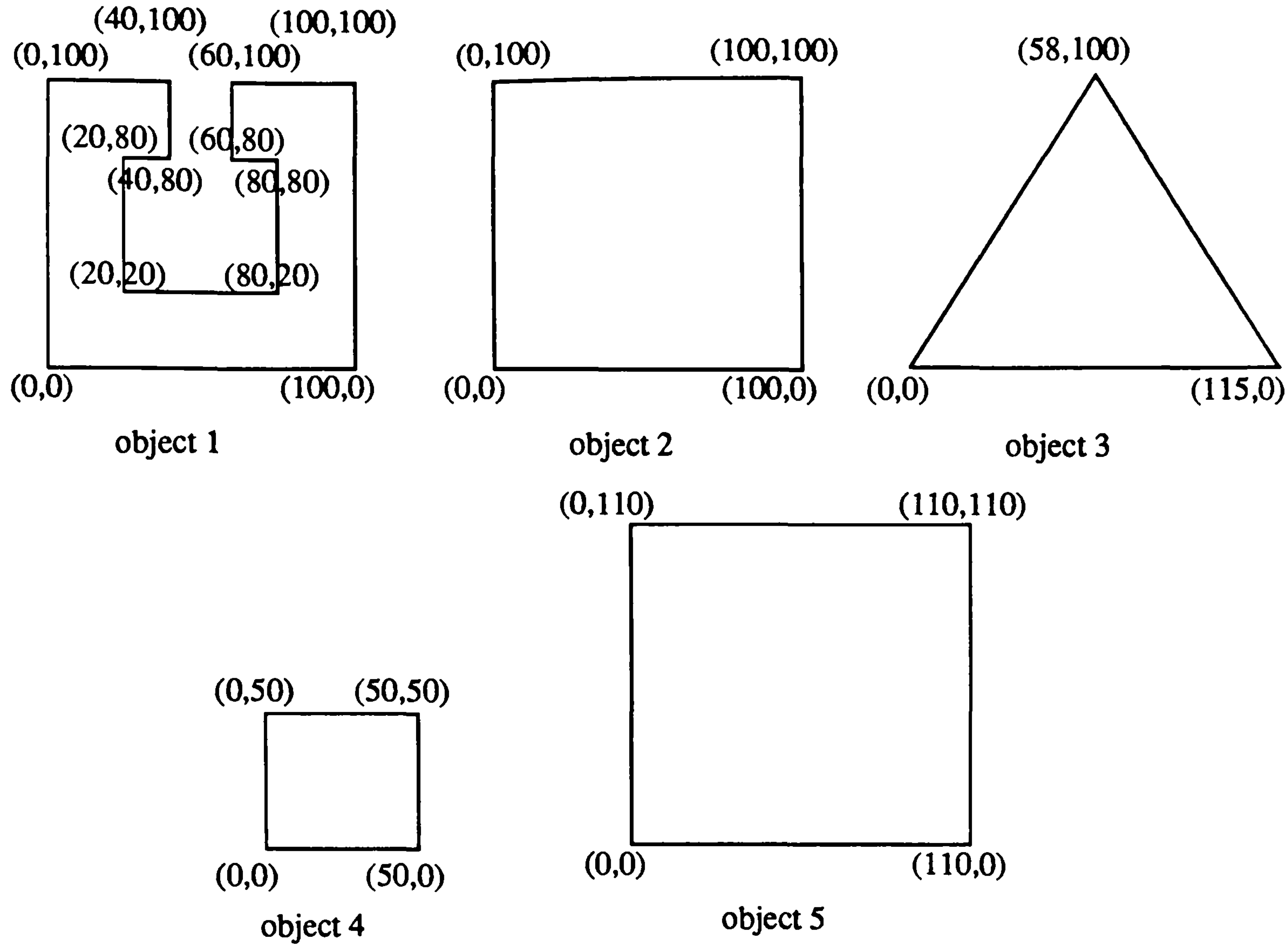

Figure 7.7: Object set.

to a coordinate system referenced to the first sensing location (located at $(0,0)$).

This example was chosen because it demonstrates the full range of capabilities of the location sensing strategy described in Sections 7.2, 7.4 and 7.5. The model set is chosen to include both convex and concave objects; simple and complex objects (such as *object* 1); objects that are similar but differ slightly in size (*object* 2 and *object* 5); and objects that are dissimilar in size to others in the set (*object* 4).

To make some allowance for finite sensor scope, the system chooses locations a small distance (in this case 1 unit) from potential object edges, to ensure that sensing will take place either inside or outside the object's boundaries.

The root node of the sensor corresponds to the first sensed point. It is assumed

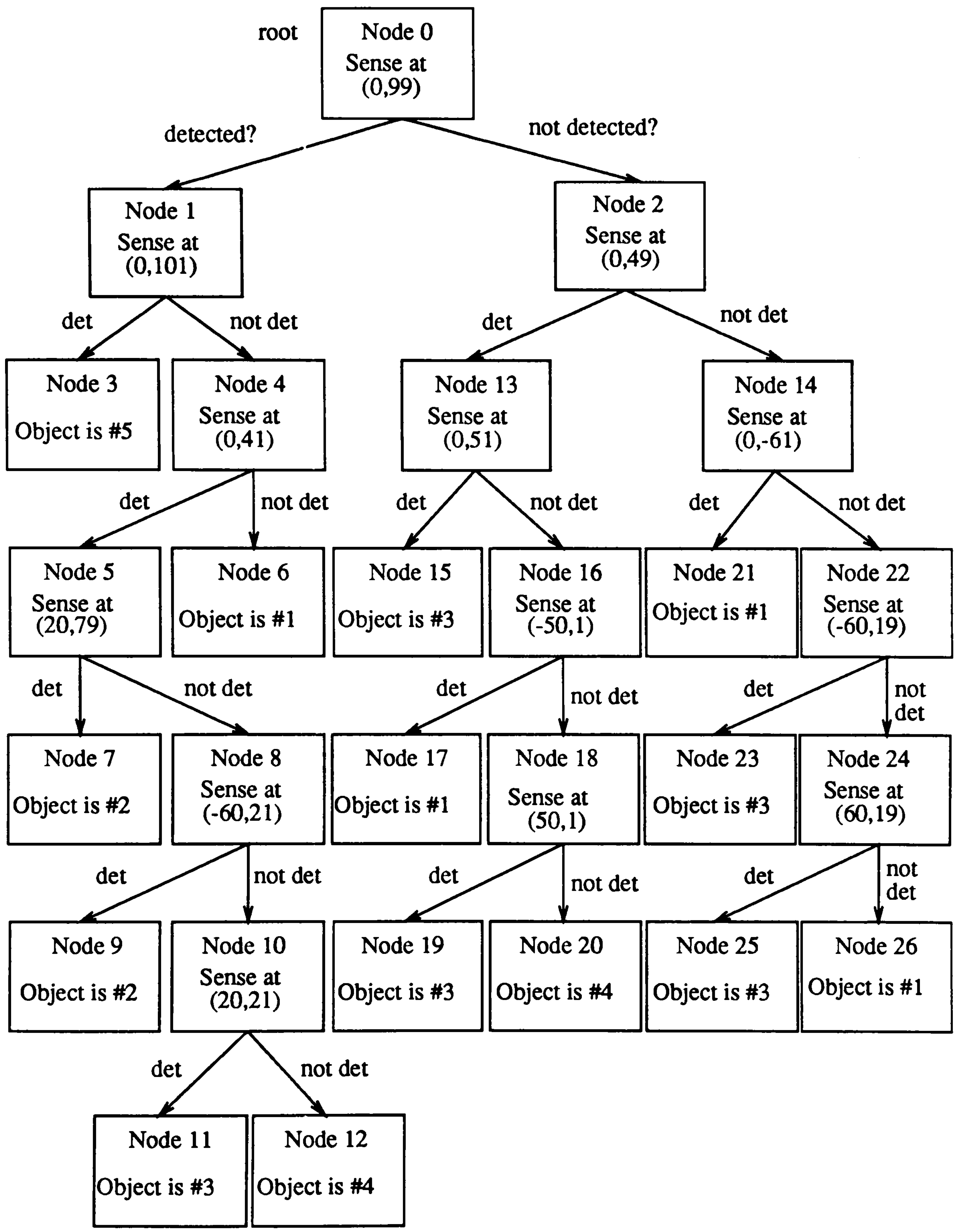

Figure 7.8: A typical strategy tree.

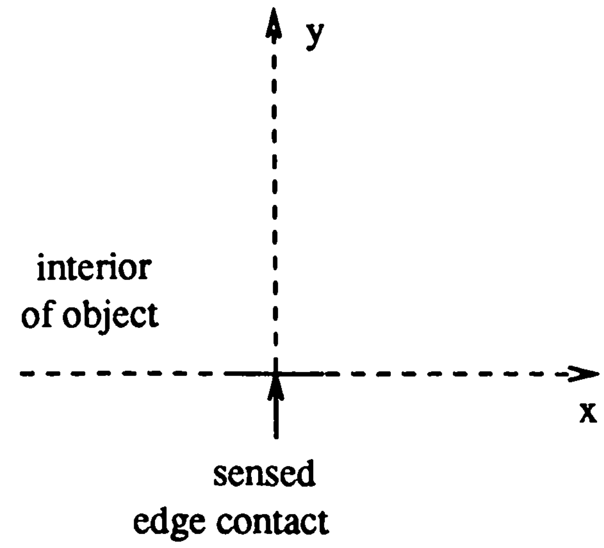

Figure 7.9: Definition of coordinate system.

that this point represents an edge contact with the object. This point and the sensed edge define the coordinate system used in the strategy. This system is defined by Figure 7.9.

The point contact is the origin of the system, and an inwardly directed normal to the object surface at this point constitutes the y-axis. The x-axis is then defined in the normal right-handed sensed with respect to the y-axis.

As data from the first sensed point is assumed known, there is only one branch leading from the root node. In other words, the location of the second sensing point is predefined, relative to the first contact location. An analysis of this choice of location and the subsequent ones chosen can now be performed.

Node 0: Sense at (0,99)

After the first sensed point has been obtained, the strategy chosen by the system is to sense next at the point (0,99).

If the object is not detected at this location, it cannot be either *object* 2 or *object* 5, both of which would be certain to be detected at this point. The remaining admissible interpretations would be *object* 1, *object* 3 and *object* 4.

Note that certain interpretations assigning sides of *object* 1 to the first sensed point can be excluded on the basis of this data, and the uncertainty ranges for other sides, and for the sides of *object* 3, are restricted.

To discriminate between these remaining interpretations the sensor proposes sensing at the location (0,49) described in Node 2.

If the object is detected at (0,99), it is clearly not *object* 4. Further, it is highly unlikely to be *object* 3, whose admissible uncertainty range is severely restricted by detection at this point. Certain assignments of edges of *object* 1 to the first sensed point can also be excluded. To reduce the uncertainty in object identity further the next sensed point suggested for this case is (0,101) [Node 1].

Node 2: Sense at (0,49)

The location (0,49) is chosen to test the hypothesis that the object is *object* 4, which is the prime candidate after the first two sensed points, as no restrictions on its range were imposed by the failure to detect at (0,99).

Clearly failure to detect at (0,49) will remove *object* 4 as a candidate. It will also severely restrict the possible placements of *object* 3, and leave *object* 1 and *object* 3 as the only remaining interpretations. To resolve between these alternatives, the system proposes sensing at (0,-61) [Node 14].

On the other hand, detection at (0,49) will strengthen confidence in *object* 4 as the correct interpretation, but does not leave it as the sole candidate. In this case, *object* 1 can be eliminated as all interpretations of *object* 1 that would be consistent with the initial sensed point and detection here, imply detection at

(0,99), which has already been discounted. Thus the remaining interpretations are *object* 3 and *object* 4, and the automatically generated strategy is to sense at (0,51) [Node 13] to discriminate between them.

Node 14: Sense at (0,-61)

Sensing below the x-axis is an attempt to identify the object as *object* 1, as this is the only concave object, and hence the only object that may exist in this region. Hence, detection here will enable the object to be identified as *object* 1.

Failure to detect will eliminate several possible interpretations of *object* 1, and will increase the expectation of *object* 3. These two interpretations remain and the next point proposed to discriminate them is (-60,19) [Node 22].

Node 22: Sense at (-60,19)

Due to the restrictions placed on the possible range of locations of *object* 1 proposed by the previous sensory data, it is no longer possible for this object to be present at this location. On the other hand, previous restrictions on the location range of *object* 3 have made this a strongly probable location at which to sense this object (although not certain).

Detection at this point implies the object is *object* 3. Failure to detect constrains the locations the locations of both significantly that they can be discriminated by a single point, namely (60,19) [Node 24].

Node 24: Sense at (60,19)

By this stage the object models for the two remaining interpretations are restricted to very small ranges of uncertainty. At this location, detection is assured for *object* 3 and impossible for *object* 1.

Hence, detection at this point implies that the object is *object* 3; failure to detect implies that it is *object* 1.

Node 13: Sense at (0,51)

At this stage the remaining admissible interpretations for the sensed object are *object* 3 and *object* 4.

Clearly, *object* 4, (which has a height of 50 units) cannot be present at both (0,0) and (0,51). Thus detection at this point implies that the object is the triangle, *object* 3.

However, detection of *object* 3 is not assured at this point, and hence failure to detect does not identify the object. In this case, the next proposed sensing point is (-50,1) [Node 16].

Node 16: Sense at (-50,1)

As the width of the small square represented by *object* 4 is only 50 units, it is impossible to detect it at both the origin and (-50,1). Hence, this location, just beyond the admissible range for *object* 4, is chosen as a sensing location as it corresponds to the point of maximum probability of detecting *object* 3 for which there is no possibility of detecting *object* 4. This corresponds to the sensor strategy of sensing at the point corresponding to the maximum difference in the PMFs for the two objects.

If the object is detected at this location, it must be *object* 3. If no detection occurs, further sensing is necessary at (50,1) [Node 18].

Node 18: Sense at (50,1)

Consideration of the PMF for the triangle *object* 3 before sensing was performed at (-50,1) [Node 16], would reveal that if the identity of the object was *object* 3 it must be present at either (50,1) or (-50,1), corresponding to the two admissible uncertainty ranges at that stage. As one of these ranges has been eliminated by sensing at (-50,1), the point corresponds to a certain detection location if the object is indeed *object* 3.

For the same reasons as discussed at Node 16, it is impossible for the small square *object* 4 to be present at (50,1).

Hence detection of the object is guaranteed by sensing at this location: If the object is detected it must be *object* 3; if it is not detected it must be *object* 4.

Node 1: Sense at (0,101)

If the first sensed point (at (0,99)) results in the detection of the object, four admissible interpretations remain, namely: *object* 1, *object* 2, *object* 3 and *object* 5. The simplest of these to test for is *object* 5, which is certain to be detected at (0,101), whilst none of the other objects can be detected at this point. This is indeed the next point nominated by the system. If detection occurs here, the object must be *object* 5; if detection does not occur, the object can be one of *object* 1, *object* 2 and *object* 3 and the next sensing location proposed in this event is (0,41) [Node 4].

Node 4: Sense at (0,41)

Previous sensor data has constrained the uncertainty in location of the triangle *object* 3 such that it is certain of being detected at (0,41). The same is true of the square *object* 2. Hence, the only admissible object at this stage which is not assured of detection at his location is *object* 1, which because of its concavity, may be sensed at (0,0) and (0,99) without necessitating detection at (0,41).

If the object is not sensed at (0,41), its identity must be *object* 1. If detection does occur however, all three objects (*object* 1, *object* 2 and *object* 3) are still admissible. To aid in their identification, the next proposed location is (20,79) [Node 5].

Node 5: Sense at (20,79)

Previous sensory data has confined the PMFs of *object* 1 and *object* 3 sufficiently that they are both by this stage zero at this location. The only object that will give rise to detection at this location (and in fact has a high expectation of detection) is *object* 2. This object is also the most likely interpretation of the sensory data thus far, having all its initial potential interpretations admissible to all the subsequent sensory data. Thus there is a reasonably high probability of detection (and hence identification) of the object at his location.

If the object is detected at (20,79), it must be *object* 2. If no detection occurs, there are still three admissible objects and little progress has been made (other than to reduce the range of uncertainty of *object* 2). The next proposed sensing location in this case is (-60,21) [Node 8].

Node 8: Sense at (-60,21)

If the square *object* 2 was not present at (20,79) it must be present at (-60,21). However, neither of the other admissible objects can be detected here.

Hence, if the object is detected at (-60,21), it can be identified as *object* 2. If no detection occurs, only *object* 1 and *object* 3 remain, and they can be discriminated by sensing at (20,21) [Node 10].

Node 10: Sense at (20,21)

By this stage, the respective locations of the two objects have been reasonably accurately determined. At his location, it is known that the triangle *object* 3 is certain to be detected, whilst *object* 1 is certain not to be detected.

Thus, detection at this point implies that the object is *object* 3; otherwise it must be *object* 1.

7.7 Discussion

All the experiments performed led to the sensed object being correctly identified by the system. This, however, can also be achieved with unguided sensing systems, which obtain data in a random manner, until sufficient information has been acquired to recognize the object. The aim of the work described here, was to develop systems that perform the recognition task in an optimally efficient fashion. Specifically the objective was to recognize the object on the basis of the minimum number of sensing actions.

On all occasions the chosen strategies survived the scrutiny of human "common sense", in that we were unable to justify an alternative sensing action to that generated automatically at any stage of the process. Unfortunately, no theory has been devised to analyze the optimality of the strategies chosen. However, the strategies can always be justified statistically. This ability to explain each step of the sensing process as being preferable to any alternative action, implies that the strategy is optimal, **providing** that the PMFs are accurate and the utility functions are themselves accurate measures. As the utility functions are subjective, it does not follow that the chosen strategies are optimal, however it seems clear that well-justified utility functions will result in good sensing strategies. The justification for the utility function used in the implementations described in this chapter was given in Chapter 5.

The methods described in Section 7.4 to enable efficient strategy generation are an important step in justifying the use of guided sensing over random data acquisition. To be truly cost effective, the processing time required to generate the sensing strategy must be small in relation to the saving in time spent on additional sensing actions. This sensing action time includes time spent relocating the sensor, acquiring data, processing the data, and updating of models. Clearly, these will vary between specific actions. For example, for the implementations described in this chapter, there were significant delays in sensing (of the order of seconds), mainly involved with relocating the sensor and robot arm. In this case, spending several seconds on calculating an efficient strategy, which will reduce the number of arm movements, appears well-justified – however, spending tens of seconds on the strategy, as with the implementations described in Sections 7.2 and 7.3, does not. It is also possible to imagine situations where data acquisition is fast and efficient, and the time spent on strategy planning may seem unnecessary.

Further, the savings in the number of sensory actions to be anticipated from guided sensing are dependent on the particular objects present in the object set. A set of objects which differ only by a small feature, may benefit greatly from guided sensing to quickly track down the salient feature, as such a feature may elude detection by random methods. On the other hand, a set of markedly different objects may be swiftly identified without any guided sensing strategy.

Unfortunately, it was difficult to apply the methods of Section 7.4 to the path sensing problem of Section 7.3. This was due to the nature of the PMF in the polar coordinate system. It was shown in Section 7.4 that the linearity of the PMFs in the x- and y-dimensions enables the investigation of only a finite number of points as potential maximums of the discrimination function. Although the PMF in polar coordinates is linear in the r-dimension, it is not linear with θ. Some other method is required to determine the maximum of the discrimination function as a function of θ. Ideas that offer promise include some form of segmented search, or a numerical estimation method.

Chapter 8

Conclusions and further work

8.1 Analysis of implementations

The implementations described in Chapter 7 supported the viability of using a statistical decision theoretic approach to develop an active sensing strategy. To be truly useful it must be demonstrated that there are real savings (in time or some other cost) associated with pursuing an intelligent data acquisition approach. If the amount of processing time required to determine the optimal sensing actions is significantly greater than the savings in time obtained by the reduced number of sensing locations necessary, then this approach may be unjustifiable. (An exception may be where it is desired to reduce the amount of sensory interaction, to avoid damaging a fragile environment.)

For this reason the implementations addressed in Sections 7.4 and 7.5 were particularly significant, as they tested methods which offered substantial reductions in the amount of processing required at each stage of the intelligent sensing process. In particular, the fourth implementation (Section 7.5), because it utilized off-line strategy generation, required only a tree look-up to determine each subsequent sensing action, and hence could be expected to be as fast between each sensing action as a random sensing strategy, with the obvious benefit of an anticipated reduction in the required number of sensing actions.

It would be desirable to have some theory describing the expected saving (in number of sensing actions) that can be anticipated by using a guided sensing strategy in comparison with random data acquisition. Such a measure is necessary to enable a quantitative analysis to be performed of the merit of applying these techniques to a specific sensing problem. Apart from the observation that the savings will depend on the specific model set, and the "unguided strategy" available, no appropriate theory has been proposed. An appropriate theory would have to be statistical in nature, to take into account the relative location of the object to the first sensed point, which will also affect the number of actions necessary to identify the sensed object in any particular instance.

In general, it can be concluded that guided sensing is likely to prove most beneficial in the following applications:

- Where the acquisition or processing of sensory data is relatively time-consuming;
- Where the objects in the admissible set are very similar in size or shape;
- Where the environment is fragile and is potentially damaged by each sensing action.

8.2 Sensing in Three Dimensions

8.2.1 Introduction

All the theory developed in this thesis has been restricted to two-dimensions. There is no theoretical difficulty in extending these principles to three-dimensional problems, however, a naive extension will result in a significant increase in data processing requirements, and it is unlikely that the much greater time required to generate the optimal sensing strategy will be justifiable.

The increase in data processing is due to more degrees of uncertainty in the relative location of the sensor to the object, and more parameters to be specified in selecting the optimal sensing action. The greater uncertainty in the relative position of the sensor and object leads to more degrees of uncertainty represented by the feature uncertainty functions. This results in more intensive computations to determine the object probability maps in three dimensions, however, the same utility functions and sequential analysis will be appropriate once the PMFs have been determined. Hence no discussion in this section is directed at the utility functions or sequential techniques as these do not differ between the two and three dimensional treatments.

It was discussed in Chapter 4 how the number of dimensions in which there is uncertainty after contact between a sensor and an object depends on the nature of the contact. If a vertex has been detected, the location of the object may be accurately specified. If an edge of the object has been located (this is the case primarily addressed in the two dimensional analysis), there is uncertainty along the direction of the contact edge. When dealing with three dimensional objects, it is most likely that the first point of contact will be on a face of the object, resulting in three degrees of uncertainty: two translational and one rotational.

Another problem of addressing the sensor placement problem in three dimensional space is the increase in parameters necessary to specify uniquely the sensing action. In three dimensions, it is more natural to consider only the path sensing paradigm discussed in Chapter 4. (This is because it is impossible to sense at a location without actually moving the sensor to the desired location, and this will not be possible if the specified position is within the interior of the object. The only reason that location sensing was possible in two dimensional space, without concern for the path, was that it was possible to direct the sensor to the desired location from outside the plane of interest. It is clearly not possible to sense in three dimensional space from outside of that space.) Assuming that sensing paths are restricted to rays, it requires four parameters to characterize a line in three dimensions, and hence the choice of sensing action must be maximized in a four dimensional sensor action space.

Thus, although an uncertainty map can be developed for the location of the object in three dimensional space, a four dimensional map is necessary to define the PMF in sensing space coordinates. It can be argued that in sensing the surface of an object the problem is in essence only two dimensional as object surfaces can be readily mapped into two-dimensional space. This argument considers only the contact point between the sensing ray and the surfaces and neglects the approach directions of the rays, which will be important, particularly if it is intended to sense concave objects possessing occluded faces. This four dimensional analysis clearly requires much greater data storage and processing time.

Different strategies may be proposed if the sensor were allowed to track in a given direction to detect an edge. Once an edge has been detected there is only uncertainty in the relative locations of the sensor and the object along the direction parallel to the sensed edge. All the analysis in this thesis, however, has relied on minimizing the number of sensing actions. (An exception has been in obtaining the first contact point when there has been no prior information to guide the search.) Sensor tracking will by nature involve continuous sensing (often approximated by discrete sensing at incremental steps) and hence multiple discrete sensing actions. To maintain consistency with the previous approach, the technique of tracking until an edge is encountered is not advocated here as a method of reducing the uncertainty to the same levels (one degree of translational uncertainty) already dealt with in the two dimensional analysis.

In the following subsection, it will be assumed that the only three dimensional objects that can be encountered are polyhedral objects, which have been initially contacted on a planar face.

The constraints imposed by sensing objects of arbitrary shape are discussed in Subsection 8.2.3.

8.2.2 Contacting a planar face

Once a face of a polyhedral object has been contacted, the general location of the object in space has been ascertained, and bounds upon its translational position can be established. However there is complete uncertainty about the rotation between the available model and the sensed object about the normal vector to the sensed surface. Hence, although some strategy can be developed about the distance to sense from the current location, the chosen direction relative to the normal vector at the first contact point must be arbitrary.

As there is complete uncertainty in the location of the object with respect to rotation about the surface normal, the uncertainty map of the location of the object will be completely symmetrical in this dimension (say θ). Hence (as it has no effect on the determination of the probability of detection) it is possible to neglect this parameter completely and define an uncertainty map in two variables: distance form the first sensed point in the tangent plane of the surface; and perpendicular distance from this plane. This symmetry can also reduce the dimensionality of the problem in path sensing coordinates, as rays which are equivalent under a transformation about the normal vector, will give rise to the same probabilities of detection. Hence, by defining an appropriate coordinate system, the four dimensional path sensing problem can be reduced to a three dimensional choice over alternative actions.

Clearly as more data is acquired, only certain ranges of θ will be consistent with the sensory data. At this stage, the problem will revert to the more general three dimensional case, with a decision over four parameters necessary to define the optimal sensing action.

8.2.3 Objects of arbitrary shape

In the previous subsection, strategies were discussed to reduce the three degrees of uncertainty present when a polyhedral object is contacted on a planar face. Three represents the maximum number of degrees of uncertainty that can result when the surface of a three-dimensional object is encountered, however, for some non-planar faces, the number of degrees of uncertainty may be lower.

In particular, if a face has different measures of curvature along two axes in the tangent plane at the point of contact, then (assuming that the sensor employed can discern the curvature[1]) these axes can be aligned with the corresponding axes in the object model to remove the degree of rotational uncertainty. Hence, in these cases there are only two degrees of uncertainty,[2] and the position of the object in space is much more accurately known. (In this subsection, we will only deal with faces which have constant curvature relative to some coordinate system.)

An example of such a face is the curved face of a cylinder, whose curvature around its axis is equal to $1/r$ (where r is the radius of the cylinder), but has zero curvature along its length.[3] An example of a curved face with three degrees

[1] As curvature is a property of the second derivative of the object surface, the ability to discern curvature at a point implies the ability to sense over a finite surface area around the point of contact. This regional sensing may be performed at one instant by a sensor array, or sequentially by accumulating point samples from the surrounding area.

[2] For a curved surface, the two degrees of uncertainty will not in actual fact be translational, but those corresponding to axes of non-zero curvature will be rotational about the centre of curvature (ie. the point which subtends the arc of local curvature.)

[3] The curvature of a curve $y = y(x)$ is given (see Faux and Pratt [18]) by the

of uncertainty is the face of a sphere, which has curvature equal to $1/r$ (where r is the radius of the sphere) relative to all axes in its tangent plane.

Not only will different axes of curvature reduce uncertainty in the location of the sensed object, but the ability to detect local curvature of a surface will in itself be an aid in identifying the object [2]. Just as edges of different slope were seen to aid in identification in the two dimensional implementations (see Section 7.3), it is clear that curvature will exhibit saliency in three dimensional recognition problems.

8.2.4 Conclusions

The application of the decision theoretic principles to three dimensional problems is an ongoing area of our research, and the results and ideas discussed here are only preliminary. No opportunity has yet been taken to test these concepts with an implementation.

Despite the difficulties of increased dimensionality that occurs in three dimensions, it is felt that by developing a theory that is sympathetic to the particular sensor used, the problem will reduce in complexity. For example, it may not be possible to specify all four configuration parameters for the specific sensor at hand, and hence the dimensionality of the problem will be reduced. Alternatively, if the object set consists only of convex objects, the direction of approach to a

formula

$$\rho = \frac{(1 + y'^2)^{3/2}}{y''}$$

face of a particular sensing action may not be important, and it may be possible to describe each sensing action by its point of contact with the plane in which the face lies.

These are aspects which we hope to investigate further in the future.

8.3 Refinements and generalization of techniques

The most significant work that needs to be done with the approach described in this thesis is to develop and then to apply and test the theory proposed in the previous section for three dimensional sensing problems. Such an experiment will require a contact sensor or a range sensor with a narrow field of view, which can be relocated in a three dimensional workspace.[4] This was beyond the capabilities of any hardware available for this project.

Work is also required to determine a less arbitrary approach to generating appropriate utility functions for a given task, along with methods of comparing proposed utility functions and defining and measuring their "optimality". The subjective nature of the utility functions appears to preclude the existence of a single optimal utility function for a task, but it is hoped that general guidelines can be established defining appropriate features for such functions.

Finally an analysis is desirable, as discussed in Section 8.1, providing a statistical analysis of the expected reduction in sensing actions obtained by employing

[4] As the object to be sensed will almost certainly be resting on a planar surface, it is probably only necessary for the sensor to be located in a three-dimensional half-space.

a particular utility function with a given object set as compared to a known unguided strategy.

8.4 Significant contributions

In conclusion, it is appropriate to itemize the significant contributions of the work described in this thesis to the field of robotic sensing. (In fact, it is hoped that the techniques developed here will find application in sensing problems outside of the area of robotics.)

1. The major contribution is a structured, mathematical approach to the problem of optimal sensor placement. Rather than acquiring sensory data at a random location, the described method evaluates all possible sensing locations for their ability to constrain the model uncertainties, and senses at the chosen location. After sensory data has been acquired, it is used to update all expectations about the sensing environment, which can then be used to order possible interpretations, and as prior information in chosing the next sensing location.

   These techniques should have particular application in systems in which the sensory data is inherently "sparse" and multiple sensory operations are required. Such sensory systems include tactile sensing and sonar, and potential tasks include object identification and localization, and robot navigation.

This problem has generally been neglected by previous researchers or treated in an *ad hoc* fashion. In this thesis, established mathematical principles of Bayesian decision theory are adapted to the problem of sensing in an uncertain environment, leading to justifiable sensing strategies. Further, these principles have been implemented on a robot sensing system, and found to be viable in practice.

Although further theoretical analysis is required in some areas (providing a greater understanding of the likely economies in number of sensing actions for a given set of objects that can be anticipated using guided sensing, and a less subjective generation of utility functions leading to a more objective strategy), the work described in this thesis provides a groundwork on which this further analysis can be based.

2. The second major contribution is the representation of uncertainty at each stage of the sensing process. This representation can be explained in terms of the probabilistic interpretation tree described in Chapter 6. Previous researchers have generally chosen to classify potential interpretations of sensor data to object models as either admissible or inadmissible. By generating expectations of alternative sensing outcomes (necessary for the strategy generation process) it is possible to update and maintain probabilistic measures of alternative interpretations. This in turn allows the techniques of sequential analysis to be applied to the sensing problem, and allows conclusions to be given within stated error bounds.

The techniques of strategy generation and interpretation representation described here, are independent but share a significant amount of computation.

Appendix A

Properties of the PMF

When $S(x, y) = \delta(x, y)$, the defining equation for the probabilistic membership function in Cartesian space becomes:

$$\Pi(x, y) = O'(x, y) * F(x, y)$$

Expanding the convolution gives

$$\begin{aligned}
\Pi(x, y) &= \iint O'(x - \mu, y - \eta) F(\mu, \eta)\, d\mu\, d\eta \\
&\leq \iint 1\, F(\mu, \eta)\, d\mu\, d\eta \\
&= \iint F(\mu, \eta)\, d\mu\, d\eta \\
&= 1 \quad (\text{as } F \text{ is a pdf})
\end{aligned}$$

Therefore,

$$\Pi(x, y) \leq 1 \;\; \forall (x, y)$$

Also,

$$\Pi(x,y) \geq 0 \;\; \forall(x,y) \quad \text{(as } O' \text{ and } F \text{ are strictly non-negative)}$$

Thus,

$$0 \leq \Pi(x,y) \leq 1 \;\; \forall(x,y)$$

Integrating Π over the whole space:

$$\begin{aligned}
\iint \Pi(x,y)\,dx\,dy &= \iint O'(x,y) * F(x,y)\,dx\,dy \\
&= \iint \left[\iint O'(x-\mu, y-\eta)F(\mu,\eta)\,d\mu\,d\eta\right] dx\,dy \\
&= \iint \left[\iint O'(x-\mu, y-\eta)\,dx\,dy\right] F(\mu,\eta)\,d\mu\,d\eta \\
&= \iint \;[\text{area of } o]\; F(\mu,\eta) d\mu\,d\eta \\
&= [\text{area of } o] \iint F(\mu,\eta)\,d\mu\,d\eta \\
&= \text{area of } o
\end{aligned}$$

Thus the probabilistic membership function, Π, satisfies the same conditions as the same conditions as the object model, O, and can be considered as an "uncertain" model of the object.

Appendix B

Evaluation of discrimination functions

B.1 Recognition from a two-object set (Case 1)

Consider the proposed utility function for the task of identifying an object from the set $\{o_1, o_2\}$:

$$U(state, (x,y)) = |2\,P(o_1|state) \;-\; 1|$$

The discrimination function (assuming only two sensing states) is:

$$\begin{aligned} D(x,y) &= U(detection = 1, (x,y))\,P(detection = 1) \\ &\quad + U(detection = 0, (x,y))\,P(detection = 0) \end{aligned}$$

Consider the term,

$$\begin{aligned} & U(state, (x,y))\,P(state) \\ &= |2\,P(o_1|state) - 1|\,P(state) \end{aligned}$$

$$
\begin{aligned}
&= |2\,\frac{P(state|o_1)\,P(o_1)}{P(state)} - 1|\,P(state) \\
&= |2\,P(state|o_1)\,P(o_1) - P(state)| \quad (\text{as } P(state) \geq 0 \text{ always}) \\
&= |2\,P(state|o_1)\,P(o_1) - (P(state|o_1)\,P(o_1) + P(state|o_2)\,P(o_2))| \\
&= |P(state|o_1)\,P(o_1) \; - \; P(state|o_2)\,P(o_2)|
\end{aligned}
$$

Substituting into the discrimination function,

$$
\begin{aligned}
D &= U(detection = 1, (x, y))\,P(detection = 1) \\
&\quad + U(detection = 0, (x, y))\,P(detection = 0) \\
&= |P(detection = 1|o_1)\,P(o_1) \; - \; P(detection = 1|o_2)\,P(o_2)| \\
&\quad + |P(detection = 0|o_1)\,P(o_1) \; - \; P(detection = 0|o_2)\,P(o_2)| \\
&= |\Pi_{o_1} P(o_1) - \Pi_{o_2} P(o_2)| \; + \; |(1 - \Pi_{o_1})P(o_1) - (1 - \Pi_{o_2})P(o_2)|
\end{aligned}
$$

If $P(o_1) = P(o_2)$ then

$$
\begin{aligned}
D &= |\Pi_{o_1} P(o_1) - \Pi_{o_2} P(o_2)| \; + \; |(1 - \Pi_{o_1})P(o_1) - (1 - \Pi_{o_2})P(o_2)| \\
&= |1/2\,\Pi_{o_1} - 1/2\,\Pi_{o_2}| \; + \; |1/2\,(1 - \Pi_{o_1}) - 1/2\,(1 - \Pi_{o_2})| \\
&= 1/2|\Pi_{o_1} - \Pi_{o_2}| \; + \; 1/2|(1 - \Pi_{o_1}) - (1 - \Pi_{o_2})| \\
&= |\Pi_{o_1} - \Pi_{o_2}|
\end{aligned}
$$

B.2 Recognition from a two-object set (Case 2)

Consider the proposed utility function for the task of identifying an object from the set $\{o_1, o_2\}$:

$$U(state, (x, y))$$

$$
\begin{aligned}
&= |P(o_1|state) - P(o_1)| + |P(o_2|state) - P(o_2)| \\
&= |P(o_1|state) - P(o_1)| + |(1 - P(o_1|state)) - (1 - P(o_1))| \\
&= 2\,|P(o_1|state) - P(o_1)|
\end{aligned}
$$

The discrimination function (assuming only two sensing states) is:

$$
\begin{aligned}
D(x,y) &= U(detection = 1, (x,y))\,P(detection = 1) \\
&\quad + U(detection = 0, (x,y))\,P(detection = 0)
\end{aligned}
$$

consider the term,

$$
\begin{aligned}
&U(state, (x,y))\,P(state) \\
&= 2\,|P(o_1|state) - P(o_1)|\,P(state) \\
&= 2\,|\frac{P(state|o_1)\,P(o_1)}{P(state)} - P(o_1)|\,P(state) \quad \text{(by Bayes Rule)} \\
&= 2\,P(o_1)\,|P(state|o_1) - P(state)| \\
&\quad \text{(as } P(state) \text{ and } P(o_1) \geq 0 \text{ always)} \\
&= 2\,P(o_1)\,|P(state|o_1) - (P(state|o_1)\,P(o_1) + P(state|o_2)\,P(o_2))| \\
&= 2\,P(o_1)\,|P(state|o_1)\,(1 - P(o_1)) - P(state|o_2)\,P(o_2)| \\
&= 2\,P(o_1)\,P(o_2)\,|P(state|o_1) - P(state|o_2)| \quad \text{(as } P(o_2) \geq 0 \text{ always)} \\
&= 2\,P(o_1)\,P(o_2)\,|\Pi_{o_1} - \Pi_{o_2}|
\end{aligned}
$$

Using this result to evaluate the discrimination function:

$$
\begin{aligned}
D &= U(detection = 1, (x,y))\,P(detection = 1) \\
&\quad + U(detection = 0, (x,y))\,P(detection = 0) \\
&= 2\,P(o_1)\,P(o_2)\,|\Pi_{o_1} - \Pi_{o_2}| + 2\,P(o_1)\,P(o_2)\,|\Pi_{o_1} - \Pi_{o_2}|
\end{aligned}
$$

$$= \quad 4\, P(o_1)\, P(o_2)\, |\Pi_{o_1} - \Pi_{o_2}| \tag{B.1}$$

As $P(o_1)$ and $P(o_2)$ are constant over (x, y), it follows that:

$$D(x, y) \propto |\Pi_{o_1}(x, y) - \Pi_{o_2}(x, y)|$$

and the optimal sensing location will be the value of (x, y) where the difference between the PMFs for the two objects is greatest.

B.3 Recognition from an n-object set (Case 1)

Consider the proposed utility function for the task of identifying an object from the set $\{o_1, o_2, ..., o_n\}$:

$$U(state, (x, y)) = \sum_i |n\, P(o_i|state) - 1|$$

The discrimination function (assuming only two sensing states) is:

$$\begin{aligned} D(x, y) \quad &= \quad U(detection = 1, (x, y))\, P(detection = 1) \\ &\quad + U(detection = 0, (x, y))\, P(detection = 0) \end{aligned}$$

Consider the term,

$$\begin{aligned} & U(state, (x, y))\, P(state) \\ = \quad & \left(\sum_i |n\, P(o_i|state) - 1|\right) P(state) \\ = \quad & \sum_i (|n\, P(o_i|state) - 1|\, P(state)) \\ = \quad & \sum_i \left(|n\, \frac{P(state|o_i)P(o_i)}{P(state)} - 1|\, P(state)\right) \text{ by Bayes Rule} \\ = \quad & \sum_i |n\, P(state|o_i)\, P(o_i) - P(state)| (\text{as } P(state) \geq 0 \text{ always}) \end{aligned}$$

$$
\begin{aligned}
&= \sum_i |n\, P(state|o_i)\, P(o_i) - \sum_j (P(state|o_j)\, P(o_j))\,| \\
&= \sum_i |\sum_j (P(state|o_i)\, P(o_i)) - \sum_j (P(state|o_j)\, P(o_j))\,| \\
&= \sum_i |\sum_j (P(state|o_i)\, P(o_i) - P(state|o_j)\, P(o_j))\,|
\end{aligned}
$$

Using this result to evaluate the discrimination function

$$
\begin{aligned}
D &= U(detection = 1, (x,y))\, P(detection = 1) \\
&\quad + U(detection = 0, (x,y))\, P(detection = 0) \\
&= \sum_i |\sum_j (P(detection = 1|o_i)\, P(o_i) - P(detection = 1|o_j)\, P(o_j))\,| \\
&\quad + \sum_i |\sum_j (P(detection = 0|o_i)\, P(o_i) - P(detection = 0|o_j)\, P(o_j))\,| \\
&= \sum_i |\sum_j \left(\Pi_{o_i}\, P(o_i) - \Pi_{o_j}\, P(o_j)\right)| \\
&\quad + \sum_i |\sum_j \left((1 - \Pi_{o_i})\, P(o_i) - (1 - \Pi_{o_j})\, P(o_j)\right)| \\
&= \sum_i |\sum_j \left(\Pi_{o_i}\, P(o_i) - \Pi_{o_j}\, P(o_j)\right)| \\
&\quad + \sum_i |\sum_j \left((1 - \Pi_{o_i})\, P(o_i) - (1 - \Pi_{o_j})\, P(o_j)\right)|
\end{aligned}
$$

If $P(o_1) = P(o_2) = ... = P(o_n) = 1/n$ then

$$
\begin{aligned}
D &= \sum_i |\sum_j \left(\Pi_{o_i}\, P(o_i) - \Pi_{o_j}\, P(o_j)\right)| \\
&\quad + \sum_i |\sum_j \left((1 - \Pi_{o_i})\, P(o_i) - (1 - \Pi_{o_j})\, P(o_j)\right)| \\
&= \sum_i |\sum_j \left(\Pi_{o_i}\, 1/n - \Pi_{o_j}\, 1/n\right)| \\
&\quad + \sum_i |\sum_j \left((1 - \Pi_{o_i})\, 1/n - (1 - \Pi_{o_j})\, 1/n\right)| \\
&= 1/n \sum_i |\sum_j \left(\Pi_{o_i} - \Pi_{o_j}\right)| + 1/n \sum_i |\sum_j \left(-\Pi_{o_i} + \Pi_{o_j})\right)| \\
&= 2/n \sum_i |\sum_j \left(\Pi_{o_i} - \Pi_{o_j}\right)|
\end{aligned}
$$

If $n = 2$

$$
\begin{aligned}
D &= \sum_i \left| \sum_j \left(\Pi_{o_i} P(o_i) - \Pi_{o_j} P(o_j) \right) \right| \\
&\quad + \sum_i \left| \sum_j \left((1 - \Pi_{o_i}) P(o_i) - (1 - \Pi_{o_j}) P(o_j) \right) \right| \\
&= |\Pi_{o_1} P(o_1) - \Pi_{o_2} P(o_2)| + |\Pi_{o_2} P(o_2) - \Pi_{o_1} P(o_1)| \\
&\quad + |(1 - \Pi_{o_1}) P(o_1) - (1 - \Pi_{o_2}) P(o_2)| \\
&\quad + |(1 - \Pi_{o_2}) P(o_2) - (1 - \Pi_{o_1}) P(o_1)| \\
&= 2\,|\Pi_{o_1} P(o_1) - \Pi_{o_2} P(o_2)| + 2\,|(1 - \Pi_{o_1}) P(o_1) - (1 - \Pi_{o_2}) P(o_2)|
\end{aligned}
$$

which is of the same form as the same result obtained earlier in Appendix B.1.

B.4 Recognition from an n-object set (Case 2)

A suitable utility function for the task of identifying an object from the set $\{o_1, o_2, ..., o_n\}$ has been proposed as:

$$U(state, (x, y)) = \sum_i |P(o_i|state) - P(o_i)|$$

The discrimination function (assuming only two sensing states) is:

$$
\begin{aligned}
D(x, y) &= U(detection = 1, (x, y))\, P(detection = 1) \\
&\quad + U(detection = 0, (x, y))\, P(detection = 0)
\end{aligned}
$$

Consider:

$$
\begin{aligned}
&U(state, (x, y))\, P(state) \\
&= \left(\sum_i |P(o_i|state) - P(o_i)| \right) P(state)
\end{aligned}
$$

$$
\begin{aligned}
&= \sum_i \left(|P(o_i|state) - P(o_i)|\, P(state)\right) \\
&= \sum_i \left(|\frac{P(state|o_i)\, P(o_i)}{P(state)} - P(o_i)|\, P(state) \right) \\
&= \sum_i \left(P(o_i)\, |P(state|o_i) - P(state)|\right) \\
&\quad \text{(as } P(state) \text{ and } P(o_i) \geq 0 \text{ always)} \\
&= \sum_i \left(P(o_i) |P(state|o_i) - \sum_j (P(o_j) P(state|o_j))\, | \right) \\
&= \sum_i \left(P(o_i)| \sum_j (P(o_j)) P(state|o_i) - \sum_j (P(o_j) P(state|o_j))| \right) \\
&= \sum_i \left(P(o_i)| \sum_j (P(o_j) P(state|o_i)) - \sum_j (P(o_j) P(state|o_j))| \right) \\
&= \sum_i \left(P(o_i)| \sum_j (P(o_j)\, (P(state|o_i) - P(state|o_j)))\, | \right)
\end{aligned}
$$

Using this result to evaluate the discrimination function:

$$
\begin{aligned}
D &= U(detection = 1, (x,y))\, P(detection = 1) \\
&\quad + U(detection = 0, (x,y))\, P(detection = 0) \\
&= \sum_i \left(P(o_i)| \sum_j (P(o_j)\, (P(detection = 1|o_i) - P(detection = 1|o_j)))\, | \right) \\
&\quad + \sum_i \left(P(o_i)| \sum_j (P(o_j)\, (P(detection = 0|o_i) - P(detection = 0|o_j)))\, | \right) \\
&= \sum_i \left(P(o_i)| \sum_j \left(P(o_j) \left(\Pi_{o_i} - \Pi_{o_j} \right) \right) | \right) \\
&\quad + \sum_i \left(P(o_i)| \sum_j \left(P(o_j) \left((1 - \Pi_{o_i}) - (1 - \Pi_{o_j}) \right) \right) | \right) \\
&= \sum_i \left(P(o_i)| \sum_j \left(P(o_j) \left(\Pi_{o_i} - \Pi_{o_j} \right) \right) | \right) \\
&\quad + \sum_i \left(P(o_i)| \sum_j \left(P(o_j) \left(-\Pi_{o_i} + \Pi_{o_j} \right) \right) | \right) \\
&= \sum_i \left(P(o_i)| \sum_j \left(P(o_j) \left(\Pi_{o_i} - \Pi_{o_j} \right) \right) | \right)
\end{aligned}
$$

$$+\sum_i \left(P(o_i) | \sum_j \left(P(o_j) \left(\Pi_{o_i} - \Pi_{o_j} \right) \right) | \right)$$

$$= 2 \sum_i \left(P(o_i) | \sum_j \left(P(o_j) \left(\Pi_{o_i} - \Pi_{o_j} \right) \right) | \right)$$

This function can be readily evaluated, and consists of a sum of the difference of probabilistic membership functions.

If $n = 2$

$$\begin{aligned} D &= 2 \sum_i \left(P(o_i) | \sum_j \left(P(o_j) \left(\Pi_{o_i} - \Pi_{o_j} \right) \right) | \right) \\ &= 2\, P(o_1) | P(o_2)(\Pi_{o_1} - \Pi_{o_2}) | + 2\, P(o_2) | P(o_1)(\Pi_{o_2} - \Pi_{o_1}) | \\ &= 2\, P(o_1) P(o_2) | \Pi_{o_1} - \Pi_{o_2} | + 2\, P(o_1) P(o_2) | \Pi_{o_2} - \Pi_{o_1} | \\ &\quad \text{(as } P(o_1) \text{ and } P(o_2) \geq 0 \text{ always)} \\ &= 4\, P(o_1) P(o_2) | \Pi_{o_1} - \Pi_{o_2} | \end{aligned}$$

which is the same result as was obtained earlier in Appendix B.2.

B.5 Recognition on the basis of multiple states

A suitable utility function for the task of identifying an object from the set $\{o_1, o_2, ..., o_n\}$ has been proposed as:

$$U(state, (x, y)) = \sum_i |P(o_i | state) - P(o_i)|$$

Consider the situation where there are more than two states which may occur at a given location in the sensing space. In this case the discrimination function is given by:

$$D(x, y) = \sum_k (U(state_k, (x, y))\, P(state_k))$$

It was shown in Section B.4 that

$$\begin{aligned} & U(state_k,(x,y))\,P(state_k) \\ = & \sum_i \left(P(o_i) | \sum_j (P(o_j)\,(P(state_k|o_i) - P(state_k|o_j))) \,| \right) \end{aligned}$$

Using this result to evaluate the discrimination function:

$$\begin{aligned} D &= \sum_k (U(state_k,(x,y))\,P(state_k)) \\ &= \sum_k \sum_i \left(P(o_i) | \sum_j (P(o_j)\,(P(state_k|o_i) - P(state_k|o_j))) \,| \right) \end{aligned} \quad \text{(B.2)}$$

Consider the two object case described in Section B.2. Expanding Equation (B.2) about i for $i = \{1, 2\}$ gives:

$$\begin{aligned} D &= \sum_k \left(\sum_i \left(P(o_i) | \sum_j (P(o_j)\,(P(state_k|o_i) - P(state_k|o_j))) \,| \right) \right) \\ &= \sum_k (P(o_1) | P(o_2)\,(P(state_k|o_1) - P(state_k|o_2)) \,| \\ &\quad + P(o_2) | P(o_1)\,(P(state_k|o_2) - P(state_k|o_1)) \,|) \\ &= \sum_k (P(o_1)\,P(o_2) | P(state_k|o_1) - P(state_k|o_2)| \\ &\quad + P(o_1)\,P(o_2) | P(state_k|o_2) - P(state_k|o_1)|) \\ &= \sum_k (P(o_1)\,P(o_2) | P(state_k|o_1) - P(state_k|o_2)| \\ &\quad + P(o_1)\,P(o_2) | P(state_k|o_1) - P(state_k|o_2)|) \\ &= \sum_k (2\,P(o_1)\,P(o_2) | P(state_k|o_1) - P(state_k|o_2)|) \\ &= 2\,P(o_1)\,P(o_2) \sum_k | P(state_k|o_1) - P(state_k|o_2)| \end{aligned}$$

In this case $P(state_k|o_i)$ is the feature PMF for object o_i for the feature represented by $state_k$. The form of this equation is simply the same as that represented in Equation (B.1) summed over k states (as opposed to the two states addressed by Equation (B.1)).

Appendix C

Probabilistic membership values for path sensing

Consider a shape whose location in space is uncertain in one dimension. The direction and magnitude of the uncertainty are assumed to be known, and can be described by a range in the x and y components as $(\Delta x, \Delta y)$. Hence the range of possible locations of each vertex are known, and the shape can be represented as shown in Figure C.1.

Consider the edge designated as *edge* 1 in the diagram. The uncertainty in the location of the endpoints of this edge are such that one must lie between (x_1, y_1) and $(x_1 + \Delta x, y_1 + \Delta y)$ and the other must lie between (x_2, y_2) and $(x_2 + \Delta x, y_2 + \Delta y)$. The slope of the edge is fixed, however, so that fixing one end-point determines the other. To determine the probabilistic membership function for this edge in polar coordinates, it is necessary to know what is the likelihood of contacting this side, with a directed ray described by the coordinates (r, θ).

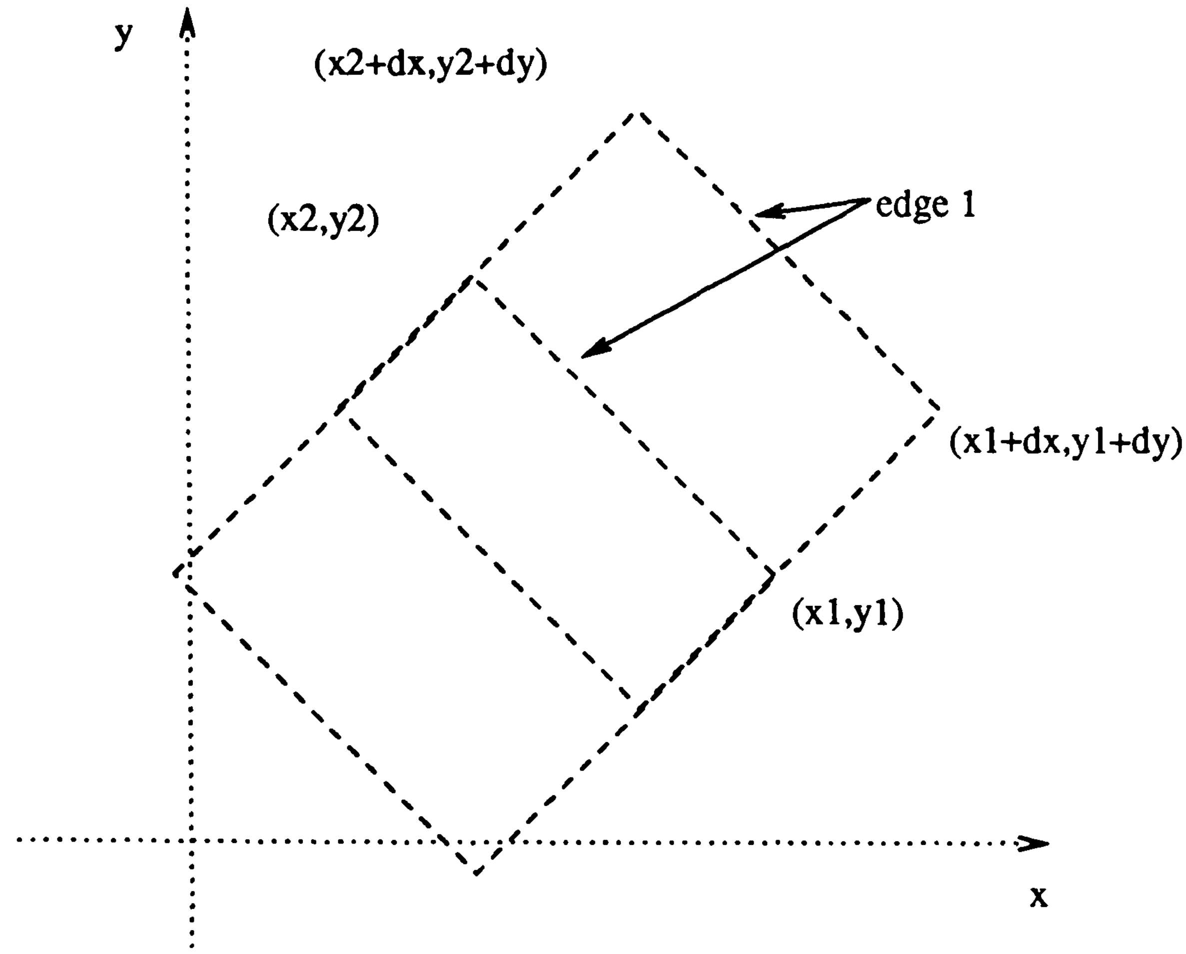

Figure C.1: Uncertainty map for the shape.

One approach is to consider the point of intersection of the ray with the uncertainty ranges for each end-point as shown in Figure C.2.

If, for example, the ray contacts with the line connecting (x_1, y_1) and $(x_1 + \Delta x, y_1 + \Delta y)$ to the right of $(x_1 + \Delta x, y_1 + \Delta y)$ and similarly for the other endpoint, then there will be no possibility of the ray contacting the edge, irrespective of the location of the edge within its uncertainty range. This situation is represented by *ray* 1 in the diagram. The same result will hold if the two endpoint uncertainty lines are contacted to the left of (x_1, y_1) and (x_2, y_2) respectively. (See *ray* 2.)

Consider the situation depicted by *ray* 3 and *ray* 4, where the ray intersects with one line to the right of its admissible uncertainty range, and to the left of the admissible range for the other. In this case contact with *edge* 1 will be guaranteed provided that the ray is incident on *edge* 1 from the exterior of the object. Thus *ray* 3 will contact *edge* 1, whereas *ray* 4 will not. (As *ray* 3 and

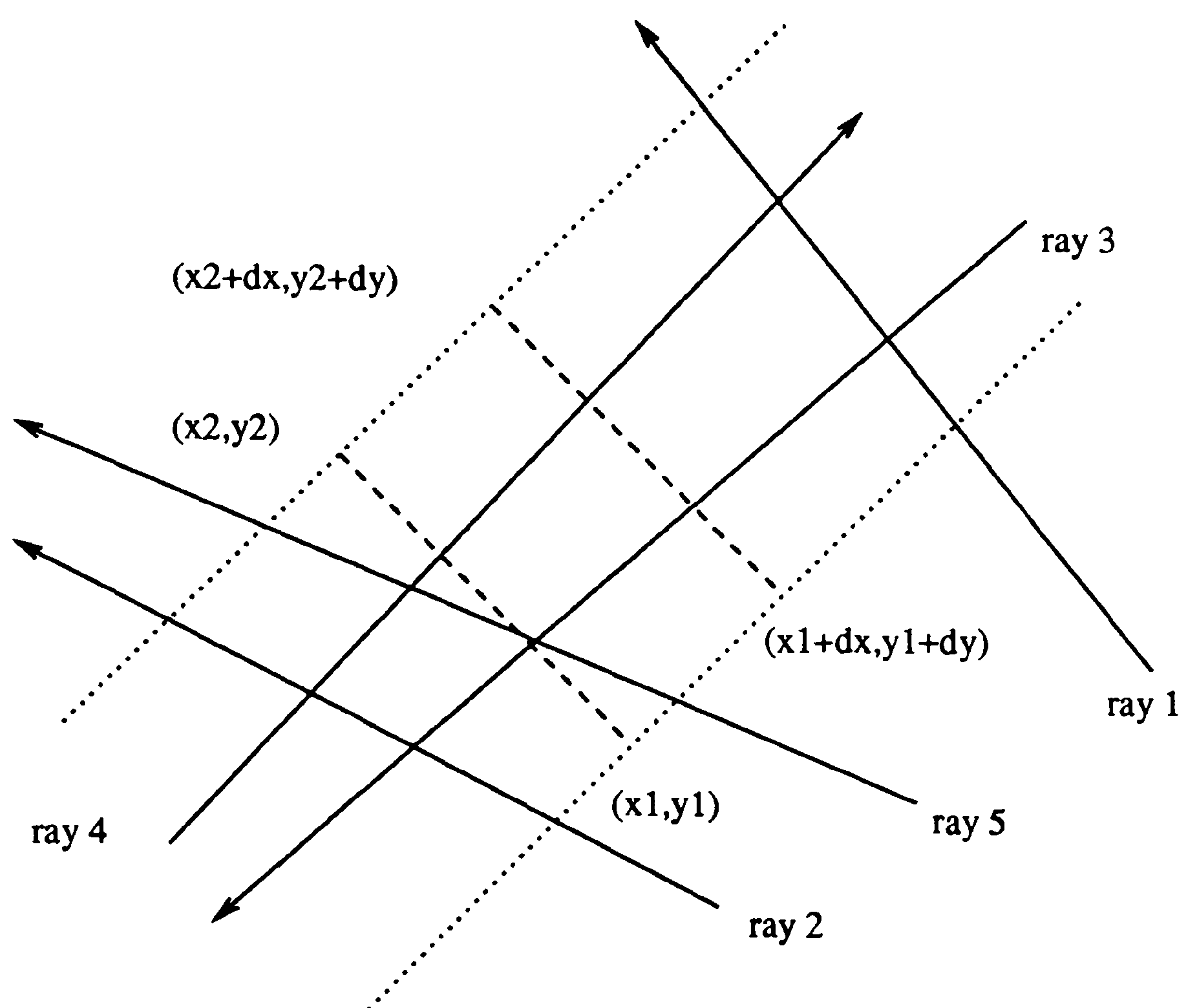

Figure C.2: Sensing rays within the workspace.

ray 4 represent the same line in space, this example underlines the importance of carefully defining the coordinate system (r, θ) used to describe the rays, so that there is no ambiguity in direction., as discussed in Section 4.6.)

If contact is made with one line within its admissible range, contact may result between the ray and the edge for some subset of the range of possible locations of the shape. This will result in a probabilistic membership value between zero and one. The particular value will indicate the proportion of the uncertainty range which would give rise to contact. Consider *ray* 5, which intersects the range of possible values for the first vertex one third of the way from (x_1, y_1) to $(x_1 + \Delta x, y_1 + \Delta y)$, and passes to the left of (x_2, y_2). It is clear that if the *edge* 1 lies between $(x_1 + \Delta x, y_1 + \Delta y)$ and $(x_2 + \Delta x, y_2 + \Delta y)$ there will be no contact between the ray and the edge. The same will be true of two thirds of the possible locations of *edge* 1. If however the edge is located between (x_1, y_1) and (x_2, y_2), or immediately to the right of this location, contact will result. Thus the probability of contact (or value of the PMF) for this ray will be 1/3.

The method used to determine the values of the probabilistic membership function, were to find a normalized, parametrized form of the uncertainty lines for each vertex in a single variable, say t, with $t = 0$ corresponding to one limit of the admissible range and $t = 1$ to the other. The values of t corresponding to the points of intersection of the ray defined by (r, θ) with both lines were determined, and values of t below zero or above one were set to zero and one respectively. The expectation of detection (and hence the value for the PMF at the point (r, θ) in sensing space) then corresponded to the difference between the two values of t, taking care to observe the direction requirement mentioned earlier.

This algorithm is readily implemented, and provides a reliable measure of the relevant probabilities sought.

Bibliography

[1] Peter Kerby Allen. *Object recognition using vision and touch.* Kluwer Academic Publishers, Boston, 1988.

[2] Haruo Asada and Michael Brady. The curvature primal sketch. *IEEE Trans Pattern Anal Mach Intell*, 8(1):2–14, January 1986.

[3] Ruzena Bajcsy. Shape from touch. In *Advances in Automation and Robotics, Vol 1*, pages 209–258, JAI Press, 1985.

[4] Ruzena Bajcsy. What can we learn form one finger experiments? In *The First International symposium on Robotics Research*, page , The MIT Press, Cambridge, Massachusetts, 1984.

[5] G. Beni, S. Hackwood, L.A. Hornak, and J.L. Jackel. Dynamic sensing for robots: an analysis and implementation. *Int Jour Robotics Research*, 2(2):51–61, 1983.

[6] James O. Berger. *Statistical Decision Theory.* Springer-Verlag, 1980.

[7] David Blackwell and M.A. Girshick. *Theory of Games and Statistical Decisions.* Dover Publications, 1954.

[8] Robert C. Bolles and Ronald A. Cain. Recognizing and locating partially visible objects: the local-feature-focus method. *Int Jour Robotics Research*, 1(3):57–82, 1982.

[9] Michael Brady. Artificial intelligence and robotics. In M. Brady, L.A. Gerhardt, and H.F. Davidson, editors, *Robotics and Artificial Intelligence*, pages 47–63, Springer-Verlag, 1984.

[10] Michael Brady. Seeds of perception. In *Third Alvey Vision Conference*, pages 259–265, Cambridge, England, 1987.

[11] Roger A. Browse. Feature-based tactile object recognition. *IEEE Trans Pattern Anal Mach Intell*, 9(6):779–786, November 1987.

[12] Alec Cameron. Optimal tactile sensor placement. In *IEEE Int Conf Robotics and Automation*, pages 308–313, Scottsdale, Arizona, May 1989.

[13] Alec Cameron, Ron Daniel, and Hugh Durrant-Whyte. Tactile geometry for images and normals. In Alícia Casals, editor, *Sensor Devices and Systems for Robots*, pages 67–77, Springer-Verlag, 1989.

[14] Alec Cameron, Ron Daniel, and Hugh Durrant-Whyte. Touch and motion. In *1988 IEEE Int Conf Robotics and Automation*, pages 1062–1067, Philadelphia, Pennsylvania, April 1988.

[15] M.H. DeGroot. *Optimal Statistical Decisions*. McGraw-Hill, 1970.

[16] Hugh F. Durrant-Whyte. *Integration, coordination and control of multi-sensor robot systems*. Kluwer Academic Publishers, Boston, 1985.

[17] R.E. Ellis. Acquiring tactile data for the recognition of planar objects. In *IEEE Int Conf Robotics and Automation*, pages 1799–1805, Raleigh, North Carolina, April 1987.

[18] I.D. Faux and M.J. Pratt. *Computational Geometry for Design and Manufacture.* Ellis Horwood, 1979.

[19] K.S. Fu, R.C. Gonzalez, and C.S.G. Lee. *Robotics: Control, Sensing, Vision, and Intelligence.* McGraw-Hill, 1987.

[20] P.C. Gaston and T. Lozano-Pérez. Tactile recognition and localization using object models: the case of a polyhedra on a plane. *IEEE Trans Pattern Anal Mach Intell*, 6(3):257–266, May 1984.

[21] G. Gordon, editor. *Active Touch.* Pergammon Press, 1978.

[22] W. Eric L. Grimson. Disambiguating sensory interpretations using minimal sets of sensory data. In *IEEE Int Conf on Robotics and Automation*, pages 286–292, San Fransisco, California, April 1986.

[23] W. Eric L. Grimson. Sensing strategies for disambiguating among multiple objects in known poses. *IEEE Jour Robotics and Automation*, 2(4):196–213, December 1986.

[24] W. Eric L. Grimson and Tomás Lozano-Pérez. Search and sensing strategies for recognition and localization of two and three dimensional objects. In *Third ISRR*, pages 81–88, .

[25] W.E.L. Grimson and T. Lozano-Pérez. Localizing overlapping parts by searching the interpretation tree. *IEEE Trans Pattern Anal Mach Intell*, 9(4):469–482, July 1987.

[26] W.E.L. Grimson and T. Lozano-Pérez. Model-based recognition and localization from sparse range or tactile data. *Int Jour Robotics Research*, 3(3):3–35, Fall 1984.

[27] Gregory D. Hager. *Active Reduction of Uncertainty in Multi-Sensor Systems.* PhD thesis, University of Pennsylvania, 1988.

[28] Leon D. Harmon. Automated tactile sensing. *Int Jour Robotics Research*, 1(2):3–32, Summer 1982.

[29] Leon D. Harmon. Robotic taction for industrial assembly. *Int Jour Robotics Research*, 3(1):72–76, Spring 1984.

[30] Leon D. Harmon. Tactile sensing for robots. In M. Brady et al, editor, *Robotics and Artificial Intelligence*, pages 109–157, Springer-Verlag, 1984.

[31] Leon D. Harmon. *Touch-Sensing Technology : A Review.* Tech Report MSR80-03, SME, 1980.

[32] W. Daniel Hillis. *Active Touch Sensing.* AI Memo 629, MIT, April 1981.

[33] W. Daniel Hillis. A high-resolution imaging touch sensor. *Int Jour Robotics Research*, 1(2):33–44, Summer 1982.

[34] John M. Hollerbach. *Computers, brains, and the control of movement.* AI Memo 686, MIT, June 1982.

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

[36] A.J. Jones. *Game Theory: Mathematical Models of Conflict.* Ellis Horwood, 1980.

[37] Roberta L. Klatzky, Ruzena Bajcsy, and Susan J. Lederman. Object exploration in one and two fingered robots. In *IEEE Int Conf Robotics and Automation*, pages 1806–1809, Raleigh, North Carolina, April 1987.

[38] Raymond McKendall and Max Mintz. *Models of Sensor Noise and Optimal Algorithms for Estimation and Quantization in Vision Systems.* Technical Report, GRASP Laboratory, University of Pennsylvania, February 1987.

[39] T. Pavlidis. *Algorithms for Graphics and Image Processing.* Springer-Verlag, 1982.

[40] Elaine Rich. *Artificial Intelligence.* McGraw-Hill, 1983.

[41] John L. Schneiter. An objective tactile sensing strategy for object recognition and localization. In *IEEE Int Conf Robotics and Automation*, pages 1262–1267, San Fransisco, California, April 1986.

[42] John N. Schneiter and Thomas B. Sheridan. An automated tactile sensing strategy for planar object recognition and localization. To be published in *IEEE Trans Pattern Anal Mach Intell.*

[43] S.A. Stansfield. Primitives, features, and exploratory procedures: building a robot tactile perceptual system. In *IEEE Int Conf Robotics and Automation*, pages 1274–1279, San Fransisco, California, April 1986.

[44] S.A. Stansfield. Visually-aided tactile exploration. In *IEEE Int Conf Robotics and Automation*, pages 1487–1492, Raleigh, North Carolina, April 1987.

[45] Sharon Ann Stansfield. *Visually-Guided Haptic Object Recognition.* PhD thesis, University of Pennsylvania, October 1987.

[46] Fridtjof Stein. *Recognition of overlapped objects.* Robotics Group Report OU-RRG-87-11, University of Oxford, 1987.

[47] J.B. Sykes, editor. *The Concise Oxford Dictionary.* Oxford University Press, 1984.

[48] Jerry L. Turney, Trevor N. Mudge, and Richard A. Volz. Recognizing partially occluded parts. *IEEE Trans Pattern Anal Mach Intell*, 7(4):410–421, July 1985.

[49] Abraham Wald. *Sequential Analysis.* Dover Publications, 1947.

[50] Abraham Wald. *Statistical Decision Functions.* John Wiley and Sons, 1950.

[51] Patrick Winston. *Artificial Intelligence.* Addison Wesley, 1977.