Perceptual Grouping of 3D Sensory Data Using Bayesian Attributed Hypergraphs

by

Ramiro Liscano

A thesis
presented to the University of Waterloo
in fulfilment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Systems Design Engineering

Waterloo, Ontario, Canada, 1998

0-612-30621-6

The University of Waterloo requires the signatures of all persons using or photocopying this thesis. Please sign below. and give address and date.

Abstract

This thesis investigates the use of Bayesian networks for the perceptual grouping of features extracted from images of intensity and range data. A formalism for the development of Bayesian networks for perceptual grouping is presented that is based on a decomposition by parts methodology for the structure of the networks. and compatibility functions for the computation of the conditional probabilities. There exists a very strong relationship between the formation of a causal network and the process used to decompose an object into its components. A set of guidelines are presented for the design of a Bayesian network for perceptual grouping.

Compatibility functions are a measure in the quality of fit between a set of features to a model that represents a grouping among those features. This model is represented as a node in a Bayesian network and compatibility functions are used to compute conditional probabilities in the formation of the grouping based on attributes and relations among the features. Five different compatibility functions are presented as examples. These include an edge. surface proximity. planar surface. coplanar surface. and parallel surface compatibility functions. The edge and surface proximity compatibility functions are unique in that they rely on computatons both in a 2-D image plane as well as 3-D space. These algorithms determine a polygonal approximation of a planar surface as well as a common virtual surface between two polygons in 3-D space.

A unified representation of Bayesian networks and attributed hypergraphs. a Bayesian attributed hypergraph (BAHG), is developed that allows for the instantiation of multiple Bayesian networks. BAHGs are a subset to attributed hypergraphs allowing for the specification and maintenance of multiple Bayesian networks without creating multiple instantiations of nodes that represent the same event. This facilitates the grouping process among multiple surfaces since continuity is maintained among groupings that share common feature sets. Also, a graphical description of the grouping network can be used to guide the creation of the BAHG. This network can be recursively applied to the features extracted from the sensory data. It is this recursive nature of perceptual grouping that makes the BAHG suitable for computer vision and image understanding problems.

Validation of the use of a BAHG is presented using a BAHG network for the detection of corners and continuity among adjacent surfaces, applied to several images of 3-D data points extracted from a portable range sensor.

Acknowledgements

I would like to acknowledge the support of my supervisors Dr. Andrew K. C. Wong and Dr. Shadia Elgazzar. They both believed in my ability to complete this thesis and supported me throughout the whole process. I would also like to acknowledge Dr. Suyahhah Abu-Hakima for allowing me to continue working in my thesis while under her supervision at the National Research Council (NRC). I am also gratefull for the leave of absence from the Institute for Information Technology at NRC for education during 1992 to 1994. I especially would like to thank my wife Kimberly, my children Alexandria, Monica, and Anthony who had to endure with me as I struggled with my research. During this period of time my parents, Ramiro and Rita Liscano, and wife's parents, Tom and Mona Hardwick, had little doubt I could accomplish this, they just wanted it over with as soon as possible.

Contents

List of Tables

List of Figures

Chapter 1

Introduction and Motivation

1.1 Automated 3-D Environment Modeling

Research in the domain of automated 3-D modeling has primarily focussed on the extraction of 3-D surfaces and volumetric primitives from 3-D sensory data of machined parts. These types of objects can easily be carried and placed in a controlled environment and scanned using high resolution active sensors. This is significantly different from the modeling of large indoor environments where it is necessary to bring the sensor to the environment. changing the characteristics of the sensed data dramatically. In large indoor environments it is difficult to manipulate the objects and the sensor to achieve ideal sensory data. The result of this is that surfaces extracted from 3-D data of these environments tend to be sparse and fragmented. Occlusion among objects in the scene is common also since little control is available of the environment. Objects occluded from the sensor and missing data make it necessary to use knowledge about the environment combined with evidence from sensory data to hypothesize the existence of particular surface groupings. In the modeling of large environments, as opposed to that of machined parts. the requirement for detail diminishes and the challenge becomes one of trying to extract from the sensory data a model containing less detail but still maintaining enough information to be able to reconstruct the overall shape and size of the environment. This is not as simple as relaxing the tolerances used for computing the surface models from the sensory data. but requires an ability to determine the type of information that is relevant to the model being created.

The inability to group the features extracted from the sensory data has resulted in the lack of modeling systems that can cope with cluttered environments and sparse data appropriately. Indoor environments. typically office and home structures. are man-made environments that contain enough structure to be able to detect the overall shape of a room. but are nearly always cluttered with items that make it difficult to hypothesize the overall room's structure. It is necessary then to develop algorithms that can hypothesize the existence of particular formations among the surfaces extracted from the data that go beyond what can actually be detected by the sensor. It is also important to maintain with these hypothesized formations a belief value of the existence of the formations. The use of belief values is important from a user's perspective in that many automated algorithms hide from the user the decision algorithms and thresholds used for making those decisions. When these are available then some confidence in these results are passed on to the user.

In the past several years advances in 3-D sensors. computer vision. and mobile robots have made it possible to have a programmable and versatile mobile sensing unit capable of being used to register the 3-D features of indoor scenes so that a computer model of the environment can be reconstructed. The procedure is simple but the actual implementation of such a system, with current technology. is not easy. The underlying reasons for this are the inability of the system to cope with sensor motion uncertainties. inconsistent sensory data resolution over several acquisitions and incomplete data. This has a direct impact on the ability to group features extracted from the sensory data into useful computer models.

Such a programmable robot. shown in figure 1.1. is equipped with several sensors to detect objects in an indoor environment and is able to estimate the robot's position. This particular robot has been fitted with ultrasound, laser. CCD camera. and odometer sensors.

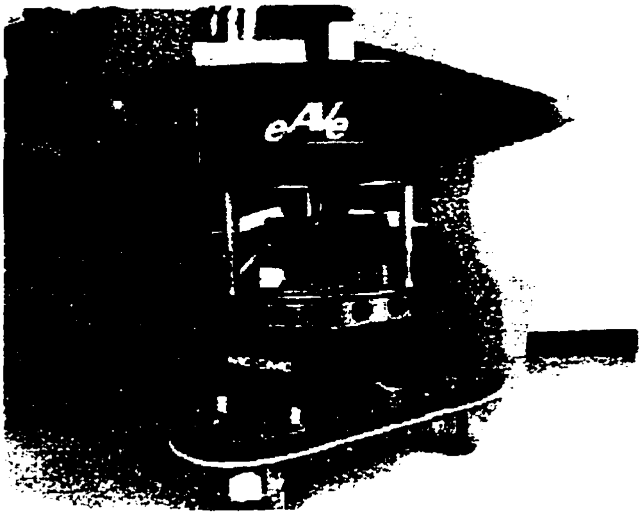


Figure 1.1: Programmable mobile robot equipped for environment modeling.

This system has been used to autonomously construct 3-D computer models of unknown environments by programming the robot to traverse about the area to be modeled and using the sensors on board the robot to register the position of objects in the environment relative to the robot's location [51, 16]. Over a series of sensor frames the data can then be synthesized to create a 3-D model of the environment. An example of this is depicted in figure 1.2 for 5 scans placed vertically one on top of the other for a stationary location of the robot.

Figure 1.2 (a) depicts the 5 intensity images stacked one above the other while figure 1.2 (b) is an orthogonal projection of the surfaces extracted from the range component

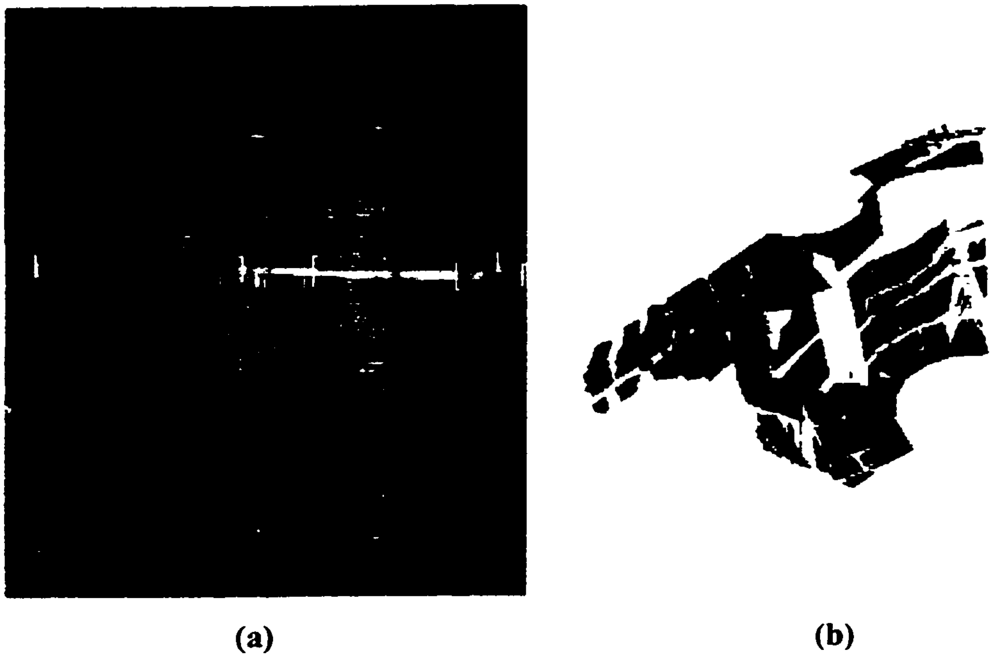


Figure 1.2: Multiple range and image scans of a mobile robot laboratory.

of the data. One can see from this data that fragmentation and occlusion are common.

This thesis focuses on the extraction and the perceptual grouping of feature sets from indoor sensory data, of the type shown in figure 1.2, for the purpose of hypothesizing the existence of formations among those features. In particular, the example used for the thesis detects corners and continuity among planar surfaces. The proposed approach maintains and distributes belief values in the hypothesized formations using a Bayesian network formalism. An approach is presented for computing the conditional probabilities required for Bayesian formalisms based on compatibility functions that measure how well a set of features match a proposed geometric model. As well a unified representation of Bayesian networks and attributed hypergraphs is presented that is termed a Bayesian Attributed Hypergraph (BAHG). This unified representation is useful in maintaining multiple Bayesian network instantiations without multiple feature set nodes. This facilitates the construction of models from the data.

1.2 Perceptual Grouping for 3-D Geometric Modeling

The term perceptual grouping in the computer vision domain refers to the process of clustering features for the purpose of extracting a compound feature that exhibits more structure and form than the original feature sets did. The term "image understanding" is the process of associating symbolic labels to features or grouped sets of features. It is synonymous to the computer vision domain of "object recognition" where particular objects are detected within a set of sensory data points. Early research in perceptual grouping [66. 67. 65. 74] by cognitive psychologists had focused primarily on the human perception of basic patterns from images without understanding the image itself. Later. researchers in the field of computer vision integrated perceptual grouping processes with object recognition such that it is difficult to separate the two processes.

The segmentation procedures used for the extraction of surfaces for 3-D geometric modeling are also part of the image understanding process but stop at the recognition of fundamental generic shapes. like patches. triangles. and planes. instead of a complete structure that can be composed of these surfaces. These operations can be considered a form of perceptual grouping since the end result is the clustering of 3-D points to surfaces that contain more structure and form than the original points.

Geometric modeling of 3-D sensory data is the process of joining and interpolating among several 3-D feature sets for the formation of structures that contain a combination of these features sets. Geometric modeling differs from image understanding in that for geometric modeling there does not exist any detailed prior geometric models that guide the grouping process but instead generic models are used. For image understanding the models tend to contain more detail that are specific to the object to be recognized. Both geometric modeling and image understanding though share similar underlying functionalities. They both rely heavily on the perception of continuity, proximity, and symmetry which are considered as fundamental perceptual grouping processes. Thus geometric modeling relies on the recognition and grouping of fundamental 3-D shapes. A significant amount of research has gone into the extraction of surfaces from 3-D sensory data, in particular for the modeling of small or manufactured parts [8, 63, 17]. This is not an easy problem. It is also hampered by the ability to gather consistent 3-D data and the difficulty in the grouping of this data into meaningful structures.

Recent attempts in the modeling of larger indoor or outdoor environments [60, 70, 39, 28] from 3-D sensory data have demonstrated it to be a difficult task, due mainly to the amount of missing, sparse, and obscured data. In most circumstances a set of heuristics have had to be declared to decide what and how the surfaces should be joined. Unfortunately these heuristics tend to be embedded into the algorithms and no formal approaches are used for estimating the certainty of the grouping procedures. This leads to systems that are not easily extendable and with no formalism to measure the confidence in the grouping.

1.3 Uncertainty Management for Perceptual Grouping

Uncertainty in the quality of the sensory data and missing information have been the main stumbling blocks in the perceptual grouping of sensory data. This uncertainty is generally measured as a deviation of the sensory data from a geometric model using a compatibility function. In most perceptual grouping examples thresholds are used at each stage in the grouping process to perform a fail or pass decision to check if the structure fits the model. This introduces a system where it is possible to stop the grouping process at an early stage of the processing if the features are not compatible with the model. It does not allow for other evidence to add support to the grouping. On the other hand. uncertainty propagation techniques define procedures for the propagation of the uncertainty among several hypothesized outcomes. This has the following advantages:

- The decision to group a set of features can be performed after all the evidence has been accumulated allowing for the influence of other information to affect the decision.
- In principal. it is simple to consider the effect of new evidence by introducing a new compatibility function into the system. In practice the approach used for uncertainty propagation will determine how the new evidence is introduced.

Introducing an uncertainty management approach into the perceptual grouping process will create a more complex system since it is necessary to combine both the perceptual grouping and the uncertainty propagation knowledge. For this reason it is important to separate the procedural and declarative knowledge to facilitate the management and updating of the system. In this particular case procedural knowledge is used for the perceptual grouping algorithms and for computing the certainty values in the groupings. Declarative knowledge maintains information about the features extracted from the sensory data and the hypothesized groupings. In many circumstances the declarative knowledge is represented as structures in a procedural type of language and embedded into a domain specific application. It is difficult to extract the representation used for the declarative knowledge from the algorithms and it is also difficult to specify any new procedural knowledge without building a new system. This becomes apparent when applying Bayesian network techniques to the perceptual grouping of 3-D features where multiple networks have to be instantiated. Previously, no unified representation for the storage of both the perceptual grouping formations and the maintenance of the Bayesian networks has been reported.

This is where the BAHG offers some advantages over simply using Bayesian networks. A BAHG maintains the declarative knowledge and belief values in the formation of the groupings specified by the nodes in the BAHG network. A BAHG also has all the required attributes and relations among the groupings that can be used to update the belief values using Bayesian statistics. A BAHG has added functionality over a Bayesian network in that it can maintain several instances of perceptual groupings among surfaces extracted from the sensory data. It is also a repository for any declarative knowledge accumulated by the perceptual grouping process. This consists of surface attributes. relations among surfaces. causal relations among the groupings. any conditional probabilities required to compute the belief in the hypothesized surface to surface grouping. Procedural knowledge for the formation and computation of the conditional probabilities is still required and is embedded in the construction routines and compatibility functions. These though are declared explicitly in a BAHG network. BAHGs also allow the use of standard hypergraph matching procedures in the grouping process. This allows the specification of multiple surface groupings. a difficult process when using only Bayesian network specification languages because one has to explicitly define all the nodes and conditional probabilities of the network.

1.4 Problem Statement and Research Goals

The ultimate goal of this research is to formulate an approach for the perceptual grouping of 3-D sensory data along with a mechanism for maintaining belief values in the formation of that grouping. Of particular interest to this thesis has been the grouping of planar surfaces extracted from 3-D data of large indoor environments. This type of data is characteristic of spurious data that can be fairly fragmented when segmented and can benefit from the integration of evidence from several sources of information.

To achieve this aim the following subgoals were achieved:

- An uncertainty management technique was developed for maintaining belief values in the perceptual grouping of 3-D sensory data. This approach is based on formalisms developed for Bayesian networks.

- Random variables that represent perceptual groupings of the 3-D sensory data were defined. These groupings have a direct correlation with the nodes of a defined Bayesian network.

- A formalism for the construction of a Bayesian network to represent the causal relations among several 3-D perceptual groupings was developed.

- A formalism for computing conditional probabilities based on compatibility functions that measure how well a set of features match a proposed geometric model was developed.

- A representation that unifies the maintenance of declarative knowledge and uncertainty based on the integration of attributed hypergraphs and Bayesian networks was developed. This representation is defined as a Bayesian Attributed Hypergraph (BAHG).

- The BAHG can be specified graphically as a network having the added value that the perceptual grouping process becomes available to the user as a network and not solely as embedded algorithms.

- The perceptual grouping system was tested using real 3-D sensory data of indoor environments.

1.5 Proposed Approach

The system commences the perceptual grouping operation from a set of features extracted from 3-D range images [25]. Figure 1.3 illustrates the processes that are executed in the goal of creating perceptual groupings from the 3-D sensory data. These perceptual groupings are maintained in a Bayesian Attributed Hypergraph.

The processes in figure 1.3 are numbered to signify the order in which they are executed and the arrows signify the flow of data. The following processes have been developed: The *Sensor Data Acquisition* module used to acquire actual 3-D sensory data and create an image of 3-D points with their corresponding intensities [25]. The *Planar Surface Extractor* is a module developed by Boulanger [14] and used in this system to extract the planar surface parametric equations and covariance matrix. An image of the labeled planar surfaces is also generated. The *Polygon Extractor* is an edge extractor algorithm developed to estimate a polygonal representation using the trim of a planar surface [50]. This knowledge is represented as a hypergraph. The *BAHG Constructor* constructs the BAHG given a BAHG network as an input parameter. The compatibility functions are executed to compute the required conditional probabilities. The output of this process is a BAHG that can be used by a *Environment Modeling* process to create geometric models from the sensory data. The *Environment Modeling* module is an optional process that has not been implemented but is illustrated here as an example of a process that uses the BAHG to finally create a model of the environment. The application domain is indoor environment modeling with a focus on the mechanism for the management of uncertainty in the sensory data that would lead to the calculation of belief values in the formation of geometric formations from the sensory data. The *Environment Modeling* process uses knowledge in the BAHG and the polygons to construct models of the environment. The belief values maintained in the BAHG are used to determine if

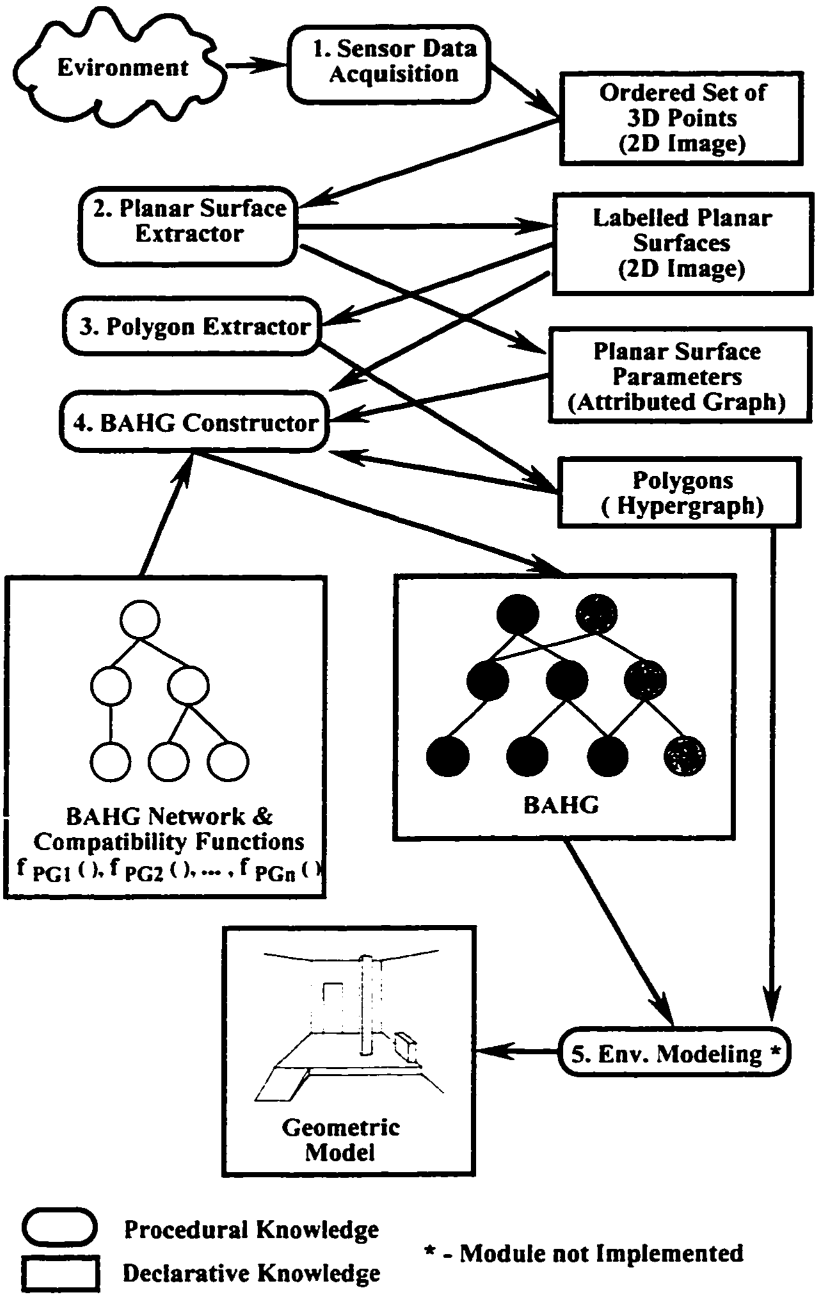

Evironment
1. Sensor Data Acquisition
Ordered Set of 3D Points (2D Image)
2. Planar Surface Extractor
Labelled Planar Surfaces (2D Image)
3. Polygon Extractor
Planar Surface Parameters (Attributed Graph)
4. BAHG Constructor
Polygons (Hypergraph)
BAHG Network & Compatibility Functions
f PG1 ( ), f PG2 ( ), ... , f PGn ( )
BAHG
Geometric Model
5. Env. Modeling *
Procedural Knowledge
Declarative Knowledge
* - Module not Implemented

Figure 1.3: Overall system architecture.

surfaces should be joined. Some examples are presented of how this would be performed in chapter 5 with actual experimental cases demonstrated in chapter 6.

1.6 Thesis Structure

This thesis consists of 6 chapters including the introduction. Chapter 2 is an overview of perceptual grouping approaches in the domain of computer vision. This is organized into two major topics: Section 2.1 presents several approaches for computing compatibility measures among feature sets in an image with respect to a reference geometric model. This can be represented as declarative knowledge computed from the feature sets that is used for computing a belief value in the formation of a grouping. Section 2.2 reviews existing methodologies for perceptual grouping. Some of these approaches integrate uncertainty management procedures into the perceptual grouping process.

Chapter 3 investigates and proposes an approach for the use of Bayesian networks for the perceptual grouping of surfaces extracted from 3-D points. The chapter aims to present an approach for the design of a Bayesian network for perceptual grouping and defines some 3-D compatibility functions used to compute a confidence measure between a set of 3-D features and a hypothesized model of the grouped surfaces. These confidence measures are used to compute the conditional probabilities required in the Bayesian network. Both a general theory is developed and a particular example is given for the detection of corners and continuity among planar surfaces.

Chapter 5 develops the formalism of the Bayesian Attributed Hypergraph as a special instance of an attributed hypergraph. This also includes a definition of a graphical interface for creating a BAHG that is similar to those used for Bayesian networks. This interface allows for the specification of a particular BAHG network which is used by a BAHG Constructor process to populate and construct a fully instantiated BAHG. The

BAHG has added advantages over a Bayesian network (BN) in that it maintains a unified representation for multiple BNs and therefore algorithms that are required to search across several surface groupings can perform this operation efficiently without having to match across separate networks.

Chapter 6 presents an overview of an experimental system used to acquire 3-D range data that was used to test the proposed compatibility functions and Bayesian network for the detection of corners and continuity among planar surfaces. The results demonstrate that the approach does form consistent groupings that are sensitive to the confidence measures computed by the compatibility functions. The results of the algorithm are hypothesized perceptual groupings and their associated belief values. No attempt was made to decide what would be appropriate belief values so as to finally join the surfaces that would make good corners or continuous surfaces.

Finally chapter 7 summarizes the thesis and makes recommendations for further research and expansion of this work. It outlines the positive aspects of perceptual grouping using BAHGs and the limitations encountered with this particular approach.

Figure 1.4 is a diagram that summarizes the contents of the thesis in a visual manner. In the left column are the chapters that discuss the topics outlined in the boxes of the right column.

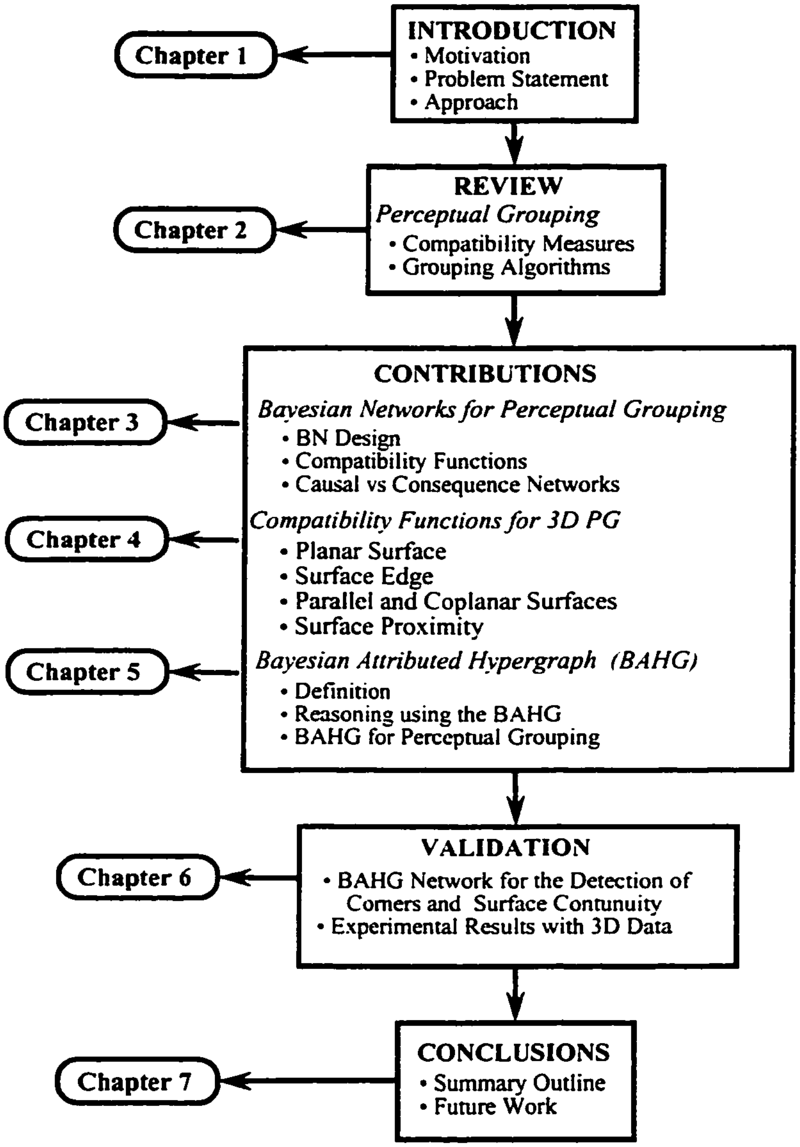

Chapter 1
INTRODUCTION
• Motivation
• Problem Statement
• Approach
Chapter 2
REVIEW
Perceptual Grouping
• Compatibility Measures
• Grouping Algorithms
CONTRIBUTIONS
Chapter 3
Bayesian Networks for Perceptual Grouping
• BN Design
• Compatibility Functions
• Causal vs Consequence Networks
Chapter 4
Compatibility Functions for 3D PG
• Planar Surface
• Surface Edge
• Parallel and Coplanar Surfaces
• Surface Proximity
Chapter 5
Bayesian Attributed Hypergraph (BAHG)
• Definition
• Reasoning using the BAHG
• BAHG for Perceptual Grouping
Chapter 6
VALIDATION
• BAHG Network for the Detection of Comers and Surface Contunuity
• Experimental Results with 3D Data
Chapter 7
CONCLUSIONS
• Summary Outline
• Future Work

Figure 1.4: Visual summary of the thesis.

Chapter 2

Perceptual Grouping

Perceptual grouping in the domain of computer vision has been used as one of the first steps in the interpretation of images. Perceptual Grouping in computer vision is a study into the grouping of features sets extracted from images captured on computer from a sensor for the purpose of facilitating "Object Recognition", "Geometric Modeling", or "Multi-view Registration".

All perceptual grouping approaches are based on the early research in psychology based on the concept known as *Perceptual Organization.* Perceptual organization has its origins in the domain of psychology commencing in the late 19^{th} century and resulted in the creation of Gestalt Psychology. These researchers studied how simple sensations could be grouped into compound, stable perceptions which exhibit form and structure. Figure 2.1 shows some of the typical groupings that have come to be recognized as crucial relationships between features and are known as the Gestalt Laws of Organization.

The Gestalt Laws of Organization are still regarded as important in the psychology community, but as shown in figure 2.1 these are mostly still in a qualitative form. According to the Gestalt Law of *Pragnanz*, (goodness of form), the pattern mostly perceived by the human is the one which is the simplest and most stable. Several researchers have

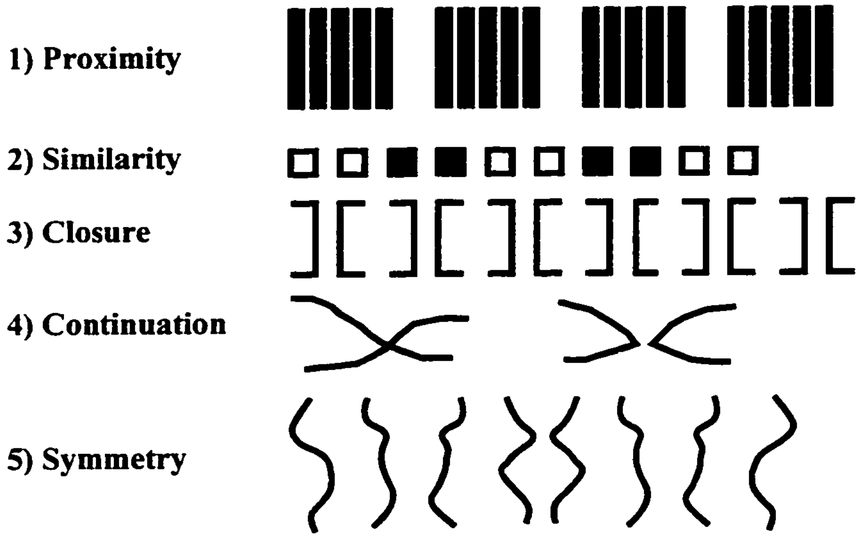


Figure 2.1: Gestalt Laws of Organization.

tried to quantify the idea of Pragnanz by using such methods as counting the number of edges [37], minimizing the information content of a group [36] [64], using the notion of non-accidentalness [54, 79], and recently a more comprehensive study using energy minimization methods [57].

Recent clinical experiments [75] have contributed some evidence that humans rely much more on higher-order structures in detecting regularity in scenes. In that work it was discovered that humans could detect regularity from dot patterns that contained mirror, translational, or rotational symmetry more readily if the images being viewed could be grouped into higher-order quadrangles and would have difficulties in detecting regularities if only lower-order groupings existed in the dot patterns.

Original studies into perceptual grouping in the computer vision domain concentrated on the grouping of features into the original Gestalt Laws. The perceptual groupings offered sufficient constraints among the features so that object recognition, geometric modeling, and multi-view registration were enhanced. The basic 5 perceptual groupings of proximity, similarity, closure, continuation, and symmetry are still widely used in computer vision in particular for object recognition. Perceptual grouping has also formed the foundation for indexing criterion in object recognition from 3-D sensory data [81, 19, 27]. This original work focused on the recognition of small objects that could be easily manipulated. More recent work has moved into environment modeling by partial recognition of components in the 3-D range data [44, 33] trying to use more generic models and the aid of humans to complete the models.

Generally in the domain of object recognition the term "perceptual grouping" is not commonly used to describe the constraints among the features that are used to guide the recognition process. Perceptual grouping has normally been closely associated with the perception of 2-D images even though many of the fundamental 3-D indexing constraints are based on the detection of continuity and proximity. Even though there exists similarities between object recognition indexing constraints and perceptual grouping relations the significant differences are the object models. In perceptual grouping the object models tend to be more generic than those used for object recognition and precision tends to be less emphasized than for object recognition. For perceptual grouping, qualitative relations are more important than precision and the objective is to determine relationships among the features. This is similar to the objectives for environment modeling where the more useful information is not object location and dimensions but the general shape and relationship of the surfaces among each other.

There are many different approaches to perceptual grouping, from deterministic methods to uncertainty propagation approaches. Deterministic approaches use thresholds for each grouping stage and grouping decisions are made at an early stage in the process. Uncertainty in the data is quickly discarded and cannot propagate to any higher levels in the grouping process. Uncertainty propagation techniques use measures of uncertainty of the hypothesized structures from the feature sets and propagate these values among other hypothesized structures. They are better suited for fragmented data and situations where evidence from several sources needs to be combined at different points in a decision algorithm. Uncertainty propagation techniques will generally be more costly than deterministic approaches due to the time taken to propagate the uncertainty throughout the other hypothesized structures, but are better suited for sensory data that contain many disjointed features sets. All approaches can lead to exponentially growing computational times if allowed to hypothesize the combination of all feature sets and therefore some basic context based heuristics must be used to limit this exponential growth.

In perceptual grouping two very basic issues are being addressed:

- Geometric measures are defined that compute a distance of compatibility between the set of features to a hypothesized perceptual grouping. These measures consist mainly of geometrical and topological relations among the feature sets. These Perceptual Grouping Measures add to the domain knowledge among surfaces.

- A particular perceptual grouping procedure is required that controls the sequencing for computing the perceptual grouping measures and instantiates a hypothesis for a grouping of features. This is the procedural knowledge for the perceptual grouping operation.

There does not exist any one particular approach for defining geometric measures and several general approaches have been developed depending on the domain of application. An overview of some perceptual grouping geometric measures for particular domains will be presented to introduce the reader to this work. This is followed by a number of perceptual grouping procedures which can be described by the generic algorithms being used.

2.1 Perceptual Grouping Geometric Measures

2.1.1 Measures for 2-D Image Points

Compatibility measures for 2-D points have been more commonly used in perceptual grouping research that have investigated the psychology of human perception. It is worth noting the work in "Geometric Probability and Stochastic Geometry" and the overview article by Small [71]. The main intention of that research was to determine a probabilistic model for the behaviour of points that form geometric patterns. An excellent model which is applicable to a variety of situations involving random points and geometric probability is the homogeneous Poisson process shown in equation 2.1.

$$P[N(A) = n] = \frac{(\lambda|A|)^n exp(-\lambda|A|)}{n!} \qquad (2.1)$$

where $N(A)$ is the number of points that reside in the geometric set A and λ is the Poisson parameter.

The primary issue then is determining appropriate measures for particular geometric shapes that are an estimate of their size and shape. These approaches have been implemented for the determination of architectural excavation sites given the positioning of rocks on the earth's surface [71].

2.1.2 Measures for 2-D Edges

The most studied area of perceptual grouping in 2-D images has been in the grouping of edges [53, 58, 31, 68, 38, 22, 78, 62, 55, 72, 2]. Geometric measures have been defined for the detection of continuity, symmetry, and proximity. Some of the common types of groupings for lines are shown in figure 2.2.

In an approach based more on psychological experiments, Wagemans et. al. [75]

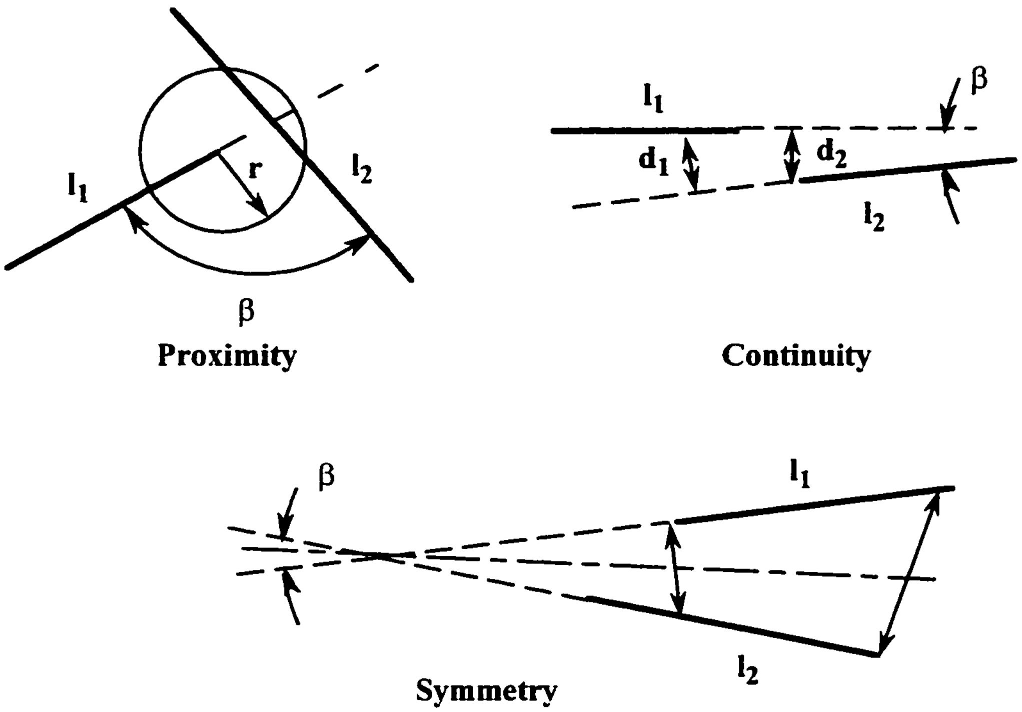


Figure 2.2: Common edge groupings.

noticed that humans tended to group points into quadrangle shaped patterns. They discovered possible groupings among a set of 4 points p_i, p_j, p_k, and p_l in an image if they could minimize the following equation.

$$\sum_{p_i p_j p_k p_l} \exp^{|\phi(p_i,p_k)-\phi(p_j,p_l)|+|\phi(p_k,p_i)-\phi(p_i,p_j)|}. \tag{2.2}$$

for the angles as shown in figure 2.3 defined by a set of 4 points in an image. This equation tends to zero in the case of existing symmetry for the quadrangle structure (i.e. possible trapezoids or parallelograms).

Wageman never considered features beyond those of dots in an image but did add credibility to the belief that humans can perceive invariant patterns that go beyond

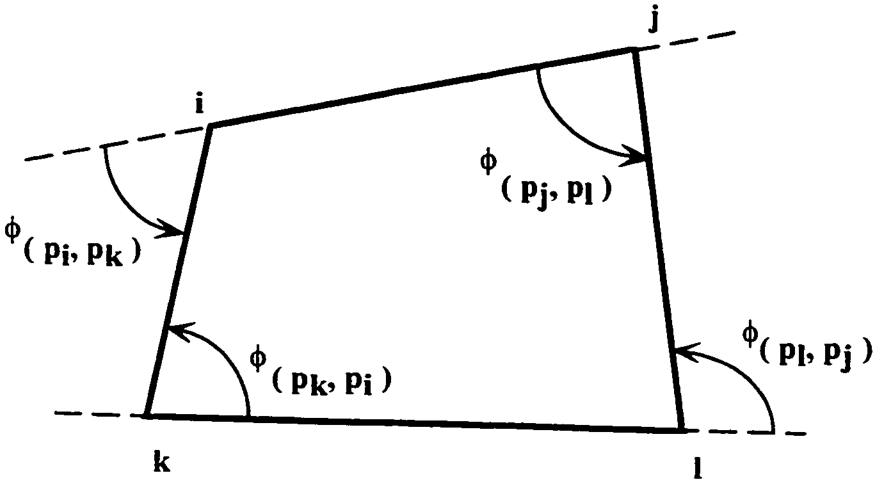


Figure 2.3: Quadrangle structure and angles used for measurement.

simple two line relationships.

Many of the examples presented in the literature look at defining measures among edges for continuity, symmetry, proximity, and closure but little effort has gone into studying the mathematics of geometry and uncertainty. The work by Durrant-Whyte [23] on "Uncertain Geometry" has set the foundation for a mathematical treatment of uncertainty in geometry. This work though has been primarily used for robotics motion estimation and not perceptual grouping.

2.1.3 Measures for 3-D Features

Perceptual grouping of 3-D edges have been used for the construction of 3-D wire frame models of manufactured parts [24] and indoor environments [3]. These early approaches tended to focus on the grouping of 3-D edges extracted from stereo cameras rather than on the grouping of surface data. Perceptual grouping of 3-D edges has also been applied in reducing the amount of time required to match features across multiple sensor scenes [85], due primarily to the fact that they are view invariant. Much of the early work in object recognition used perceptual grouping among 3-D edges to reduce the search space among the models. For example. in Grimson's work in object recognition [34] he proposed the use of binary constraints among two 3-D lines in space. These constraints were angle . distance. and component constraints which are primarily proximity and symmetry perceptual grouping measures. Several excellent survey papers in edge based object recognition have been published. for example [6. 7].

The core work in the processing of 3-D data has historically been in the domain of the extraction of surface data from a series of 3-D points. The non-uniformity of the data has a significant impact in the extraction of surfaces such that it is necessary to consider as much evidence as can be gathered from the sensor or other sensors for this process. Surface reconstruction is a low primitive form of perceptual grouping that considers the grouping of the 3-D data points into surfaces with smooth transitions and continuity. Several excellent books exist on this subject [8] and Boulanger [15] has a good tutorial on the analysis of range data.

For 3-D image understanding further relationships among the surfaces are crucial and perceptual grouping relationships like surface intersections. parallel relations. and coplanar measures are commonly used to limit the search space. Two common planar surface relationships are the parallel (symmetry for planar surfaces) and coplanar measure (continuity for planar surfaces) shown in figure 2.4.

For environment modeling the focus is on the reduction of data and hypothesizing shape. structure. and relations among the 3-D surfaces. The focus shifts from surface extraction to the discovery of fundamental groupings among the surfaces so that one can infer junctions and surface continuity. For example, hypothesizing the existence of corners and continuous surfaces is crucial for re-constructing a model from the sensory data. Several approaches have been proposed for the detection of corners for indoor environ-

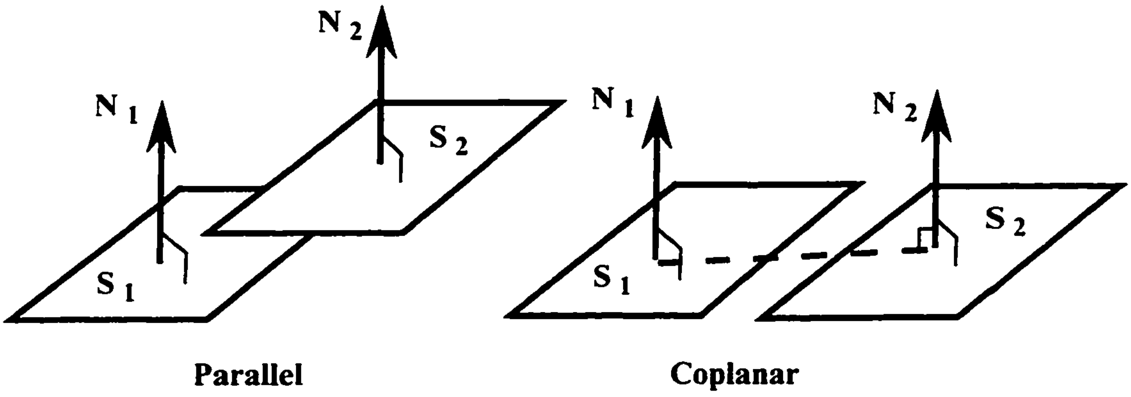


Figure 2.4: Common planar surface groupings.

ments [60. 70. 39] and the detection of ridges and valleys for outdoor environments [28]. Unfortunately the knowledge required to perform these surface groupings has been embedded in procedures that are not easily accessible and none of the approaches have tried to maintain a belief value in the formation of the groupings. Therefore the results have to be taken at face value without the ability to determine why particular surfaces were not joined.

2.2 Perceptual Grouping Approaches

2.2.1 Deterministic Approaches

Production Rules

Production rules are by far the most popular approach to perceptual grouping in which a set of heuristic rules are used to act upon the geometric measures among the features sets. These approaches can have positive results if the set of rules are small enough to be manageable and the data is relatively free of abnormalities. In the domain of wire frame modeling the thrust has been in the grouping of edges that form closed polygons. The most common approach is to develop a set of heuristics to classify the type of junction [12, 55, 35] and proceed with an algorithm to close the set of edges into a polygon. An example of this type of procedure is the edge tracking algorithm developed by Wong et al. [83] that tries to choose the best branching direction that will maintain the tracking within the same half-plane of the other edges in the group.

Production rules can be deterministic or can use uncertainty values to maintain and propagate the uncertainty among other heuristics. Uncertainty calculi developed for rule based systems have shown to have serious shortcomings [42] in particular to coherent inferencing and for that reason few perceptual grouping systems have been developed that use uncertainty propagation techniques with heuristic rules. Most, if not all, of the systems that use production rules have chosen to make a deterministic choice at an early stage in the grouping process.

Constraint Satisfaction Systems

Another interesting approach to perceptual grouping, used primarily in object recognition, is the use of constraint satisfaction networks. These were used successfully in 3-D MOSAIC [76, 4] for hypothesizing polyhedral surfaces from aerial views of buildings. The underlying theory is based on the concept that faces, edges, and vertices mutually constrain each other in forming a global interpretation of the objects in the scene. The approach is one of choosing low-order features which have a high probability of corresponding to individual object parts. The geometrical constraint functions can be both supportive and competitive in nature.

The network uses geometric constraints among edge features that add or subtract support in the formation of particular groupings. These groupings were perceptual groupings among straight lines that eventually could lead to the detection of "U-contours" and

"Rectangles" among those lines. Mohan and Nevatia have shown that the more complex operations are: the extraction of parallel line features ($\mathcal{O}(n^2)$); and the formation of the network ($n\mathcal{O}(np^2)$), where n is the number of line segments and p the number of parallel line segment groups. This concurs with the exponential nature of trying to check for groupings among all the features.

Constraint searches using *interpretation trees* [34, 73, 19, 29] have been widely used for object recognition and again many of the fundamental constraints have been based on perceptual grouping properties.

In a very comprehensive study into the investigation of perceptual organization in human and machine vision McCafferty [57] used energy minimization techniques to solve a set of constraints that could be interpreted as a perceptual grouping. In his approach the problem was to find the grouping g which minimizes:

$$E_{Total}(g) = \lambda_1 \|E_{Prox}(g)\|^2 + \ldots + \lambda_n \|E_{PG}(g)\|^2. \tag{2.3}$$

where E_{PG} represents geometrical constraint functions that enforce perceptual grouping and λ_n represents regularisation parameters. For example, in the above equation E_{Prox} represents a geometrical constraint function for proximity and λ_1 is the regularisation parameter for proximity. McCafferty expressed the fact that, similar to human perception there does not exist one ideal grouping but several groupings that can dominate depending on the regularisation parameters. Therefore he also chose to determine the set of groupings that minimized the total energy of all the groupings i.e.,

$$min \sum_{k=1}^{j} E_{Total}(g_k). \tag{2.4}$$

This is akin to the multi-resolution problem in computer vision where many possible perceptions are possible depending on the resolution and context of the domain.

2.2.2 Uncertainty Management Approaches

Several approaches have been investigated for either perceptual grouping or object recognition that take into account uncertainty in the reasoning component. These systems generally offer more information to the user and options of possible perceptual groupings.

Fuzzy Logic Inferencing

Another grouping methodology developed by Kang et al. [45] uses the Gestalt Laws for grouping and manages the uncertainty in the measures among the features using fuzzy logic as the basis for making decisions. Fuzzy logic approaches are primarily a heuristic implementation of perceptual grouping using fuzzy sets instead of discrete values. Some recent work has investigated the use of fuzzy sets with the grouping of 3-D sensory data [46] but these systems still suffer from the intractability when many perceptual grouping rules exist.

Bayesian Networks

Bayesian networks have been used in the domain of computer vision for several applications. Some particular ones worth citing are. object recognition [11. 47. 59. 77. 49]. multi-agent vision systems [43]. road scene recognition [30]. tracking of dynamic objects in images [32]. and perceptual grouping for 2-D images [68]. The construction of the Bayesian network is based on the idea of representing the attributes and relations among the features extracted from the image in a network and using Bayesian probability as a measure of uncertainty in the formation of groupings. The two principal things to determine are the causal relations among the feature sets and the geometrical compatibility functions that are used as a measure between the feature sets and the hypothesized groupings.

Bayesian networks take advantage of a-priori knowledge and use Bayes theory to represent relational knowledge among the nodes in the network. They leverage on the use of causal relations among the hypothesized groupings to help design the network. A significant part of the research is in determining appropriate causal relations among the feature sets. Bayesian networks also require a formalism to be developed that can compute the conditional probabilities among the hypothesized groupings. This is generally done by either performing many tests and determining a probability density function or subjectively.

Bayesian networks offer a relatively simple manner in which to manage uncertainty for the grouping of features extracted from a sensor. but the difficulties are still in being able to determine what events are relevant and how to determine the conditional probability values of these events. Techniques for transforming feature measurements into uncertainty values are still in their infancy.

An example of a Bayesian network is the perceptual inference network (PIN) [68]. reproduced in figure 2.5. for the detection of particular polygonal shapes from 2-D edges.

This is a fairly interesting system because it is the closest in similarity to the approach taken in this thesis and therefore warrants further discussion. The PIN network operated on a set of ribbons extracted from an intensity image. Similar to other perceptual grouping processes the most time consuming operations occur at the extraction and first attempt to group the ribbons based on continuity. This was done using standard graph matching algorithms with computing complexities in the order $\mathcal{O}(n^3)$ and $\mathcal{O}(n^2)$

Dempster-Shafer Networks

The Dempster-Shafer Theory is an approach for representing and combining uncertainty from aggregate sources as belief values. It differs from Bayesian formalism in that probability values can be assigned to subsets of the set of all possible outcomes of a random

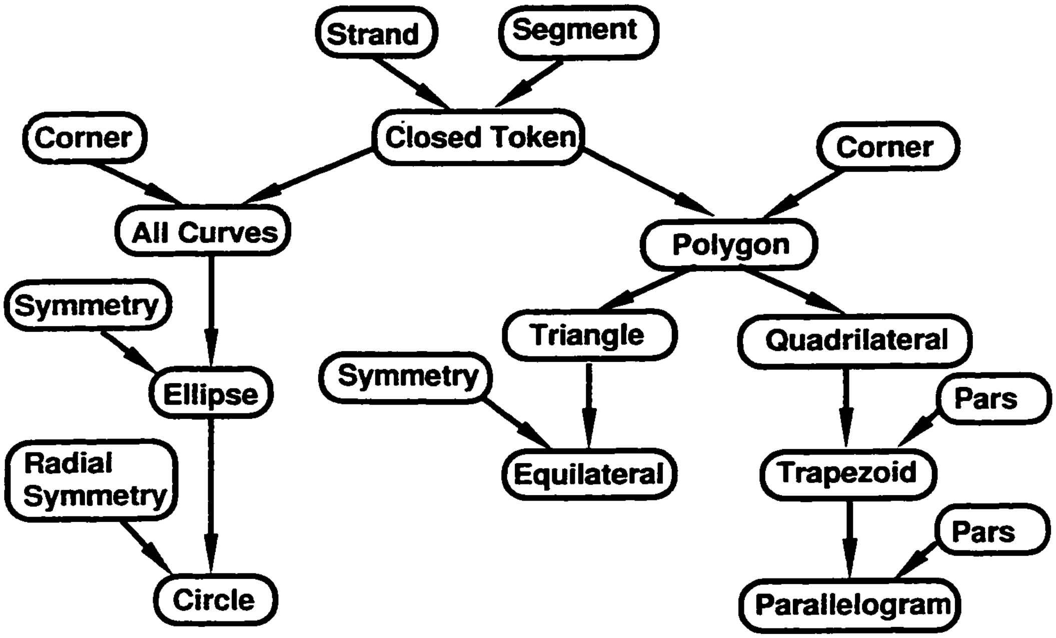


Figure 2.5: Perceptual inference network as specified by Sarkar and Boyer.

experiment instead of only singleton subsets. It also differs in that the law for additivity for belief in disjoint propositions does not apply. Dempster-Shafer formalisms are suitable for situations when considering a hierarchy of hypotheses and it may not be clear which particular subclass of a node actually exists. In other words it is possible to compute a belief value in the formation of a particular parent node without knowing the beliefs in each individual subset of the parent.

The hierarchical blackboard object-recognition system called PSEIKI developed by Andress and Kak [1, 40] is one of the few perceptual grouping systems that use the Dempster-Shafer formalism for evidential reasoning. It is based on a two panel five level blackboard that decomposes a model of an object into its sub-parts and then compares features from these sub-parts to those detected by the sensor. The sensor feature sets are grouped based on proximity and junction formations and are compared to those similar formations in the model.

Dempster-Shafer formalisms are not easy to implement or understand and can lead to worst case behavior computational times that are exponential in the size of the hypothesis set. This has led to the addition of heuristics to the grouping process that can detect disjoint subsets that would not have to be combined. In other terms, it becomes necessary to limit the size of the number of hypotheses by using relational constraints among the surfaces. Even with these limitations the Dempster-Shafer formalism is a powerful tool for the management of uncertainty and it is beyond the scope of this thesis to perform a detailed assessment between Bayesian and Dempster-Shafer formalisms for scene analysis.

2.3 Summary of Perceptual Grouping

Perceptual grouping is a vital preliminary step for the interpretation of sensory data in particular computer vision sensor data. It forms the foundation for object recognition, computer modeling, and multi-view sensor registration. Perceptual grouping using 2-D intensity data and edges from stereo systems has been explored since the early 1980's. This has not been the case as much with 3-D surface data. In the domain of 3-D vision the majority of perceptual grouping operations have been embedded in object recognition systems that primarily have dealt with the recognition of manufactured parts. In those domains very explicit models have been used as references for computing the compatibility functions since in most circumstances Computer Aided Design (CAD) data are available for the components.

Once one tries to create models of large indoor environments the perceptual grouping of surfaces becomes important in deciding how and which surfaces are to be brought together. Much redundant data exists and useful CAD models are impossible without the grouping of surfaces. It is also important to create an infrastructure that will allow the accumulation and distribution of evidence and uncertainty. This system must be expandable to handle different forms of evidence, therefore it must separate procedural from declarative knowledge to facilitate the addition and removal of information.

This thesis has adopted the use of a Bayesian network for managing the evidence gathered from 3-D sensory data for the purpose of inferring higher order perceptual groupings of the surfaces extracted from the 3-D data. Bayesian networks offer a well developed mechanism for the management of uncertainty in reasoning. The use of Bayesian networks for the Perceptual Grouping of 3-D data is developed in greater detail in the next chapter.

Chapter 3

Bayesian Networks for Perceptual Grouping

In this thesis. Bayesian networks are investigated as a tool for perceptual grouping of image features. The methodologies developed are generic enough to be applied to intensity images but examples have been derived primarily for the grouping of 3-D surfaces. The hypothesis brought forward is that the network be designed using causation principles but quantification of the network is done in the reverse direction. from data to hypothesis. using compatibility functions that measure how well a set of features fits a hypothesized model. All perceptual grouping algorithms take advantage of causation indirectly by defining models that relate the detected sensory data to the possible geometric formations among the data. Bayesian networks offer a formalism for representing the causation among the groupings and an approach for maintaining a belief value in the formation of those groupings. The Bayesian network allows the encoding of the expected formations that the sensor may detect when viewing a scene and infer from the features a belief value of the existence of the grouping of the features.

3.1 Bayesian Networks

A Bayesian network is a particular type of network that uses Bayes probabilistic theory to manage the uncertainty in the values of the variables of the network. A causal network is a directed acyclic graph with the nodes representing a hypothesis of the existence of a proposition and the arcs signifying the causal relationship from one proposition to the next. For example. figure 3.1 shows three simple causal networks that depict the types of connections by which evidence can be transmitted in Bayesian networks. These three methods are: serial. divergent. and convergent connections.

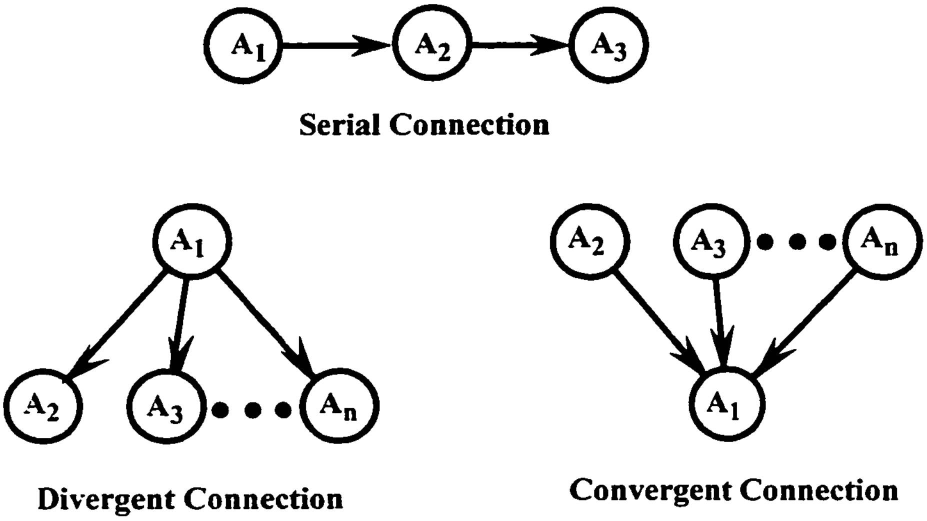


Figure 3.1: The three different types of connections in a Bayesian network.

Variables in a causal network are random variables that represent events and can have any number of discrete states. In this thesis only Boolean variables are considered and these represent the existence of a perceptual grouping.

The network depicts dependencies among the nodes and the type of connection determines how evidence is transmitted along the network. The evidence of a variable is a statement of the certainty of the state of the variable. *Hard* evidence gives the exact state of the variable and the variable is referred as being *instantiated*, while *soft* evidence is a prediction of the state of the variable. For the perceptual grouping Bayesian networks proposed in this thesis only the nodes that represent actual sensory features are instantiated. The rest of the nodes represent hypotheses of geometrical formations among the features.

Several probabilistic quantities can be computed using Bayesian networks, like expectations and best explanations. One of the most common inferences desired involves the calculation of posterior marginals for a particular set of variables. This is called belief assessment or belief updating. Belief values in a Bayesian formalism pertains to the computation of the conditional probability $P(A_i|\mathbf{e})$ where A_i is a particular variable and $\mathbf{e}$ are the instantiated variables.

Bayesian networks use Bayes probability theory to quantify the networks uncertainty values and offer a compact representation of the joint probability $P(U)$ corresponding to a discourse of variables $U = \{A_1, A_2, \ldots, A_n\}$ represented in a Bayesian network. All that needs to be specified to compute a complete joint probability of the Bayesian network are the conditional probabilities of the variables conditioned on their parents. From $P(U)$ it is then possible to calculate a belief in the state of event A_i given the instantiated evidence $\mathbf{e}$, i.e. $BEL(A_i) = P(A_i|\mathbf{e})$.

Definition 1 *A* **Bayesian network** *is a directed acyclic graph $BN = (U, De)$ where $U = \{A_1, \ldots, A_p, \ldots, A_q, \ldots, A_n\}$ are a set of discrete random variables represented as nodes in a graph and $De = \{\ldots, de_{pq}, \ldots\}$ is a set of directed edges. The directed edge de_{pq} connects the node A_p to A_q. The joint probability of the discourse of variables U is given as $P(U) = \prod_i P(A_i|pa(A_i))$ where $pa(A_i)$ is the parent set of A_i.*

The foundation of this definition is the relation between conditional independence and the concept of *d-separation*. Evidence among the variables is transmitted as long as they are not d-separated. This notion is the fundamental concept for the transmission of evidence in Bayesian networks. If two variables (nodes) are d-separated then they are conditionally independent.

Definition 2 *Two variables in a causal network are* **d-separated** *if for all paths between A_i and A_j there is an intermediate variable A_k such that either, the connection is serial or divergent and the variable A_k is instantiated, or the connection is convergent and neither A_k or any of its children have received evidence.*

The concept of d-separation is best explained using the connections depicted in figure 3.1. For serial connections evidence from node A_1 will influence node A_3 if and only if node A_2 has not been instantiated. If node A_2 is instantiated then variables A_1 and A_3 are independent. For divergent connections evidence among the children nodes $A_2 \ldots A_n$ will not be transmitted through an instantiated parent node A_1. Basically, variables $A_2 \ldots A_n$ are dependent as long as the value of A_1 is unknown. Once the state of A_1 is determined then the variables $A_2 \ldots A_n$ are independent. Convergent connections are slightly more complicated. Evidence from the parent nodes, $A_2 \ldots A_n$, is not transmitted among each other. They are independent of each other. On the other hand if evidence influences the belief value of A_1 then the parent nodes, $A_2 \ldots A_n$, become dependent and their belief values change appropriately. The blocking relations described by the d-separation concept are reflected as conditional independence in Bayes theory. The variable A_1 and A_2 are independent conditioned on variable A_3 and therefore $P(A_1|A_2, A_3) = P(A_1|A_2)$.

Several approaches and systems are already available for the development of Bayesian networks. These include HUGIN [41], JavaBayes [21], ERGO [5], and Microsoft's Belief Network software. This thesis leverages on the development of JavaBayes and explores primarily how Bayesian networks can be used in a more comprehensive system for the perceptual grouping of 3-D surfaces.

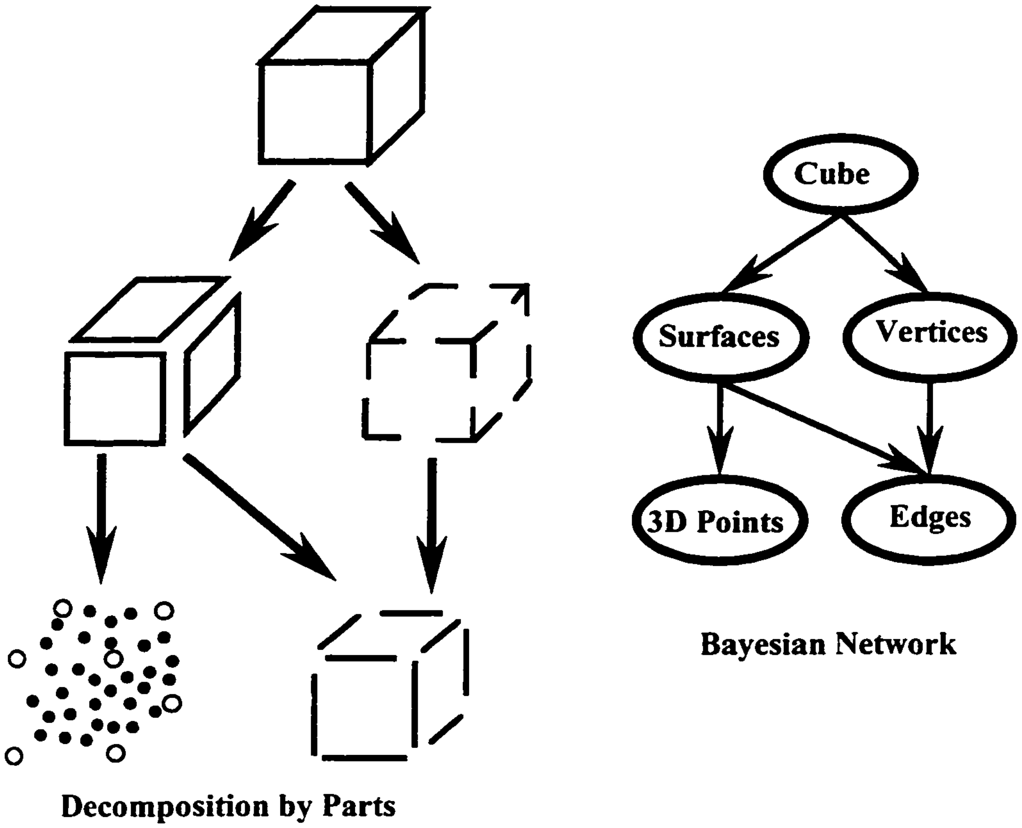
3.2 Bayesian Networks for Perceptual Grouping

In this thesis Bayesian networks are used for the specification of the grouping of surfaces and the calculation of belief values in those formations. The nodes in the network represent the existence of geometric formations resulting from the grouping of the 3-D sensory data and the arcs connecting the nodes represent a causal relationship among the formations. For example, a formation of a set of coplanar surfaces causes the existence of two parallel planar surfaces. In this particular case the environment is being modeled first by the use of a Bayesian network and evidential reasoning occurs within the Bayesian network. Each parent node in the network can be considered as a hypothesis of a composite feature that resulted from the evidence represented by the children nodes. This evidence is reflected as constraints on the attributes and/or relation among the features detected by the sensors.

3.2.1 Network Structure by Part Decomposition

Deciding on the structure and variables of a Bayesian network is a mutual exercise. This is done by applying a theory of hierarchical decomposition by parts used for the modeling of objects [48, 9, 10, 56]. An example of this type of modeling is demonstrated in figure 3.2 that illustrates a cube decomposed into its components that directly maps to the Bayesian network shown to the right of the decomposition hierarchy.

The process of decomposing by parts can lead to many solutions and must be guided by the type of sensory data and any intermediate groupings that can be extracted from that data while gathering evidence for the formation of the object. This is done by considering the type of constraints that need to be applied to the data so the model



Bayesian Network

Decomposition by Parts

Figure 3.2: Decomposition by parts of a cube.

is achieved. Traversal from the leaf nodes (feature sets) up to the root node (object) implies another constraint applied to the data on top of those used by the children. It is possible to apply all these constraints at one time but this would lead to a complicated function with little ability to add evidence from other sources into the network. For example, the simple decomposition of the cube depicted in figure 3.2 has intermediate formations depicting the planar surfaces and junctions that have been derived from a set of 3-D points and/or edge data. These two branches in the network account for the the availability of both range values and edge data. If any of these were not available, the associated branch in the network would have to be eliminated.

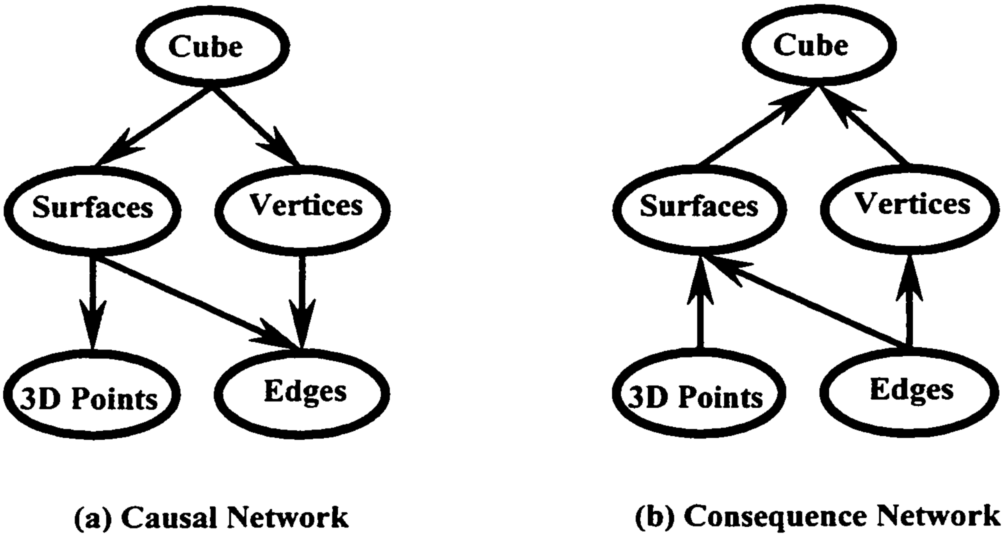
The leaf nodes in the network tend to correspond to features that are extracted from the sensor. these are the feature variables and tend to correspond to readily detectable 2-D or 3-D features. for example. edges. 3-D points. intensity. and/or color. It is at this point that any hard evidence will be entered into the network. All other intermediate nodes correspond to hypothesized formations and groupings among the feature sets. These may also be relevant groupings depending on the goals of the application. For example. applications interested in the navigation of autonomous vehicles are more interested in traversable surfaces. For environment modeling the purpose is to detect particular general surface formations that could be used for extracting overall room shapes.

Throughout this operation it is important to keep in mind that the purpose of the Bayesian network is to compute estimates of belief in the formation of unknown groupings. i.e. those that cannot be readily detected but must be hypothesized from the data. The intermediate variables and root variables symbolize the formation of groupings that are not readily detectable. This is ideal for the domain of environment modeling from 3-D data since nearly all formations are hypothesized from the data.

3.2.2 Quantifying the Network

Bayesian networks tend to model causation. the relation of cause to effect but can easily be converted to model consequences. events to hypotheses. by inverting the direction of the Bayesian network. This type of network will be referred to as a consequence network in this thesis due to the fact that it models the inference of evidence to hypothesis. A network that models cause to effect is of course known as a causal network.

The example depicted in figure 3.2 utilized the principal of decomposition by parts to create the perceptual grouping Bayesian network for the formation of a cube. Decomposition by parts results in a hierarchical geometric model commencing from an object to its components. This can easily be translated directly to a causal network, see figure 3.3 (a). The same network can be inverted and presented as a consequence network as shown in figure 3.3 (b). These networks are identical except for the direction of the edges.



(a) Causal Network

(b) Consequence Network

Figure 3.3: Causal and consequence networks for the cube formation depicted in figure 3.2.

Mathematically, the networks are similar in that the propagation algorithm for computing the belief at each node does not change, only the direction of propagation in the network are opposite to each other. The difficulty exists when additional relationships need to be added to the network for the introduction of evidence from new sources of information. This is apparent from the computation of $P(Cube|Surfaces, Vertices)$ needed for the consequence network depicted in figure 3.3 (b), where it is necessary to combine values computed from the nodes $Surfaces$ and $Vertices$. An approach to this problem is presented in section 3.3. This thesis then maintains that modeling of the grouping process should be performed by causation but when actually quantifying the network this is performed simpler in the consequence direction using algorithms refered to as compatibility functions.

Compatibility functions use the attributes and/or relations among the features to compute the conditional probabilities required for the Bayesian network. see section 3.3. Compatibility functions. $f_{PG}(\mathbf{A})$, are based on geometric constraints applied on a set of attributes **A** computed from the sensory data and measure how well these features match the perceptual grouping *PG*. These are not directly the same groupings that are represented by the nodes in the Bayesian network but are primarily fundamental constraints applied to the features that when combined with other constraints can lead to the formations depicted by the Bayesian network nodes. These perceptual groupings conform closer to those fundamental sets defined originally by the Gestalt laws of organization and depicted in figure 2.1 in chapter 2. i.e. symmetry. proximity. continuity. .... The difference is that they have been developed for 3-D surface data. From this point on. compatibility functions will be referred solely as f_{PG} without the argument **A**.

For example. the relation between the formation of surfaces from a set of range values is based upon a proximity compatibility function. f_{plan}, that measures how well the points represent a planar surface. Beyond this. other compatibility functions are used to measure how well the surfaces conform to a cube. This is similar in concept to the system developed by Levitt et al. [47] in that the belief values of the nodes in the Bayesian network represent a probability of the existence of that feature or geometrical formation of features. It differs in that these compatibility functions are not derived from the hierarchical structure imposed by manufacturing parts. but from generic formations of indoor environments. Data from these environments tend to be sparse 3-D points and several intermediate groupings are required that are not apparent from manufactured parts.

Compatibility functions however tend to favor the use of consequence networks over causal networks. There are two reasons for this. First, compatibility functions determine how well evidence fits the desired hypothesized model. For example, the perceptual groupings presented for 2-D edges in section 2.1.2 offer a good example of how compatibility functions are a measure of how well the evidence fits the desired hypothesized model. These measures of symmetry and proximity among the edges can be used to measure how well a set of edges matches common 2-D geometrical shapes like, triangles and parallelograms. Secondly, compatibility functions cannot model the conditions when the hypothesized formations do not exist, which is a requirement for causal networks. The consequence of this is that the conditional probabilities are set to zero for those situations when the states of the parents are FALSE. When these conditional probabilities are used in a causal network the result is that the belief value for the formation of the parent node is unity, i.e. the hypothesized formation exists irrelevant of the values calculated using the compatibility functions. Basically the result is independent of the compatibility function and renders the use of these irrelevant in a causal network.

If this procedure is now reversed so that the belief values are computed from evidence to hypotheses then more meaningful results are apparent since the conditional probabilities are combined to compute belief values in the formation of the hypothesis. This approach is closer to the Perceptual Inference Network [68] except that the network is designed using causation and the compatibility functions are associated with particular links.

3.3 Conditional Probabilities and Compatibility Functions

In order to place each compatibility function on the same level of reference, a mapping function is used to map the compatibility functions to a certainty value that exhibits the following properties.

- It is bounded between the interval (1, 0).

- It is a decreasing monotonic function, so that a compatibility value equal to 0 signifies a certainty value of 1. This is equivalent to a perfect match between the features **A** and the perceptual grouping *PG*.

Several mappings will exhibit these properties, such as logarithmic, power, linear, and trigonometric functions. For this particular case the declining S-Curve function, used extensively in Fuzzy sets [20], is proposed as a mapping function. The declining S-Curve favors rapid changes in the mid-section of the curve and small changes at both the beginning and end points. This reflects the condition where for compatibility values close to 0 there will not be much change in the certainty value but rapid changes occur once the value moves away from 0 significantly. This is a desirable property that seems to conform well with human judgment where for a particular range of values the certainty in the occurence of a particular formation is high but reduces quickly after a particular point. This mapping is depicted as a graph in figure 3.4 and represented by equation 3.1 where parameters γ and β affect the rate of change in the S-Curve.

$$S(f_{PG}, 0, \gamma, \beta) = \begin{bmatrix} 1 & \rightarrow & f_{PG} = 0 \\ 1 - 2(f_{PG}/\gamma)^2 & \rightarrow & 0 < f_{PG} \leq \beta \\ 2((f_{PG} - \gamma)/\gamma)^2 & \rightarrow & \beta < f_{PG} \leq \gamma \\ 0 & \rightarrow & f_{PG} > \gamma \end{bmatrix} \tag{3.1}$$

Mapping the compatibility function to a certainty value using an S-Curve mapping offers several advantages to the user over simply using the compatibility function.

- It offers a clean separation between the definition of compatibility functions and conditional probabilities.

- It introduces a subjective measure into the definition of the conditional probability that can act as a method for defining the resolution of a match of the geometric

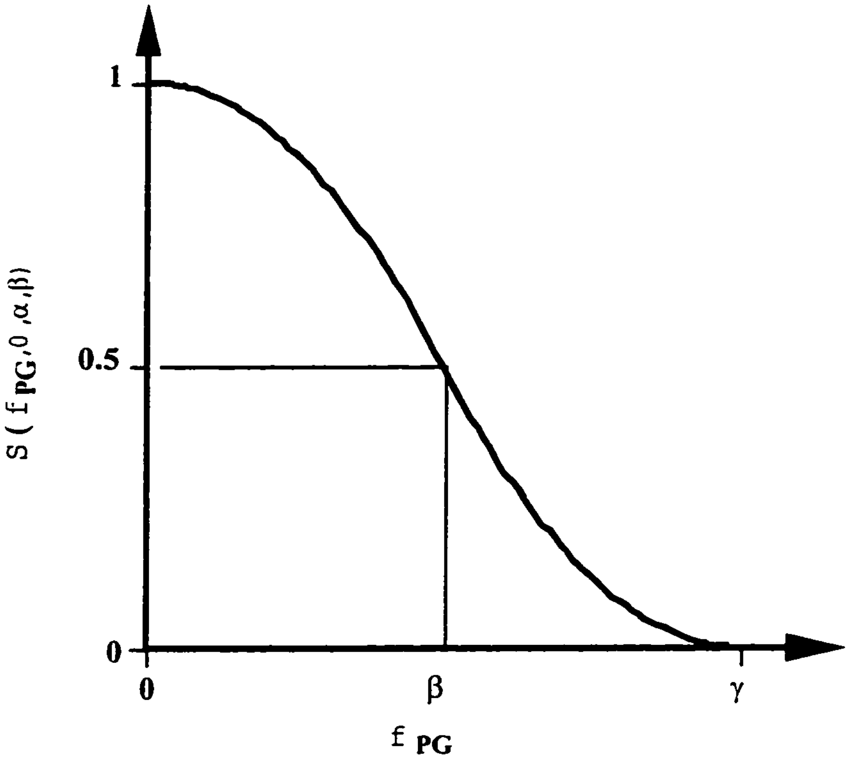


Figure 3.4: A declining S-curve.

features to the hypothesis.

- It places the results of the compatibility functions on a uniform level. The range of the values returned from compatibility functions can have unlimited dimension. the S-Curve mapping places all the results within the same bounds.

The S-Curves are the primary interface to the user and correspond to the manner in which a user can decide the bounds on the compatibility functions. The values for β and γ are determined subjectively or through experimentation. For example, a user of the system may subjectively decide that two surfaces whose normals are less than 10 Rads apart are parallel and those greater than 20 are not. These values then can be used correspondingly as β and γ. Alternatively, an experiment could be performed among a number of users to statistically determine the expected values for the angular difference between the normals that correspond to parallel and non-parallel surfaces. In the examples given in chapter 6 the values for β and γ were determined subjectively.

The S-curve mappings for the compatibility functions can be used directly as conditional probabilities for nodes with one parent. Take for example any of the nodes in a serial connection, similar to those depicted in figure 3.1 (a) or children nodes in divergent connections similar to those shown in figure 3.1 (b). For these type of connections the conditional probabilities can be stated as,

$$\begin{aligned} P(a_i|a_j) &= S(f_{PG}, 0, \beta, \gamma) \\ P(\neg a_i|a_j) &= 1 - S(f_{PG}, 0, \beta, \gamma) \end{aligned} \tag{3.2}$$

where A_j is the parent of node A_i and the states of any variable A_i are $(a_i, \neg a_i)$.

The above example only addresses conditional probabilities when the state of the parent node is TRUE, i.e. $A_j = a_j$ without taking into consideration the condition when $A_j = \neg a_j$. A complete specification of conditional probabilities requires that all states of the variables are covered. The compatibility functions do not address this condition properly and it is necessary to choose values that are logically correct. For single parent nodes the natural choice for these conditional probabilities are,

$$\begin{aligned} P(a_i|\neg a_j) &= 0 \\ P(\neg a_i|\neg a_j) &= 1 \end{aligned} \tag{3.3}$$

This implies that if the parent formation A_j does not exist then the grouping A_i cannot exist either.

For nodes with multiple parents, i.e. convergent nodes, the approach presented above cannot easily be applied but must be modified to account for the existence of the other parent nodes. Each directed edge de_{pq} has associated with it a compatibility function and corresponding S-Curve mapping. For the above single parent examples the edge de_{ji} corresponds to the edge between A_j and A_i and has associated with it a compatibility function f_{PG} and a mapping $\mathcal{S}(f_{PG}, 0, \beta, \gamma)$, where PG corresponds to a particular perceptual grouping. With this in mind the following calculation for conditional probabilities is proposed.

$$
\begin{aligned}
P(a_i|a_j, \ldots, a_m) &= \prod_{A_n \epsilon pa(A_i)} \mathcal{S}(f_{PG}(\epsilon_{A_n, A_i}), 0, \beta, \gamma) \\
P(\neg a_i|a_j, \ldots, a_m) &= 1 - \prod_{a_j \epsilon pa(A_i)} \mathcal{S}(f_{PG}(\epsilon_{A_n, A_i}), 0, \beta, \gamma)
\end{aligned}
\tag{3.4}
$$

where $f_{PG}(\epsilon_{A_j, A_i})$ is the result of the compatibility function f_{PG} corresponding to the edge associated with the nodes A_j and A_i.

Similarly to the single parent case above, when the state of any of the parent nodes A_j is FALSE then the following conditional probabilities apply.

$$
\begin{aligned}
P(a_i|a_j, \ldots, \neg a_k, \ldots, a_l) &= 0 \\
P(\neg a_i|a_j, \ldots, \neg a_k, \ldots, a_l) &= 1
\end{aligned}
\tag{3.5}
$$

where the states of $pa(A_i)$ correspond to $(a_j, \ldots, \neg a_k, \ldots, a_l)$.

Combining the declining S-curve values as a product is a conservative approach to the accumulation of evidence since when more evidence items are added the value of the product lowers. In particular, the weaker S-curve values can quickly lower the overall conditional probability, therefore having a significant impact. This is important if the emphasis is on making certain that all the evidence plays a significant role in the belief of the formations. Other combinations can be used. For example, an average of the S-curve values or perhaps choose the maximum or minimum S-curve value. The product rule is a simple and conservative approach without having to analyse the data.

3.4 Summary

In this chapter an approach was presented for the definition of a Bayesian network for the perceptual grouping of features extracted from a sensor. It was demonstrated that the theory of hierarchical decomposition by parts can be used to create the structure of the network. Conditional probabilities, required for the network, are determined using compatibility functions based on geometrical constraints among the features extracted from the sensory data. It was also demonstrated that the approach of using compatibility functions can be used for specifying conditional probabilities for the network but imposes restrictions in the direction of the network to be from evidence to hypotheses, i.e. bottom up.

The following steps were presented in the construction of a Bayesian network for perceptual grouping:

- Define a set of variables that represent possible perceptual groupings in the sensory data by using a decomposition by parts methodology where the desired formation is decomposed into sub-groupings and finally into the actual sensed features. These variables are boolean random variables that take on the values TRUE or FALSE to represent the existence of a particular geometric formation.

- From this decomposition the structure of the Bayesian network will be determined by mapping the links from a parent sub-group to its child with a directed edge. These are causal links representing the cause effect relation between the parent grouping to the child sub-grouping. During the decomposition by parts procedure the compatibility functions are defined for each link in the network by considering what perceptual grouping constraint function could be used in determining a certainty value in the quality of fit of the data to the perceptual grouping.

- For each compatibility function define a declining S-curve that will be used to map the values from the compatibility function to conditional probabilities.
- Define the conditional probabilities for each link using the procedure defined in section 3.3.

Chapter 4

Compatibility Functions for 3-D Perceptual Grouping

In this chapter several compatibility functions have been defined for the perceptual grouping of 3-D surfaces that use both attributes extracted from the surfaces and their respective boundaries. These compatibility functions try to measure some generic perceptual grouping operations among planar surfaces with the intent of detecting corners and continuity among the surfaces. Later in section 6.2 a Bayesian network is presented that uses these compatibility functions to compute the conditional probabilities of the network.

Five compatibility functions have been defined. these are: an edge compatibility function. a planar surface compatibility function. a parallel compatibility function. a coplanar compatibility function, and a surface proximity function. These 5 compatibility functions are not dependent on the particular network being deployed but can be used with other object recognition or perceptual grouping approaches for 3-D surfaces. Some of the compatibility functions are relatively simple and are common measurements in the literature. for example planar surface quality, coplanar surface, and parallel surface measurements.

Edge quality and proximity are unique approaches developed for this particular thesis. All approaches have been presented for coherency. Along with the development of the compatibility functions, examples are presented of the functions applied to actual 3-D data points. For consistency, the compatibility functions are applied to the same set of 3-D data points depicted in figure 4.1 and figure 4.2. Figure 4.1 is an intensity image while figure 4.2 is an isometric view of the 3-D data points. The 3-D data is captured at the same time as the intensity data and stored as an image and can be referenced using the image coordinates.

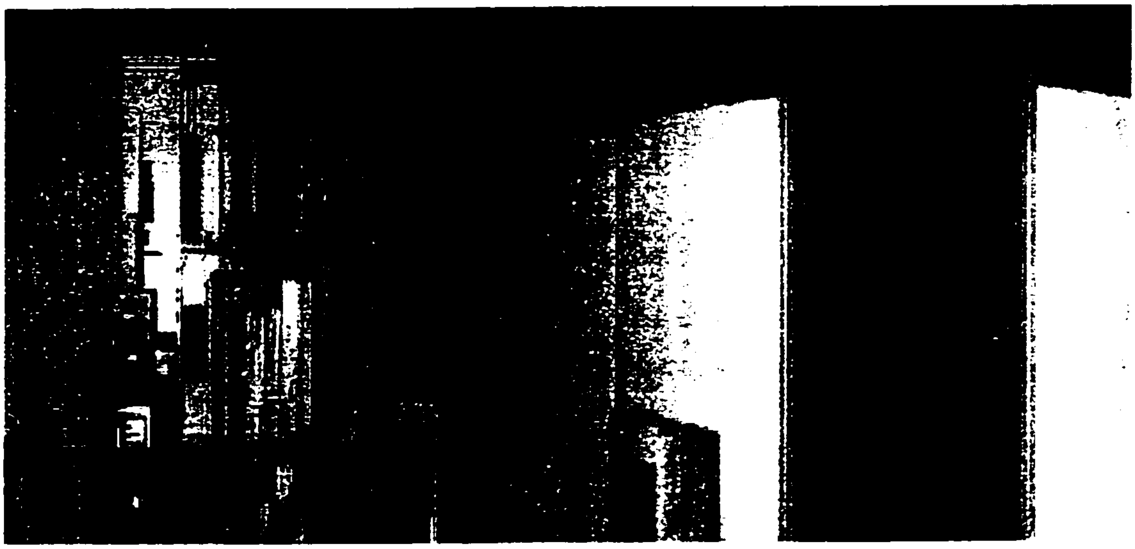


Figure 4.1: A typical intensity image of a room captured from the BIRIS portable camera.

This image was taken using a small portable 3-D laser scanner, known as BIRIS [13], of a laboratory measuring approximately 5m square. A schematic diagram of the top view of the room is shown in figure 4.3 showing the position of the camera at the center of the Cartesian coordinates and displaying a scan angle of approximately 140 deg to the left bottom of the figure.

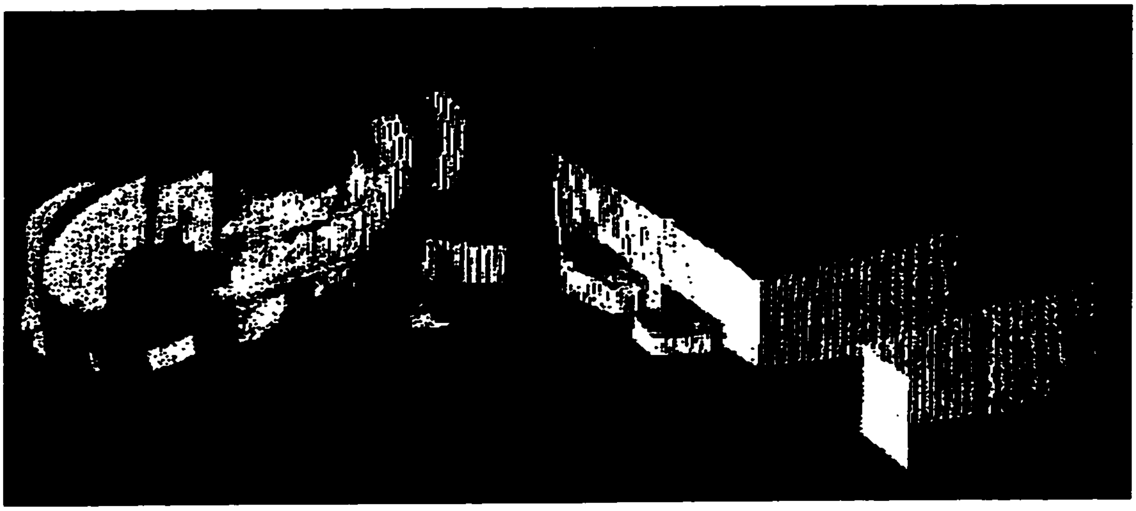


Figure 4.2: An isometric view of the 3-D data points corresponding to the points in figure 4.1.

As one can see there are many missing points and fragmented surfaces. To the right in the image is a hallway constructed of temporary partitions which can be clearly seen. but to the left is a series of bookshelves and table tops that are nearly indecipherable by the human eye.

4.1 Planar Surface Compatibility

Let us assume that the data is normally distributed about a planar surface and therefore all information about the planar surfaces distribution is available through its covariance matrix which is defined in the following manner.

$$Cov_{S_i} = \begin{bmatrix} (x_i - \bar{x}_i)^2 & (x_i - \bar{x}_i)(y_i - \bar{y}_i) & (x_i - \bar{x}_i)(z_i - \bar{z}_i) \\ (y_i - \bar{y}_i)(x_i - \bar{x}_i) & (y_i - \bar{y}_i)^2 & (y_i - \bar{y}_i)(z_i - \bar{z}_i) \\ (z_i - \bar{z}_i)(x_i - \bar{x}_i) & (z_i - \bar{z}_i)(y_i - \bar{y}_i) & (z_i - \bar{z}_i)^2 \end{bmatrix} \tag{4.1}$$

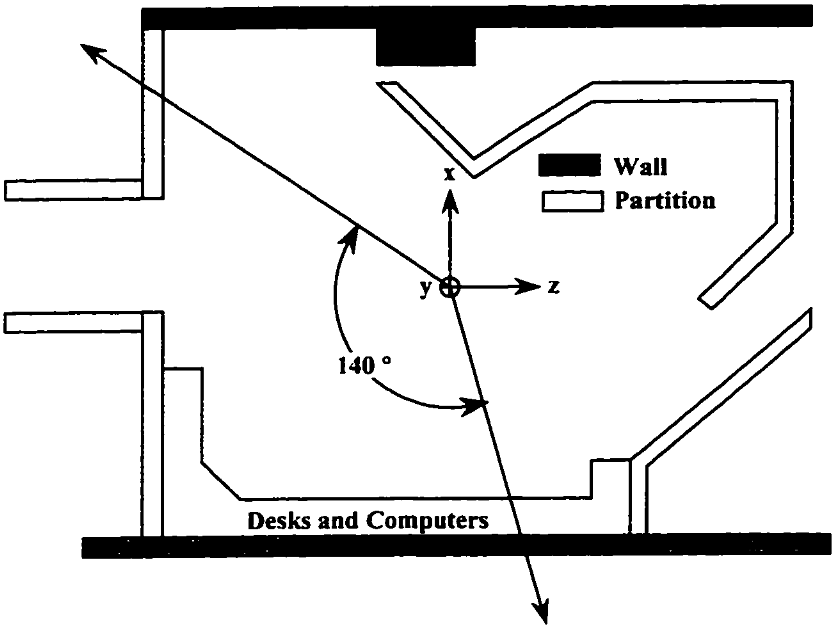


Figure 4.3: A schematic of the top view of the room where the scan in figure 4.2 was taken.

where $\bar{x}_i$, $\bar{y}_i$, and $\bar{z}_i$ are the mean values of the x, y, and z values of surface S_i.

A compatibility function for determining a quality of fit for the points to the surface is to take the mean of the squared distance of the points from the planar surface. The eigenvectors of the covariance vector are used to compute the parametric equation for the planar surfaces and the distance from a point to that surface can be computed from this equation. The result is the following compatibility function for planar surfaces from 3-D points,

$$f_{plan}(d_j) = \frac{1}{Np}\sum_j d_j^2, \tag{4.2}$$

where d_j is the distance of point j from the surface and Np are the number of points that define the surface.

The surfaces are extracted from the data using a hierarchical segmentation algorithm developed by Boulanger [14] and the results of this are presented both as an image map, see figure 4.4 where each surface has been represented by an intensity gray level value and labeled, and as a set of reconstructed 3-D points in figure 4.5.

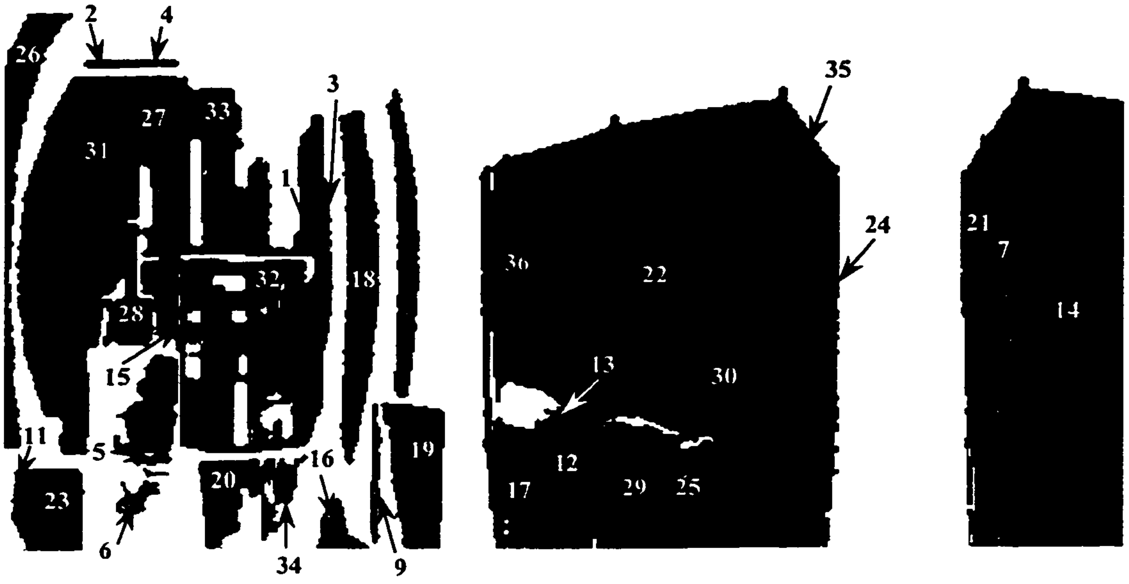


Figure 4.4: Surfaces extracted from the range data of the image in figure 4.2.

One can again clearly see that on the right there is a series of planar surfaces defining a hallway but to the left the surfaces do not really show any form of coherence. Table 4.1 shows the results of applying the planar compatibility function to the surfaces depicted in figure 4.5. The S-curve value is computed using the following arguments, $S(f_{plan}, 0, .578, 1.156)$.

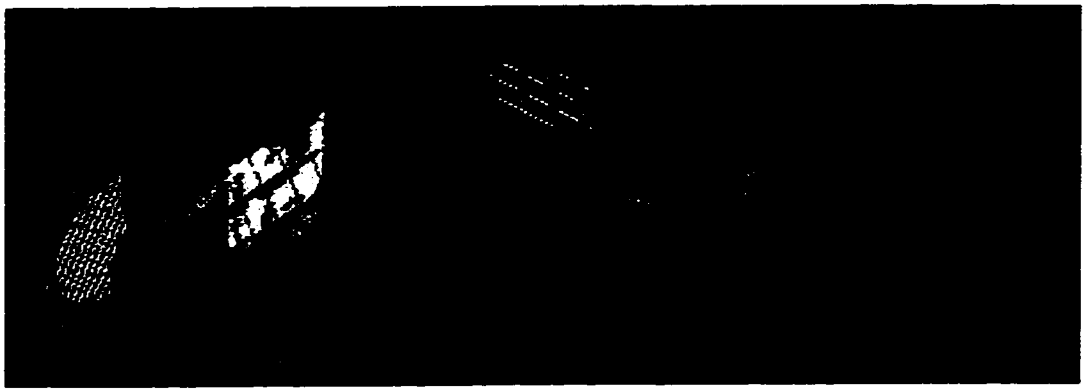


Figure 4.5: Surfaces extracted from the range data of the image in figure 4.1 shown as reconstructed 3-D points.

4.2 Surface Edge Compatibility

This compatibility function is primarily one that measures how well the edge of a planar surface defines a polygon that represents that surface. This computation can be performed using a 2-D image of a set of 3-D points since a polygon in 3-D space can easily be mapped to a 2-D surface. The advantage of this is the computations are simpler than in 3-D and the operation can be applied to an image map and therefore can also be used on intensity images.

The compatibility function is rather simple and involves a ratio of the area defined by the polygon to the area of the surface itself in the image plane and is computed using the following equation.

$$f_{edge}(S_i, Pol_{S_i}) = \frac{Area(S_i) - Area(Pol_{S_i})}{Area(S_i)}. \tag{4.3}$$

where $Area(S_i)$ computes the surface area of surface S_i based on all the points within the surface and $Area(Pol_{S_i})$ computes the surface area of surface S_i based the polygon.

S_i	f_{plan}	$S(f_{plan})$	S_i	f_{plan}	$S(f_{plan})$	S_i	f_{plan}	$S(f_{plan})$
1	0.00	1.00	13	0.42	0.74	25	0.03	1.00
2	0.27	0.89	14	0.02	1.00	26	0.10	0.98
3	0.11	0.98	15	0.06	0.99	27	0.12	0.98
4	0.16	0.96	16	0.25	0.91	28	1.00	0.04
5	0.16	0.96	17	0.03	1.00	29	0.04	1.00
6	0.62	0.42	18	0.06	1.00	30	0.05	1.00
7	0.00	1.00	19	0.17	0.96	31	0.03	1.00
8	0.08	0.99	20	0.20	0.94	32	0.28	0.88
9	0.35	0.82	21	0.01	1.00	33	0.17	0.96
10	0.06	0.99	22	0.00	1.00	34	1.80	0.00
11	0.66	0.36	23	0.07	0.99	35	0.01	1.00
12	0.21	0.93	24	0.01	1.00	36	0.01	1.00

Table 4.1: Example of using the planar compatibility function.

Pol_{S_i}. around S_i. These two values in most situations are not equivalent since the edges extracted from 3-D data tend to be irregular and noisy.

The challenge is in developing an algorithm to extract a set of straight lines from the surface's border that is a reasonable representation of a polygon for that surface. This is performed by tracking the surface's border in the image plane and at the same time computing an estimate for the curvature of the edge. This algorithm is an extension of the one developed by Gao et al. [31] for the examination of a new invariant measure for curve detection based on perceptual grouping. This algorithm uses an intensity gradient operator to determine the edge to track. This same gradient vector can be used to get an estimate of the curvature of the boundary of the surface since it is applied to one surface at a time in the image plane. The curvature along the edge is computed as a difference in a running average of the gradient along the boundary. Currently this filter uses the average of 3 pixel gradient values and appears to be able to filter the majority of large changes in the gradient values which are due primarily to the discretization of the image.

The high curvature pixel points can be used as control points in defining a polygonal surface which represents the planar surfaces extracted from the 3-D sensor values. This still results in a large number of points labeled as high-curvature points especially a great number of continuous high curvature points. In most cases boundaries with many continuous high curvature points also tend to be representative of fictitious boundaries caused by sensor degradation rather than actual surface discontinuities. With this in mind. sections of a boundary edge with many continuous high curvature points can be represented with a straight line that joins the first point of a set of continuous high curvature points to the last one in the set.

The edge extracting algorithm was applied to surface 22 from figure 4.4 and the results are depicted in figure 4.6 (a) along with the polygon that was computed from a selected set of high curvature points.

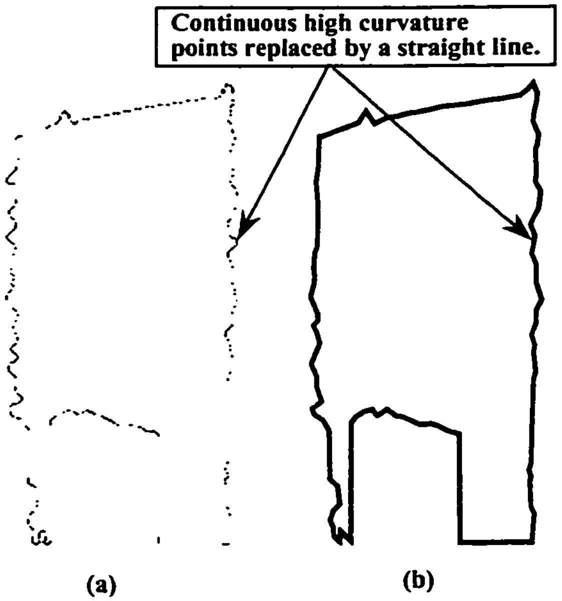

Continuous high curvature points replaced by a straight line.
(a)
(b)

Figure 4.6: Boundary of a surface from the scene depicted in figure 4.4 and its accompanying polygon.

The boundary is a chain of dark and gray pixel values. Those pixels with dark values are pixels marked as pixels with a curvature above a threshold value. The light gray pixels are boundary points with a curvature value near 0. Figure 4.6 (b) is a polygon representation for that surface. Notice how a number of extrusions on the right hand edge have been replaced with straight line segments. This is where a series of continuous high curvature points existed. This same algorithm can be applied to all the surfaces depicted in figure 4.4 so as to extract the polygonns representing the 3-D planar surfaces. The results of this operation are depicted in figure 4.7 where the polygons are formed by connecting the high curvature points (shown as white points) with straight lines (shown as light gray lines).

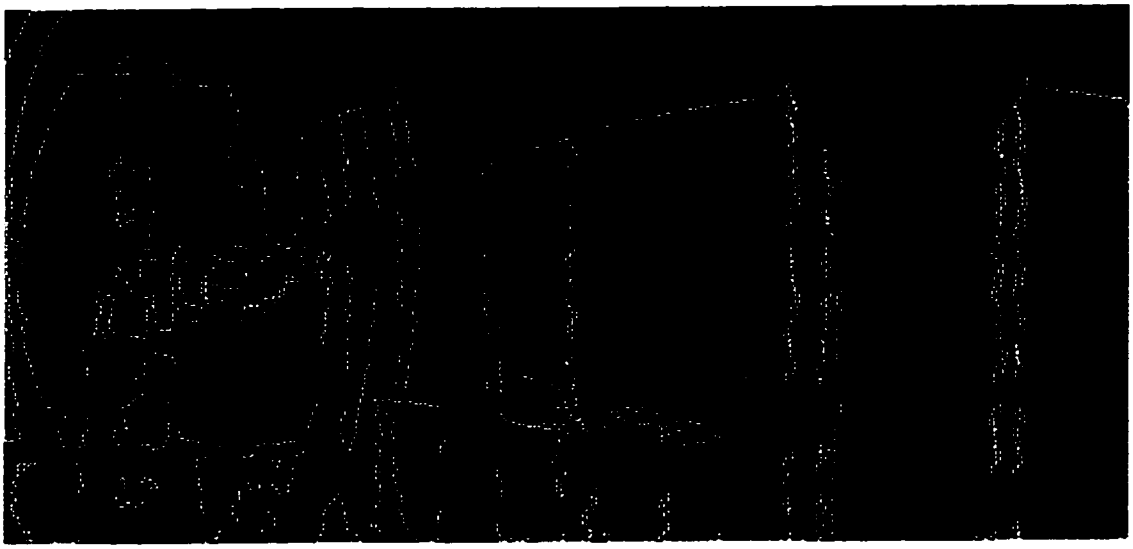


Figure 4.7: Boundary edges extracted from the surfaces depicted in figure 4.4.

Table 4.2 shows the results of applying the edge compatibility function to the polygons and surfaces depicted in figure 4.7. The S-curve value is computed using the following arguments. $S(f_{edge}, 0. 0.25, 0.50)$.

Several of the smaller surfaces have been removed from the compatibility function

S_i	f_{edge}	$S(f_{edge})$	S_i	f_{edge}	$S(f_{edge})$	S_i	f_{edge}	$S(f_{edge})$
3	0.42	0.05	16	0.25	0.49	26	0.25	0.52
4	0.37	0.14	17	0.07	0.96	27	0.16	0.79
6	0.33	0.23	18	0.17	0.78	28	0.32	0.28
7	0.20	0.68	19	0.12	0.88	29	0.10	0.93
8	0.21	0.63	20	0.05	0.98	31	0.06	0.97
9	0.41	0.05	21	0.64	0.00	32	0.16	0.80
10	0.11	0.91	22	0.04	0.99	33	0.00	1.00
11	0.19	0.71	23	0.14	0.85	34	0.27	0.42
12	0.18	0.75	24	0.21	0.66	35	0.12	0.89
13	0.36	0.16	25	0.11	0.91	36	0.08	0.95
14	0.04	0.99						

Table 4.2: Example of using the polygon compatibility function.

calculations. These are surfaces 1, 2, 5, 15, and 30.

4.3 Parallel and Coplanar Surface Compatibility

The compatibility functions for parallel surfaces and coplanar surfaces involve the comparison of two planar surfaces. A measure of parallelism between two planar surfaces can be derived by evaluating the length of the vector computed from the cross product of the normals to the surfaces. This leads to the following geometrical compatibility function,

$$f_{prll}(S_i, S_j) = |\mathbf{N}_{S_i} \times \mathbf{N}_{S_j}|, \tag{4.4}$$

where $\mathbf{N}_{S_i}$ and $\mathbf{N}_{S_j}$ are the normals corresponding to the planar surfaces S_i and S_j.

Coplanar surfaces are a more restricted case of parallel surfaces in which the two surfaces are in fact the same surface if the boundaries were removed and the surfaces extended. To determine coplanarity two conditions must be satisfied: the surfaces must be parallel to each other and the angle between the normals of the surfaces and the line joining the center point of the two surfaces is approximately 90°. The second restriction ensures that the two planar surfaces share the same common planar surface. Equation 4.4 can be used for measuring the degree of parallelism while a geometrical compatibility function based on the dot product between one of the surface normals and the vector joining the two centre points of the surfaces can be defined as an added constraint. The following equation as a measure of coplanarity was used.

$$f_{copl}(S_i, S_j) = < N_{S_i}, C_{S_j} - C_{S_i} >. \tag{4.5}$$

where C_{S_j} and C_{S_i} correspond to the location of the centers of the planar surfaces S_j and S_i respectively and the $< a, b >$ operator is a dot product operation between vectors a and b.

Table 4.3, at the end of the chapter, shows the results of applying the coplanar compatibility function to the surfaces depicted in figure 4.4. The S-curve value is computed using the following arguments, $S(f_{copl}, 0, 0.1, 0.2)$.

There are many surfaces with f_{prll} and f_{copl} equal to 0. This is due to the small value for γ that defines surfaces with normal angular deviations greater then 0.2 radians not to be parallel. In a similar fashion this limit is placed on the coplanar calculation. Results again seem consistent with a person's judgment when viewing the surfaces in figure 4.5 and determining if they are parallel or coplanar. There have been no formal tests performed to determine how well the results match human judgment.

The above surface parallel and coplanar relations rely on the detection of adjacent surfaces. This is rather simple for adjoining surfaces but relatively difficult for surfaces separated by undetected sensory readings. It is important to determine surfaces that may be possible neighbours even across gaps and for that a unique neighbour detection algorithm was developed that can also estimate among surfaces a common gap.

4.4 Adjacent Surfaces

Neighbourhood calculations among surfaces is not a simple procedure due to the roughness of surface boundaries and gaps between surfaces. Again, one can take advantage of the fact that the 3-D data is ordered and stored as an image allowing the neighbourhood computation to be performed in the 2-D image plane using the labeled image of the surfaces and their corresponding boundaries depicted in figures 4.4 and 4.7 respectively.

After the surface's boundary has been defined, adjacent surfaces can be determined by projecting from each pixel on a surface's boundary out in a direction away from the surface until it intersects with another surface's boundary. Unlike conventional region growing algorithms, this algorithm will mark the particular intersection point with another adjacent surface's boundary. This is important in estimating a common region between surfaces and will become more apparent in section 4.5 when a proximity measure is defined between two surfaces. In region growing techniques the particular neighbour associated with a pixel on a surface's edge is lost as regions are grown outward. Propagation of boundary normals are affected dramatically by noisy edges but this has been substantially minimized by using a polygon representation of the surface.

The result of using this neighbourhood determination algorithm is a larger set of possible adjacent segments than were originally computed using direct surface to surface contact. This of course is crucial for fragmented data like the type that is common in the scanning of indoor environments.

The procedure for the determination of adjacent surfaces and their respective points on the adjacent surface's boundary is depicted in figure 4.8. The points on the boundary of surface S_2 which intersect with the normal projections of surface's S_1 boundary are labeled as adjacent boundary points of S_1. The dark points are the extrema adjacent boundary points while all the other adjacent boundary points (shown in white) are in between these two points. The extrema adjacent boundary points define a segment on the boundary of surface S_2 that encompases all the adjacent boundary points. The collection of the adjacent boundary points of surface S_2 along with their respective corner points of surface S_1 can be used to compute a common gap between the two adjacent surfaces. The determination of this common gap is discussed in greater detail in section 4.5. The results of the procedure is also shown applied to surface 22 and depicted as an image in figure 4.9.

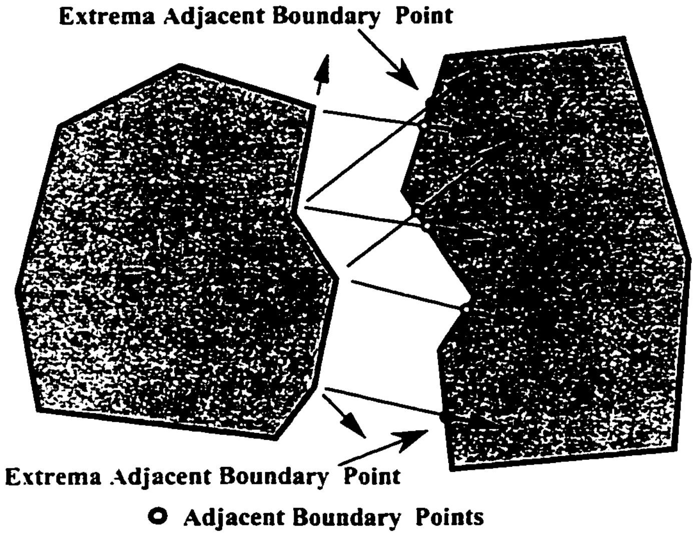

Extrema Adjacent Boundary Point
Extrema Adjacent Boundary Point
○ Adjacent Boundary Points

Figure 4.8: Adjacent boundary points of surface S_1 with surface S_2.

In this image the points corresponding to the boundary of a segment are the same gray intensity as their adjacent segment. The black points correspond to the vertices that define the polygon around the surface. The top and bottom boundary points tend to be white and signify that their immediate neighbour is the background since no other surface was detected when the boundaries were projected outwards.

Now that an algorithm has been developed for the determination of neighbouring sur-

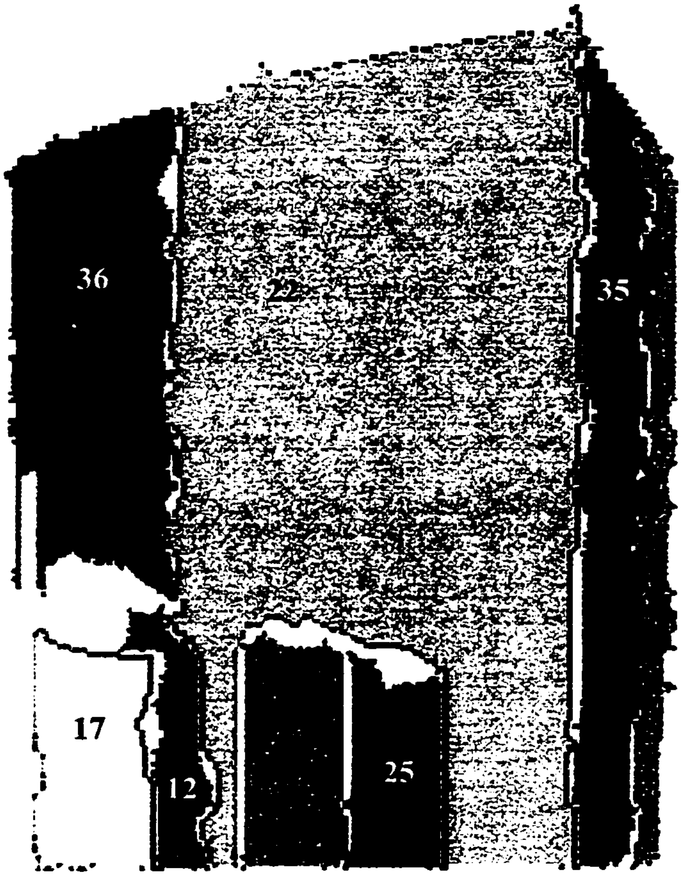


Figure 4.9: Adjacent boundary points for surface 22 and adjoining surfaces.

faces and their corresponding neighbouring boundary points these can be used to decide which surfaces are neighbours and proceed to compute a surface proximity compatibility function.

4.5 Surface Proximity Compatibility

The final step for the detection of corners and continuous coplanar surfaces is based on a proximity measure between two planar surfaces. Defining a proximity compatibility function for 3-D surfaces is challenging, because the definition of proximity is not unique for this type of data. Defining a proximity measure requires one to decide on a viewing direction to the data and on a definition of distance between the surfaces. The proposed approach defines a compatibility function among the surfaces by measuring the distance between the boundary of two surfaces. This approach is similar to Fan et al. [26] proximity measure for 2-D lines except it is applied to 3-D data. In this case the lines are replaced by surfaces and the gap by the area between the edges of the surfaces. With this in mind the following measure of proximity is proposed.

$$f_{prox}(Pol_{S_i}, Pol_{S_j}, S_i, S_j) = \frac{Area(Gap(Pol_{S_i}, Pol_{S_j}))}{Area(S_i + S_j)}. \qquad (4.6)$$

where $Gap(Pol_{S_i}, Pol_{S_j})$ computes a polygon that defines a common gap between the surfaces S_i and S_j based on their respective boundaries.

In section 4.4 an approach was presented for the determination of points between the boundaries of two neighbouring surfaces that could be used for defining a common gap among the surfaces. An example of a common area was shown in figure 4.8 as a hashed out polygon between surface S_1 and S_2. From this polygon it is possible to estimate a 3-D common surface by triangulating the surface and estimating its surface area. The approach for computing the proximity measure is primarily performed in the 2-D image space until the actual surface areas are required for computing the proximity compatibility function. The advantage of performing the computation in the 2-D parametric space is efficiency, since the shapes of the surfaces and the gap between them are polygons and computations are much simpler in a 2-D image space than 3-D.

An approximation for the shape of the gap between the surfaces is done by collecting the neighbour points and ordering them such that a polygonal surface is defined common to both surfaces S_1 and S_2. This is done in the following manner.

1. Determine the extrema points (dark circles in figure 4.8) of the neighbour points on S_2 that correspond to S_1.

2. Determine the segment of the boundary on S_2 that contains all the neighbour points including the extrema points.

3. Extract from this segment the corner points. since all that is required to define the segment are the extrema points and the corner points within the segment.

4. The gap is defined by joining this segment with the segment from the boundary of S_1 that originally defined the neighbour points from the boundary of S_2.

With this approach it is possible to get different answers depending on which surface is chosen as the originator and which as the neighbour. This is due to the fact that the normal projections are different between the two surfaces. see figure 4.10 (a) and (b) as an example.

Since for any two surfaces there exists two possible approaches in determining the gap between the surfaces. one will get two solutions for the proximity compatibility measure and these can be combined by taking an average of the size of the gap or the union of the two gaps.

Up to this point all computations were performed in the 2-D image plane while actual proximity measures must occur using the 3-D surface. To get the correct estimate for the proximity measure between the surfaces it is necessary to compute the area of the 3-D surfaces and not the 2-D image polygons. Since these particular surfaces are planar their areas can easily be computed by using any common algorithm for computing the area of a

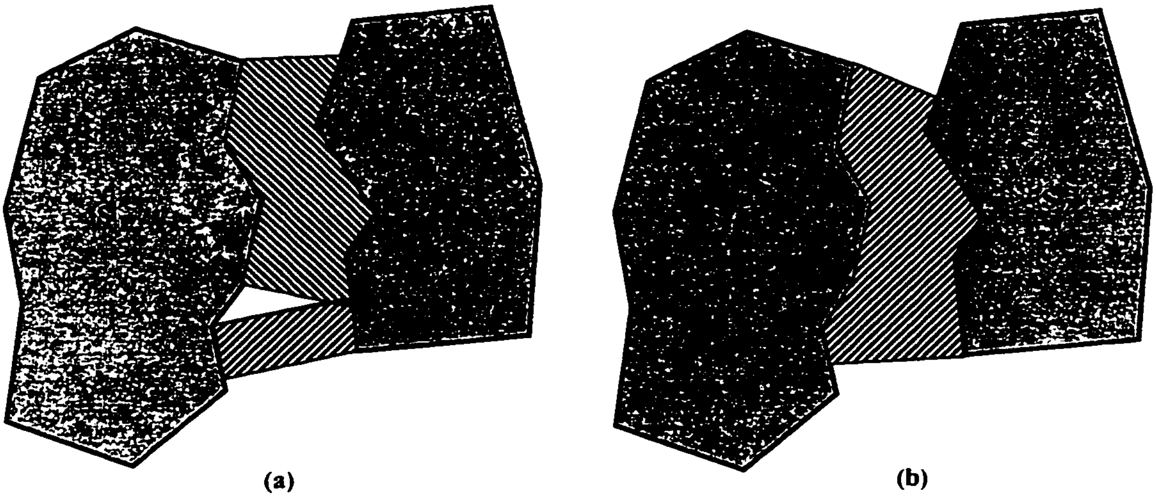


Figure 4.10: Proximity gaps between surface S_1 and surface S_2 (a) and between surface S_2 and surface S_1.

polygon in 3-D space. An estimate of the area of the gap is performed by approximating it as a triangular mesh [69] and summing the areas of the triangles. This triangular mesh approximation is necessary because the shape of the gap can be an irregular shaped surface and not planar.

An example of this is shown in figure 4.11 that shows the gap computed between surfaces $S_{17}$and S_{13}. The gap surface had to be triangulated and is shown in figure 4.11 (a) as an image and in figure 4.11 (b) as a 3-D surface along with $S_{17}$and S_{13}.

Table 4.4. at the end of the chapter. shows the results of applying the proximity compatibility function to the adjacent surfaces depicted in figure 4.4. The S-curve value is computed using the following function $S(f_{prox}, 0, 0.25, 0.5)$. and the gap size is set to the average of the two surface area results.

Several comments can be made about these results. In some situations the proximity compatibility function returns drastically different values when the originating surface

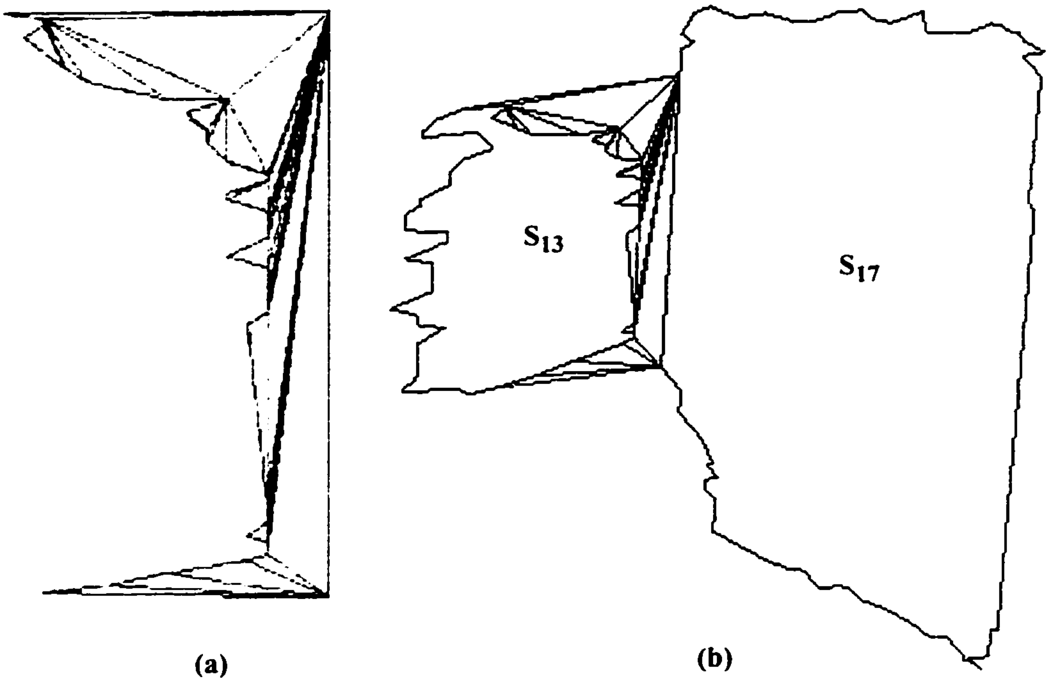


Figure 4.11: Triangulated proximity gap between surfaces S_{17} and S_{13}.

is switched with the neighbouring surface. For these situations both results have been displayed in the table. For example, surfaces 6 and 20, 10 and 31, 16 and 34, and 23 and 31 have dramatically different proximity values for the reciprocal direction. The primary reason for this difference is that the shape of the boundary between the two surfaces are very different, in particular the segment of the boundary that is considered as common to both. This is a pitfall in the algorithm but perhaps not a deterrent since it offers another clue that the surfaces more likely are not part of the same surface but two completely different surfaces. When the shape of the gap between the surfaces is nearly the same from the perspective of the two surfaces there exists a sense of symmetry as well as proximity. These two measures give a strong indication of a fragmented surface instead of two separate surfaces. For this thesis an average value for the proximity measure was used and no attempt was made to integrate any knowledge about boundary symmetry.

Note that a proximity measure was not available for all adjacent surfaces. For example surfaces 20 and 34. 22 and 35. and 22 and 36 do not have any proximity values. This has to do with the fact that the algorithm cannot determine a polygon between adjacent surfaces due to the numerous intersections that can occur when the points are so close to each other. The algorithm tries to correct for this but eventually fails if it cannot remove all the intersections. A recommendation for this is to determine this gap by directly computing the distance between two adjacent points and integrating along all the adjoining points between two surfaces. For coplanar surfaces this area will approach zero but for non coplanar surfaces the area can be fairly significant.

4.6 Summary

In this chapter. compatibility functions were presented that compute a measure of fitness in a perceptual grouping for a surface or a set of surfaces. These compatibility functions measure. the planar quality of a surface. how well the boundary represented a polygon for a surface. a measure for parallel and coplanar surfaces. and a proximity measure among surfaces. Examples were presented of these algorithms applied to an actual set of 3-D points to demonstrate their actual performance. Along with these results were computations of S-curve values that are used to compute the conditional probabilities in a Bayesian network used for grouping the surfaces. It is now possible to actually compute belief values for the formation of corners and continuous surfaces. This procedure though results in numerous unrelated Bayesian networks and an inability to specify the grouping operation in a graphical declarative manner. To be able to perform any further reasoning on the results it is better to develop a unified representation of knowledge extracted from the sensory data and uncertainty in the perception of that data. This representation is developed and presented in chapter 5 and referred to as a Bayesian attributed hypergraph.

S_i	S_j	f_{prll}	$S(f_{prll})$	f_{copl}	$S(f_{copl})$
3	33	0.26	0.00	-0.64	0.00
3	18	0.23	0.00	0.26	0.00
4	27	0.96	0.00	0.93	0.00
6	10	0.76	0.00	-0.56	0.00
6	20	0.46	0.00	0.68	0.00
6	23	0.51	0.00	-0.36	0.00
7	14	0.99	0.00	-0.60	0.00
7	21	0.01	1.00	0.00	1.00
7	35	0.05	0.89	1.00	0.00
8	22	1.00	0.00	0.99	0.00
8	36	1.00	0.00	0.97	0.00
8	18	0.10	0.47	0.07	0.73
8	33	0.17	0.05	0.33	0.00
9	19	0.07	0.76	0.02	0.98
9	16	0.79	0.00	0.98	0.00
9	34	0.91	0.00	0.52	0.00
9	18	0.76	0.00	-0.58	0.00
10	33	0.07	0.73	0.61	0.00
10	31	0.11	0.44	0.80	0.00
10	28	0.32	0.00	0.50	0.00
11	23	0.21	0.00	0.93	0.00
12	22	1.00	0.00	-0.90	0.00
12	17	1.00	0.00	0.91	0.00
13	36	1.00	0.00	-0.36	0.00
13	17	1.00	0.00	0.45	0.00
14	22	0.08	0.69	-0.08	0.72
16	34	0.42	0.00	0.41	0.00
17	36	0.06	0.84	0.54	0.00
17	19	0.73	0.00	-0.35	0.00
18	36	1.00	0.00	-0.89	0.00
18	33	0.18	0.01	-0.35	0.00
20	33	0.45	0.00	-0.71	0.00
20	34	0.98	0.00	-0.63	0.00
20	23	0.12	0.31	-0.50	0.00
21	24	0.04	0.93	1.00	0.00
21	35	0.05	0.85	0.84	0.00
22	35	1.00	0.00	0.47	0.00
22	25	0.98	0.00	-0.74	0.00
22	29	0.24	0.00	-0.92	0.00
22	36	0.02	0.98	0.73	0.00
23	31	0.24	0.00	-0.87	0.00
24	35	0.02	0.99	-0.01	1.00
25	29	1.00	0.00	-0.65	0.00
26	31	0.06	0.85	-0.07	0.73
27	33	0.22	0.00	0.37	0.00
27	32	0.24	0.00	0.02	0.98
27	28	0.19	0.01	0.92	0.00
27	31	0.07	0.74	-0.18	0.02
28	31	0.22	0.00	-0.82	0.00
32	33	0.45	0.00	-0.95	0.00
33	34	0.97	0.00	0.23	0.00

Table 4.3: Example of using the parallel and coplanar compatibility functions.

S_i	S_j	f_{prox}	$S(f_{prox})$	S_i	S_j	f_{prox}	$S(f_{prox})$
3	33	0.02	1.00	13	17	0.04	0.99
3	18	0.01	1.00	14	22	0.15	0.83
4	27	0.01	1.00	16	34	0.36	0.15
6	10	0.06	0.98	34	16	0.01	0.99
6	20	0.01	0.99	17	36	0.15	0.81
20	6	6.36	0.00	17	19	0.09	0.93
6	23	0.07	0.97	18	36	0.15	0.83
7	14	0.02	1.00	18	33	0.02	1.00
7	21	0.01	1.00	20	33	0.16	0.80
7	35	0.02	1.00	20	23	0.01	1.00
8	22	0.08	0.95	21	24	2.10	0.00
8	36	0.14	0.85	21	35	0.02	1.00
8	18	0.19	0.70	22	25	0.38	0.12
8	33	0.05	0.98	22	29	0.40	0.07
9	19	0.03	0.99	23	31	0.52	0.00
9	16	8.53	0.00	31	23	0.01	0.99
9	34	0.04	0.98	24	35	0.02	1.00
9	18	0.02	1.00	25	29	0.04	1.00
10	33	0.17	0.77	26	31	0.02	1.00
10	31	0.33	0.24	27	33	0.07	0.96
31	10	0.01	1.00	27	32	0.01	1.00
10	28	0.15	0.81	27	31	0.02	1.00
11	23	0.18	0.74	27	28	0.08	0.95
12	22	0.01	1.00	28	31	0.01	1.00
12	17	0.01	1.00	32	33	0.17	0.76
13	36	0.05	0.98	33	34	0.02	0.99

Table 4.4: Example of using the proximity compatibility function.

Chapter 5

Bayesian Attributed Hypergraphs

It is necessary to define a suitable representation for maintaining the features extracted from the sensory data and for the models hypothesized from those feature sets. Along with the hypothesized models is the need of a belief value in the formation of those models. Previous chapters, in particular chapter 3, have dealt primarily with the procedural knowledge required to compute belief values in the formation of groupings from the sensory data. This chapter introduces a representation, called the Bayesian Attributed Hypergraph that is a unified representation for declarative knowledge and the knowledge required for the updating of belief values using Bayesian networks. It is a generic representation that is not influenced by the limitations that are imposed by the use of compatibility functions but can be used by different approaches.

A BAHG maintains one representation for the grouping of sensory features but may contain several instantiations of Bayesian networks representing the formation of higher order groupings. This is required when analyzing computer images for the formation of perceptual groupings from feature sets due to the combinatorial checking among those feature sets. If a unified representation is not maintained several separate instantiations of the perceptual groupings occur and it becomes an unmanageable task to determine which of these are the same grouping. This will be demonstrated in more detail in section 5.3. Another advantage of a BAHG is that conventional graph operations can be performed on the BAHG that allow for the grouping of several feature sets where the number of feature sets is not known priori. This is not easily specified with conventional BN development environments where the number of feature sets must be explicitly mentioned. Fortunately this limitation is not a limitation imposed by the Bayesian network itself but primarily a problem with the creation and instantiation of the network. It is important to still allow the user to explicitly specify multiple feature groupings in some language and use the compatibility functions to create groupings from multiple feature sets.

BAHGs also facilitate decisions that must be taken after the sensory data has been grouped. For example, in the domain of environment modeling, sensory data is grouped to hypothesize a formation not readily detected by the sensor. After using a Bayesian network to compute the belief value in that formation it is necessary to act upon that knowledge and construct a model using the hypothesized formation. This is a separate procedure than that of maintaining belief values in the perceptual groupings. A unified representation, like the BAHG, facilitates this process by acting as the common repository of declarative knowledge.

This chapter will present a definition for the BAHG and an approach to the creation of a BAHG for perceptual grouping that uses a graphical network interface and compatibility functions.

5.1 Bayesian Attributed Hypergraph (BAHG)

Bayesian Attributed Hypergraphs are an extension of Attributed Hypergraphs (AHGs) which have been used successfully for object recognition in the early 1990s [81, 26, 80, 52, 82]. Most of this previous work had focussed on the construction and definition of

AHGs from existing Computer Aided Design (CAD) models and the comparison of AHG's derived from models to those extracted from sensory data. All of these approaches use rule-based heuristics as the knowledge that guides the pattern matching process. Rule-based systems are difficult to maintain and develop because the dependencies among the variables are not easy to model and do not offer a declarative manner to express the often hierarchical structure seen in perceptual grouping and object recognition. In most circumstances decisions on the goodness of the sensory data were done at an early stage in the process and therefore evidence cannot be accumulated throughout the object recognition procedure.

BAHGs on the other hand maintain belief values in the formation of hyperedges and represent instantiations of Bayesian networks that are used to update these belief values. Because a Bayesian network approach is taken, the dependencies among the hyperedges are specified using a BAHG network which is similar to a Bayesian network but allows for the specification of construction functions for creating the hyperedges, and PDF functions for computing the conditional probabilities. The BAHG is an instantiation of a type of hyperedge so it is necessary to present a number of formal definitions for attributed graphs and hypergraphs.

Definition 3 *An* **attribute pair** *is an ordered pair* (A_n, A_d) *where* A_n *is an attribute name and* A_d *is the attribute value.*

Definition 4 *An* **attribute set** *is an m-tuple* $\{p_1, p_2, \cdots, p_m\}$ *where each element is an attribute pair. The attribute set of a surface, for example, could be* $S = [(type, planar), (parameters, (n_1, n_2, n_3)), (color, red)]$.

Definition 5 *An* **attributed vertex** *is a vertex associated with a vertex attribute set. An attributed edge is an edge associated with an edge attribute set.*

Definition 6 *An* **attributed graph** *is a graph $G_a = (V_a, A_a)$ where $V_a = \{v_1, \cdots, v_p, \cdots, v_q, \cdots, v_m\}$ is a set of attributed vertices and $A_a = \{\cdots, a_{pq}, \cdots\}$ is a set of attributed edges. The edge a_{pq} connects vertices v_p and v_q. If the edges are directed, we call them arcs.*

Attributed graphs have been used extensively for representing attributes and relations among geometric features either for sensory perception or computer graphics. The fundamental features used as examples in this thesis, i.e. planar surfaces and their respective boundaries, have been originally represented as attributed graphs. A simple example is the line drawing of a room depicted in figure 5.1 (a). It is a typical drawing of a room that the human has little difficulty in perceiving the shape and structure of the room. The processing required on this drawing is simpler than is required for actual sensory data since there is no noise in the image and the detection of junctions among the edges is relatively easy. Figure 5.1 (b) is the attributed graph representation for this room that depicts the lines and junctions among those lines.

This attributed graph is a direct representation of what is depicted in the image, i.e. the lines and the types of junctions among the lines. It is fairly limited as one can tell by the fact that it cannot represent the actual lines that form a particular type of junction but simply represents all the lines that connect to the line. For example line l_6 has 4 connections with other lines. Three of these are "T-junctions" while the other is a "Y-junction". The attributed graph can differentiate between these connections but only among any 2 set of lines. Therefore, the "T-junctions" are properly represented with one attributed edge between the two lines forming the "T-junction", but the "Y-junction" is formed between 3 lines and requires two edges to represent this connection. The result is that the node representing l_6 will contain 5 edges instead of 4 representing the particular type of connections.

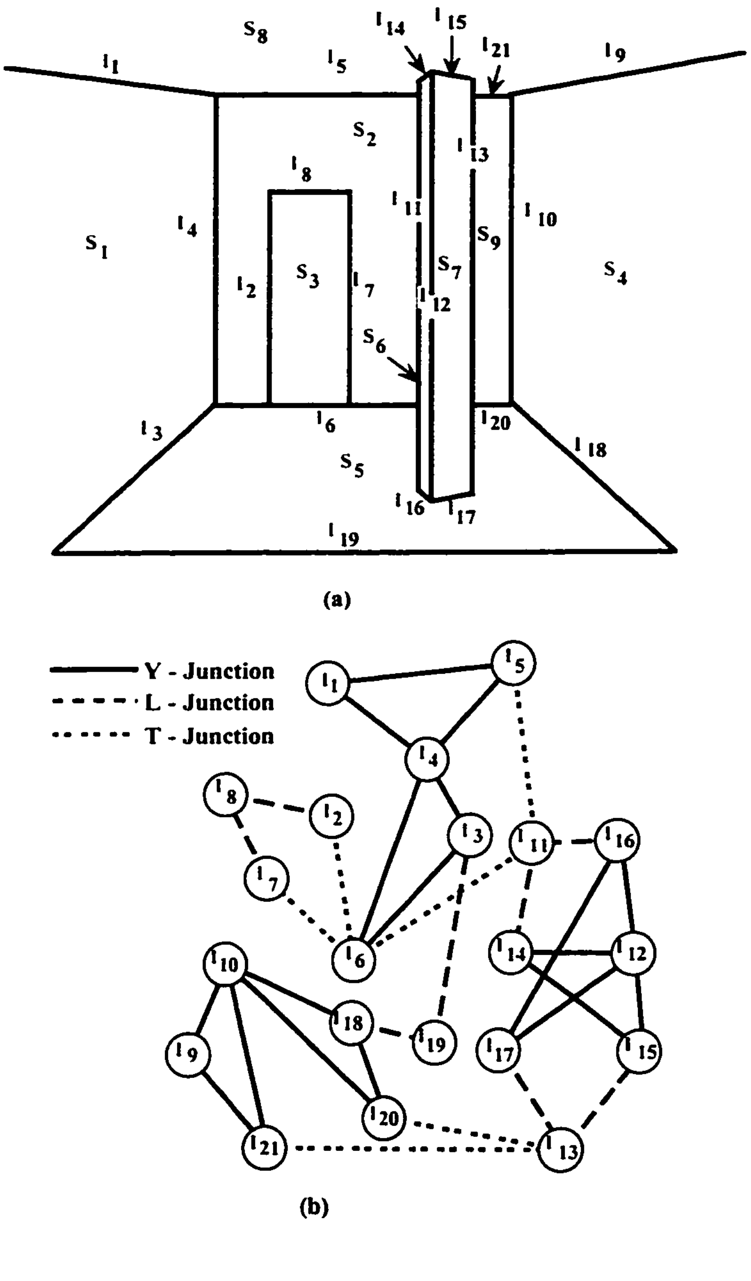


Figure 5.1: Line drawing of a room (a) and its attributed graph representation (b).

Hypergraphs on the other hand are a representation in which an arbitrary number of features extracted from the sensory data can be grouped based on a compatibility function. So for the above example using line l_6, all that is required are 4 hyperedges that can represent the particular junctions. Hypergraphs are ideal for representing the perceptual grouping of feature sets since the hypothesized formations are easily represented as hyperedges. Hypergraphs and attributed hypergraphs are defined in the following manner.

Definition 7 *A* **hypergraph** *is defined as an ordered pair* $H = (V, E)$ *where* $V = \{ v_1, \cdots, v_n \}$ *are vertices and* $E = \{ e_1, \cdots, e_m \}$ *are the hyperedges which are subsets of* X *such that* $e_i \neq \emptyset$, $i = 1, \cdots, m$ *and* $\bigcup_{i=1}^{m} e_i = X$.

Definition 8 *An* **attributed hypergraph (AHG)** $H_a = (V_a, E_a)$ *consists of a set of attributed vertices* V_a *and a set of attributed hyperedges* E_a. *The attributes of the vertices are similar to those of attributed graphs. The attributes of a hyperedge are mappings from the set of vertices in the hyperedge into a range which provides a description of the set. The range can be (1) an attributed graph defined on the vertex set, (2) a set of nominal or ordinal values, (3) a geometrical configuration of that set, or (4) a set of hyperedges.*

Using the data in the attributed graph depicted in figure 5.1 (b) it is possible to perceive certain formations from the image and represent them as hypergraphs. This is demonstrated in figure 5.2 where several surfaces have been perceived from the lines and their respective junctions. For a line image one can use the approach developed by Wong et al. [83] based on the concept of trying to extract closed polygons from a set of edges. The hyperedges can represent the actual edges that form the surfaces while the attributed graph cannot.

Even with this simple example of a line drawing, that is free of noise imperfections, a great deal of perception is occuring and ambiguity of the definition of a surface still

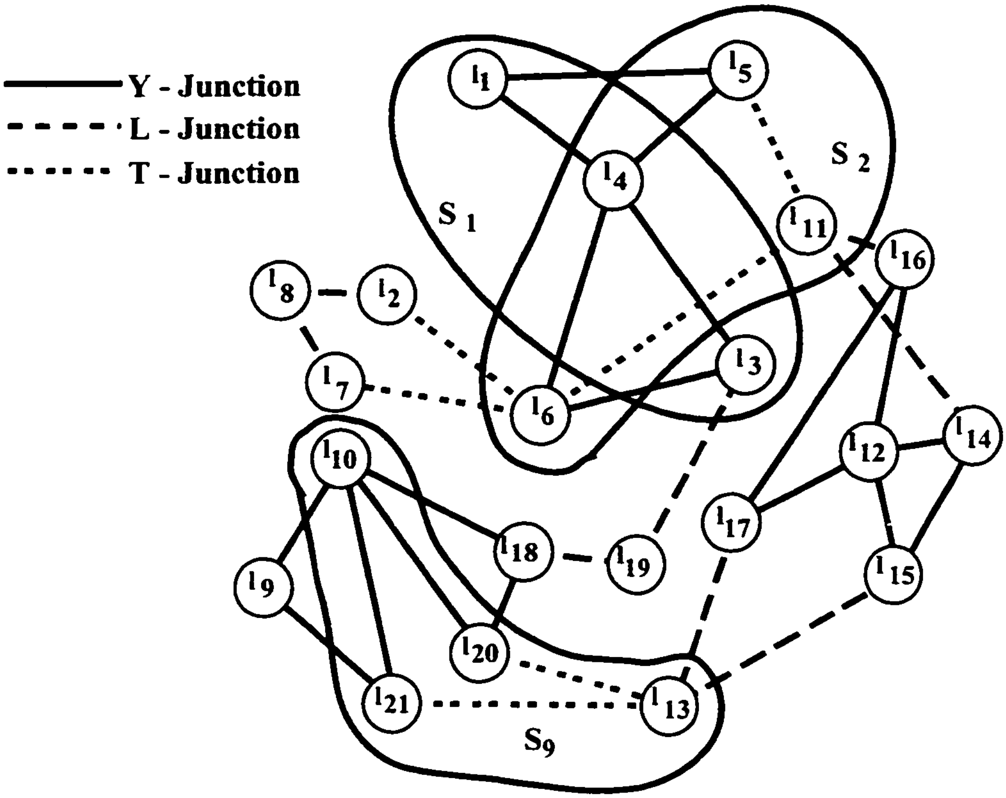


Figure 5.2: An attributed hypergraph showing surface formations hypothesized from the attributed graph in figure 5.1.

exists. For example, from the lines defined in figure 5.1 it is not clear how surface S_2 is defined. Using the lines l_5.l_4.and l_6 is ambiguous because it encompasses S_3. on the other hand if lines l_2. l_8. and l_7 are incorporated into the definition then only portions of line l_6 define S_2. These examples just emphasize the difficulties associated with interpreting even clean line drawings.

Hypergraphs can also be presented as hierarchical networks since the nodes that are part of the hyperedge can be considered as the children of the hyperedge. The children can be other hyperedges and therefore it is possible to create hierarchical directed networks similar to those used for Bayesian networks and hierarchical decomposition.

AHGs are very generic representation that are very powerful for object recognition and perceptual grouping but do not have a mechanism for maintaining beliefs in the formation of groupings among feature sets. Bayesian networks are a natural fit with AHGs because they can be considered as hypergraphs [61] instead of simple directed graphs. Therefore. it is natural to combine the strengths of these approaches. Bayesian networks are designed as a mechanism for reasoning with uncertainty. They do not try to maintain any other type of declarative knowledge or describe an approach for the management of multiple instantiations of Bayesian networks required for many image analysis problems. A solution for this is to formulate an AHG that can maintain levels of belief in the formation of the hyperedges and from the AHG be able to formulate Bayesian networks to maintain the belief values. This differs from creating a table that points to several Bayesian networks and instead tries to approach the problem from an attributed hypergraph perspective. With this in mind the following is a definition for a Bayesian attributed hypergraph.

Definition 9 *A* **Bayesian Attributed Hypergraph** $BAHG_a = (V_a. E_a^i)$ *is an AHG where the vertices* V_a *are hyperedges and all the hyperedges have associated with them a belief value. The hyperedges must contain the following attributes:* **state.bel^i**. **pdf^i**.*and* **basis^i**. *Attribute* **state** *defines the states the hyperedge represents. Attribute* **bel^i** *maintains a belief in the states of the hyperedge. Attribute* **pdf^i** *is a set of conditional probabilities conditioned on the* **basis** *hyperedges. Attribute* **basis^i** *is a list of hyperedges that represent the parents of* E_a^i. *The index* i *allows for multiple instantiations of Bayesian networks.*

One of the strengths of a BAHG is the ability to maintain multiple instantiations of Bayesian networks while only requiring single instantiations of the hyperedges. For each Bayesian hyperedge there may be several belief, conditional probabilities, and basis that pertain to an instance of a particular Bayesian network. Figure 5.3 shows a BAHG representation for the formations of corners from the surfaces depicted in figure 5.2.

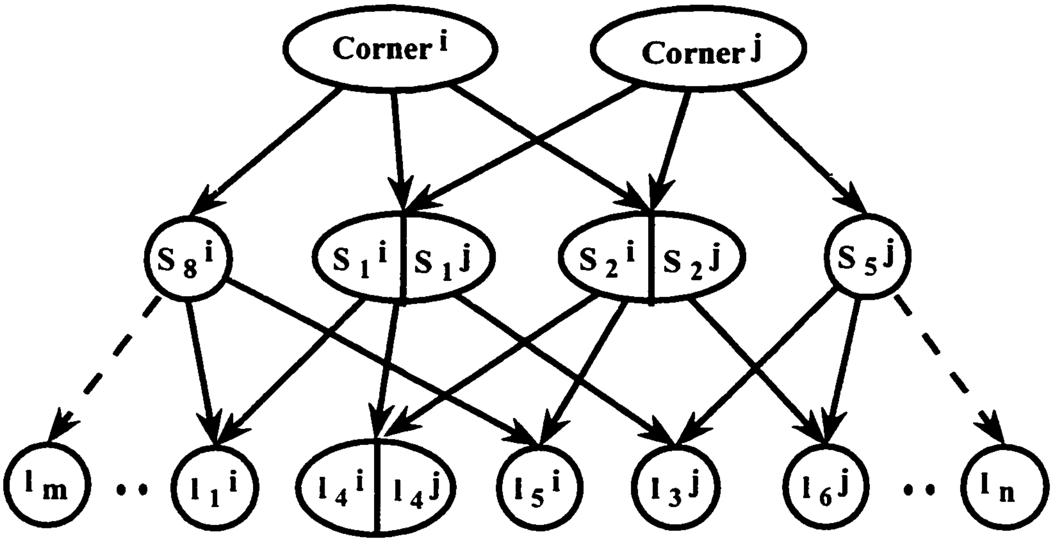


Figure 5.3: A Bayesian attributed hypergraph showing the formation of surfaces and coplanar surfaces represented in the AHG in figure 5.2.

The BAHG is shown now as a hierarchical directed graph because it is not possible to represent hierarchical structures using the freehand drawn bubbles in figure 5.2. The elements that form the hyperedge are depicted using directed edges from the parent hyperedge to the children elements. In a BAHG all nodes are considered hyperedges so the elements that form the hyperedge are themselves hyperedges. In this particular BAHG two corner formations are depicted: *Corner*^{i} and *Corner*^{j} that are formed from surfaces S_1, S_2, and S_8 and S_1, S_2, and S_5 respectively. This particular BAHG also is indexed with i and j to represent two distinct Bayesian network paths and two corner instantiations. Section 5.3 gives a more detailed justification and presentation of the maintenance of multiple Bayesian networks using a BAHG. For now it suffices to simply show this as a split node.

BAHGs are generic structures that do not constrain the direction of computation of the belief values of the nodes. They simply store relations. belief values and conditional probabilities that allow Bayes theory to be applied in order to compute the belief values in the hyperedges. How and where these values are obtained are not part of the definition of the BAHG and can be specified or computed using compatibility functions. When BAHGs are applied to the perceptual grouping process using compatibility functions the direction of computation is important and the arrows of the BAHG shown in figure 5.3 need to be inverted.

One of the unique differences between this network and a typical Bayesian network are the instantiation of multiple sub-Bayesian networks within the BAHG. This can simplify the computation of conditional probabilities significantly and also allows for the specification of a generic BAHG network. see section 5.4. that can be used to construct the BAHG recursively.

5.2 Reasoning using the BAHG

The simplest form of reasoning that can be performed using the BAHG is to search for connections among the nodes in the BAHG. This is usefull when the BAHG is being created and when it is used for further reasoning after the belief values have been computed for each hypothesis. Because the BAHG is a particular instantiation of an AHG then it inherits the features common to all AHGs as well as the ability of using any graph matching techniques on the BAHG. For example. connectivity among surfaces is important during the process of detecting corner formations among the surfaces. Figure 5.4 is part of the AHG depicted in figure 5.2 showing the formation of surfaces S_1. S_2. and S_8

from the lines.

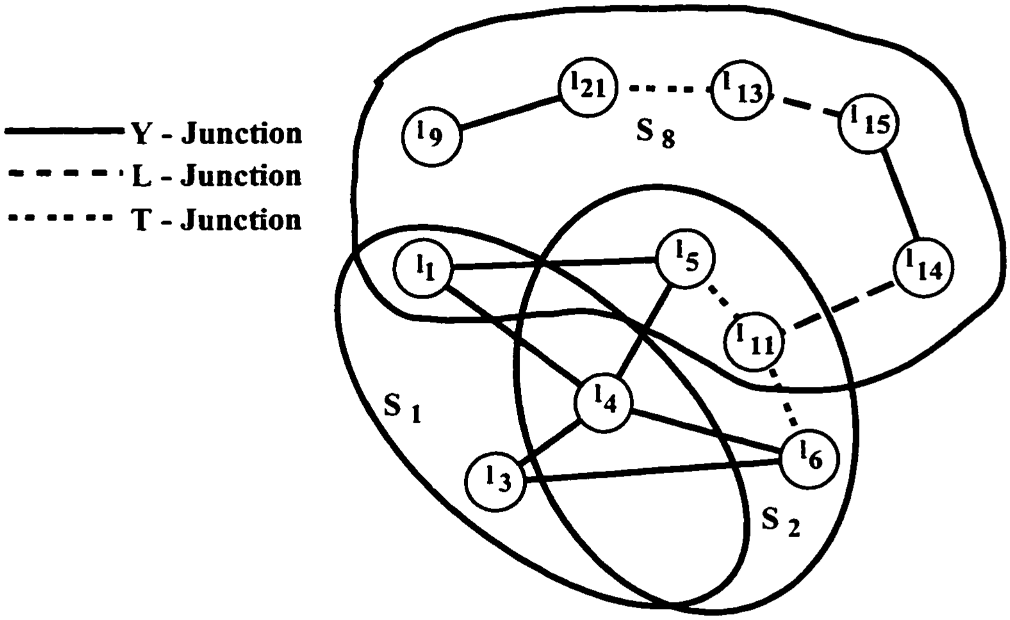


Figure 5.4: An attributed hypergraph showing surface formations for surfaces S_1, S_2, and S_8.

This is one of the first steps in the formation of the BAHG and figure 5.4 can be considered as the first layer in the BAHG depicted in figure 5.3. The knowledge required to make the corner hypothesis is located in the BAHG by searching for surfaces with common lines that form a "Y-junction". In this example the lines l_1, l_5, and l_4 are shared among two surfaces and form a "Y-junction". Since the Bayesian knowledge sources are treated as attributes of a hyperedge then the knowledge required to form the Bayesian networks as well as the networks themselves are maintained in the BAHG.

Another example is in the use of the BAHG for performing some particular action after belief values have been computed for the hyperedges. This type of action is either planning a path among the surfaces or trying to decide which surfaces should be connected so that a model can be created from the sensory data. Later, for the actual experiments using 3-D range data in chapter 6 several examples are presented that use the BAHG to decide which surfaces are to be connected.

5.3 Maintaining Multiple Bayesian Network Instantiations

Most cause effect relationships in a Bayesian network can be represented as a simple divergent node as that depicted in figure 5.5 (a) that models the hypothesis H and a number of feature sets, $Fs_i \ldots Fs_k$ that are a result of the hypothesis.

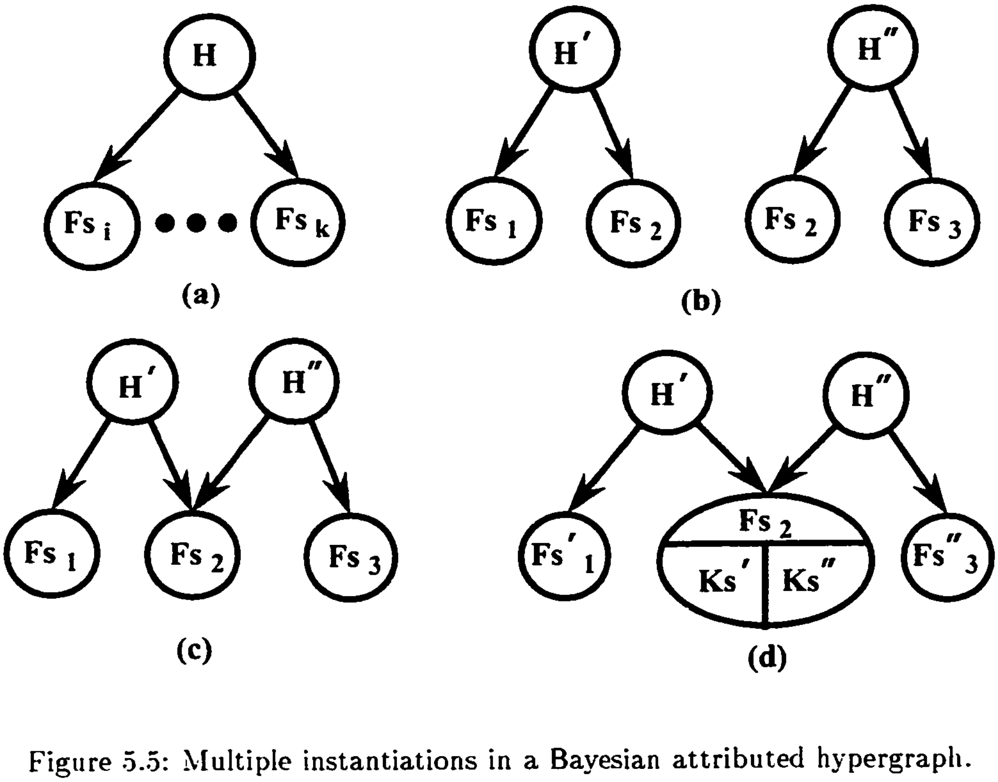


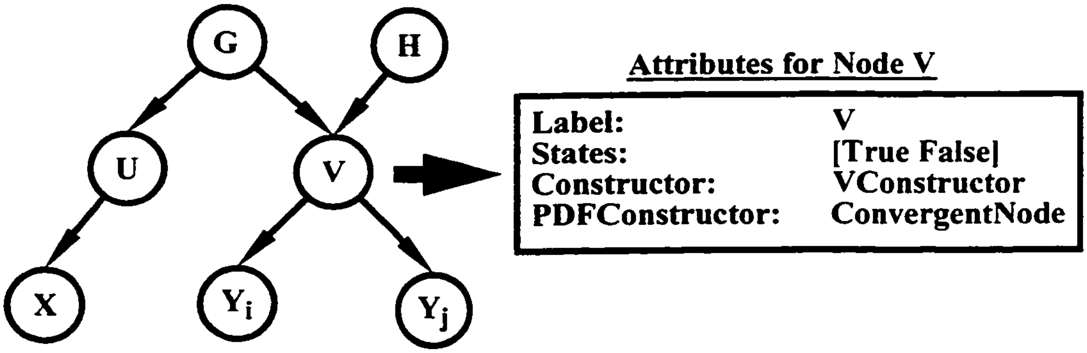
Figure 5.5: Multiple instantiations in a Bayesian attributed hypergraph.

When this network is applied to a number of feature sets, for example in an image, then a multiple number of networks are instantiated. An example of this is depicted in figure 5.5 (b) where two particular networks have been created for the hypothesis H' and H'' from the feature sets Fs_1 and Fs_2. and Fs_2 and Fs_3 respectively. The drawback of this is that nodes are duplicated and no relationship exists between each individual network. For problems where feature sets are combinatorial. like in image analysis. it would be beneficial to maintain one representation for a particular feature set. When the networks are combined the conditional probabilities are affected and the network is no longer the same as that originally depicted. For example. figure 5.5 (c) is the combined network of H' and H'' that use one instantiation for the feature set Fs_2. This results in conditional probabilities for Fs_2 conditioned upon both H' and H'' while previously Fs_2 would be conditioned solely on H' or H''. In many circumstances there exists an approach to compute $P(Fs_i|H)$ but none would exist for multiple instantiations of H. Figure 5.5 (c) implies that the feature set Fs_2 is dependent on both H' and H''. This may be a true statement but requires an approach to dynamically compute the conditional probabilities based on the number of instantiated hypotheses. It is then difficult to use a declarative approach to specifying the network. The use of compatibility functions and those procedures outlined in section 3.3 can be used for computing these dynamic conditional probabilities. but a simpler representation was chosen that maintains an independent relationship among each instantiated network using a single representation for the feature set. This new structure is represented in figure 5.5 (d) using the new feature set node Fs_2 that has two sets of knowledge sources for maintaining separate conditionals and belief values in the knowledge sources Ks' and Ks''. The individual Bayesian networks are indexed. in this case with single and double quotes. so as to differentiate them. The advantage of this is a simpler computation for beliefs but unfortunately multiple belief values depending on the Bayesian network. Fortunately it is primarily the feature sets that have the multiple instantiations and not the eventual hypotheses. This condition was also depicted in the example of the BAHG in figure 5.3 that detected the occurrence of corners from the line drawing of the room in figure 5.1 (a).

5.4 Constructing a BAHG for Perceptual Grouping

A general procedure for the formation of an AHG from CAD models was developed by Weichung et al. [80] and is similar to the construction by parts methodology used in CAD environments. This approach has been extended to be used for the construction of BAHGs for perceptual grouping. This procedure has two components, a BAHG network which is similar to a Bayesian network but contains pointers to construction and conditional functions and a BAHG constructor that reads the BAHG network and creates the BAHG from that information.

An example of a BAHG network is depicted in figure 5.6 for a general type of network. This BAHG network shows the information required in a node as an expanded window to the right of node V. Here the name of the node, the states, a constructor function and a function for computing the conditionals are specified.



Attributes for Node V

Label:	**V**
States:	**[True False]**
Constructor:	**VConstructor**
PDFConstructor:	**ConvergentNode**

Figure 5.6: Example of a BAHG network and its associated attributes.

The *Label* for the Node is used as a base name for labeling the BAHG hyperedges.

Each instantiation of this particular hyperedge adds a numeric value to the base name. i.e. $V_1, V_2, \ldots V_n$. The hyperedge constructor is *VConstructor* which will be used to determine what hyperedges are collected from the BAHG to form this particular hyperedge. This information is partially available from the network by looking at the children of a particular hyperedge except that the constructor function can control this procedure. This construction is generally performed in a bottom up direction from detected feature sets to hypothesized formations. Fortunately when using compatibility functions this in fact is the preferred method for computing the belief values so that it simplifies the computation required for the conditional probabilities. The PDF constructor in this example is a generic PDF constructor named *ConvergentNode* that uses the compatibility functions to determine the conditional probabilities using the approach described for convergent nodes in section 3.3. The other generic PDF constructor is one for nodes with one parent. A feature in the BAHG constructor that is useful is the subscripted indices that signify the grouping of similar feature sets. This is a very common occurrence in perceptual grouping where several similar features are grouped according to a compatibility function.

The formation of a BAHG is performed by a process known as a *BAHG Constructor* and depicted in figure 5.7 at the end of the chapter. It reads the BAHG network and uses that declarative knowledge and relations among the nodes to construct the BAHG representation.

The BAHG Constructor is a process that uses the information in the BAHG network to perform two principal functions: create the hyperedges of the BAHG along with their respective conditional probabilities and then interfaces with any conventional Bayesian network package to compute the belief values in the formation of the hyperedges.

This particular BAHG illustrates several possible situations that can arise in the construction of the BAHG. Two instantiations of a BN have been created: indexed using the superscript m and n. The BAHG maintains separate conditional probabilities and belief values for these instantiations. This BAHG represents 4 hypothesized perceptual groupings from the sensory data, these are represented by the roots of the BAHG and are G^m, G^n, H^m, and H^n. The perceptual groupings G^m and H^m caused the fundamental feature sets labeled as X^m, Y_i^m, and Y_j^m while the perceptual groupings G^n and H^n caused the feature sets labeled as X^n, Y_i^n, and Y_k^n. The BAHG maintains only one instantiation of a hyperedge of a particular perceptual grouping. The subscripted labels signify that two different instantiations of that hyperedge are to be considered in the creation of the parent hyperedge. For example, the nodes Y_i and Y_j specify that two different instantiations of the same hyperedge Y are to be considered as possible candidates in the formation of the hypothesis V. This is not unusual in pattern recognition where similar feature sets are often combined to form groupings.

At this point it is necessary to consider the direction of the network for computing the conditional probabilities. Is the network depicting a causal network or a consequence network? This is important because the network for perceptual grouping is formed from evidence to hypotheses and depending on the type of network the construction process differs slightly. The following procedure deals with the construction of a BAHG evaluated in the hypotheses direction, bottom-up, since the use of compatibility functions requires this type of computation.

- Construct an AG of the elementary features extracted from the sensor. These can be intensity edges for 2-D data and surfaces for 3-D data. The edges in the AG represent an adjacency relation among the elementary features.

- Create the hyperedges that represent the features extracted from the sensory data. These nodes will be instantiated as TRUE when evaluating the network. These hyperedges are special in that they do not require any construction procedure but simply a transition from the attributed graph nodes to a hyperedge so as to maintain consistency among the hyperedges of the BAHG. If the network is a consequence network then set the conditional probabilities for these nodes to (0.5.0.5) to represent an unbiased prior probability.

- Commencing with one of the existing hyperedges. construct the parents for this particular hyperedge using the *Constructor* function and making certain that each parent has the same value for an index. Note that each of the construction procedures uses a compatibility function to determine a quality of fit of the components to the hypotheses.
- After the parents have been constructed execute the *PDFConstructor* function to compute the hyperedge's conditional probabilities. These *PDFConstructor* functions use the procedures developed in section 3 to compute the conditional probabilities from the results of the compatibility functions.
- Repeat this process for all the nodes required to be constructed in the BAHG.

The construction process for a BAHG that is based on a causal network is similar except that the conditional probabilities cannot be computed at the same time as the hyperedge gets created. This is because the compatibility functions rely on the construction of the parent nodes and in a causal network the children are constructed before the parents. while in a consequence network the reverse condition persists.

After the BAHG has been constructed the BAHG Constructor goes through the BAHG computing the belief values in the existence of the hyperedges based on any conventional Bayesian network algorithm. Currently the JavaBayes [21] package is used and it is based on a generai algorithm for probabilistic inference known as variable elimination adapted from Zhang and Poole [84].

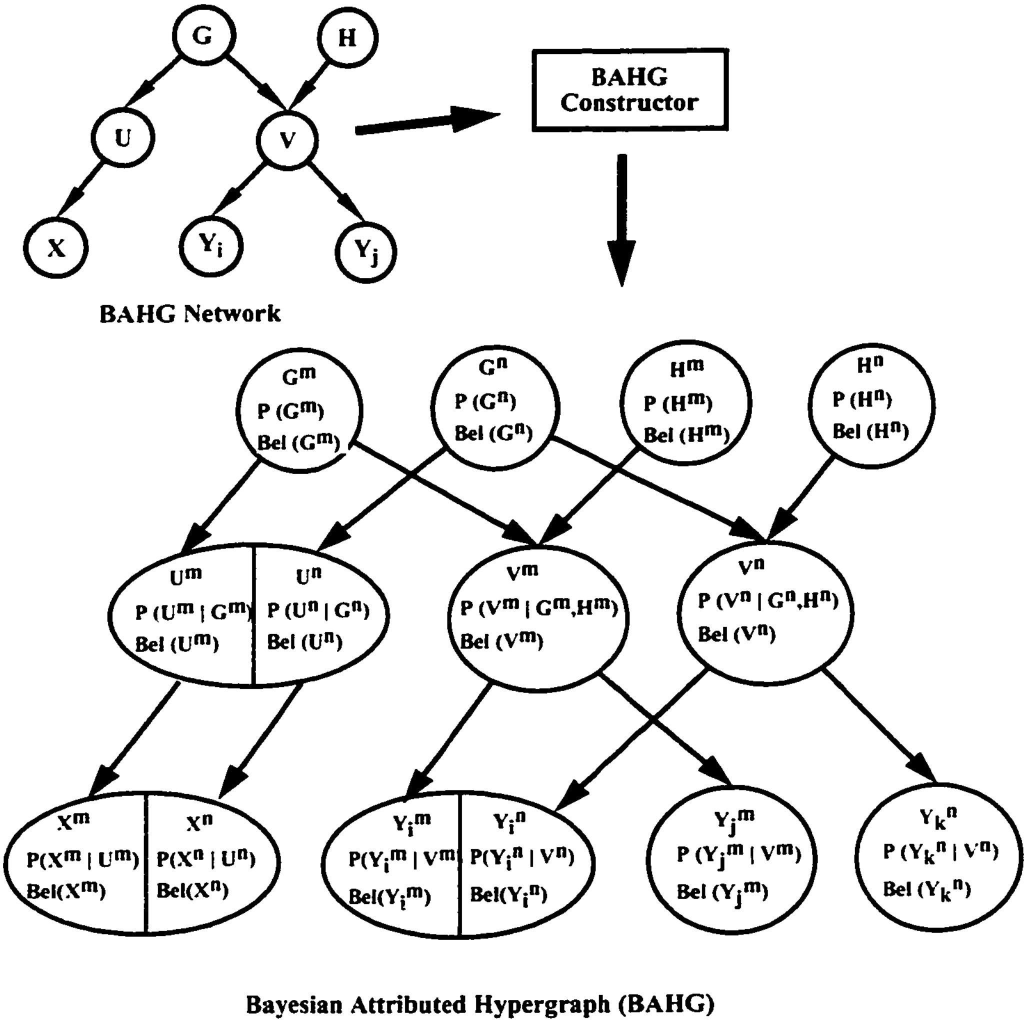
Another advantage of using the unified BAHG representation is that it allows the specification for the grouping of multiple features without having to know apriori the number of features to group. This can be done by taking advantage of the connected nodes in the BAHG. It is possible to specify the grouping of a set of features that are connected in the BAHG using a compatibility function to measure the quality of the grouping and then compute a conditional probability for that grouping. This is done by encoding the grouping procedure into the *Constructor* function and using the knowledge of adjacency declared in the BAHG to guide the grouping process.

5.5 Summary

The BAHG network is an interface to the user for specifying the construction of the BAHG. It offers a mechanism for specifying the construction and PDF functions that represent the procedural knowledge required to construct the BAHG hyperedges with their respective conditional probabilities.

The BAHG is a unified representation of attributed hypergraphs and Bayesian networks. The desire to do this comes from the need to integrate declarative knowledge into one representation along with the belief in the formation of hypotheses infered from the data. Even though the BAHG is not application dependent it is better suited for domain applications where a Bayesian network has to be applied recursively across a set of ordered data points, like an image. In this type of data, features are shared among several formations and one formation may not be the only cause of the feature. The BAHG becomes important when other procedures have to act upon the knowledge collected from the grouping process. For example, in the creation of models of indoor environments from planar surfaces it is necessary to use the grouping knowledge stored in the BAHG to join surfaces and remove surfaces that do not convey any added information to the shape of the room. This knowledge can also be used for other application specific tasks like object recognition. path planning, dynamic object tracking, and surface reconstruction. Most Bayesian networks that are used in the computer vision domain maintain separate representations for domain knowledge and uncertainty management. The result of that is the embedding of the knowledge gathered from the image processing into the algorithms so this knowledge becomes unavailable for further use.

For the specification of the construction of a BAHG the BAHG network has been developed. The strength of the BAHG network is the ability to specify declaratively a structure for the grouping of sensory data. The BAHG network allows for the specification of construction functions and therefore force the packaging of these procedures to particular nodes in the network. This declarative approach is extremely vital in helping to explain how a BAHG was created. This is akin to packaging the procedural knowledge required to construct the BAHG and compute new declarative knowledge at each node.


G
H
U
V
X
Y_i
Y_j
BAHG Constructor
G^m $P(G^m)$ $Bel(G^m)$
G^n $P(G^n)$ $Bel(G^n)$
H^m $P(H^m)$ $Bel(H^m)$
H^n $P(H^n)$ $Bel(H^n)$
U^m $P(U^m \mid G^m)$ $Bel(U^m)$
U^n $P(U^n \mid G^n)$ $Bel(U^n)$
V^m $P(V^m \mid G^m, H^m)$ $Bel(V^m)$
V^n $P(V^n \mid G^n, H^n)$ $Bel(V^n)$
X^m $P(X^m \mid U^m)$ $Bel(X^m)$
X^n $P(X^n \mid U^n)$ $Bel(X^n)$
Y_i^m $P(Y_i^m \mid V^m)$ $Bel(Y_i^m)$
Y_i^n $P(Y_i^n \mid V^n)$ $Bel(Y_i^n)$
Y_j^m $P(Y_j^m \mid V^m)$ $Bel(Y_j^m)$
Y_k^n $P(Y_k^n \mid V^n)$ $Bel(Y_k^n)$

BAHG Network

Bayesian Attributed Hypergraph (BAHG)

Figure 5.7: Example of the construction of a Bayesian attributed hypergraph.

Chapter 6

Experimental Tests and Results

This chapter covers the details of an experimental implementation of a Bayesian Attributed Hypergraph Network for the detection of corners and continuous surfaces utilizing data acquired from a compact laser camera called BIRIS [13]. This is the same sensor that was used for acquiring the data used in the examples to test the compatibility functions for perceptual grouping of 3-D data that were presented in chapter 4.

6.1 The BIRIS Range Camera

The BIRIS range measuring camera is composed of a standard CCD camera. a laser projector. and a modified lens. The lens is modified so that the iris is replaced by a double aperture iris. hence the name Bi-IRIS. The effect of the double iris is to create an out of focus image on the CCD so that any ray of light from a light source contacts the CCD at two different spots. figure 6.1.

The offset in the two contact spots can be used to measure the distance of the light source from the camera. The complete BIRIS system consists of a laser projector mounted at an offset from the BIRIS detector as shown in figure 6.2. The result is a compact laser

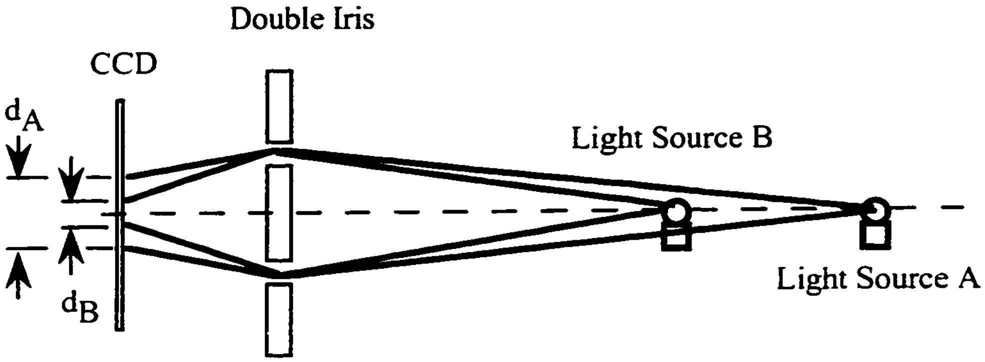


Figure 6.1: Method of computing range in the BIRIS sensor.

ranging device, the size of a standard CCD camera, without any moving parts. The sensor has a high immunity to ambient light and the ability to discriminate false measurements due to the selective region in which the reflected laser light can fall on the CCD sensor.

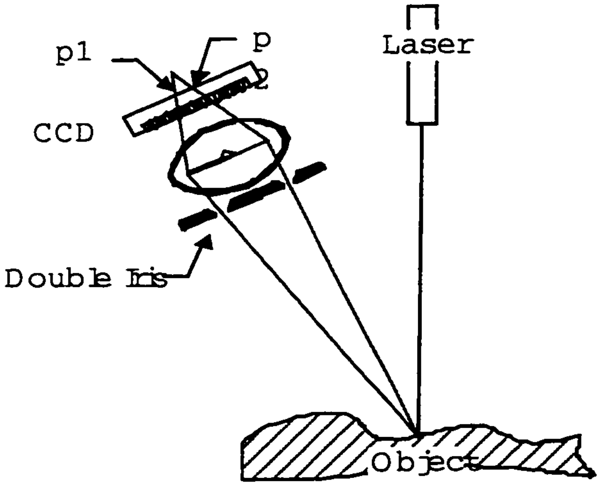


Figure 6.2: The BIRIS laser ranging system.

To acquire more data than a single point the projected laser light goes through a cylindrical lens resulting in a projected plane of laser light, figure 6.3 (a) and therefore eliminating the need of mirrors to scan the beam. The plane of light is aligned in the same direction as the vertical scan of the camera and therefore it is only necessary to measure the offset in the reflected stripe on the image plane along the horizontal scan. With a real-time dedicated processor it is possible to compute the range and intensity data at video rates. One acquisition (in $1/30^{th}$ s) from the sensor results in the computation of 256 range and intensity values along the projected plane of light reflected on the CCD detector. The characteristics of this particular version of the sensor are shown in table 6.1.

Laser power	24 mW, (two 12 mW laser projectors)
wavelength	680 nm (visible red)
Field of view	19 deg.
Focal Length	20 mm
Accuracy	3 mm @1 m, 14 mm @ 2 m, 42 mm @ 3 m
Range	0.5 m - 5.0 m

Table 6.1: Specifications for the BIRIS range sensor.

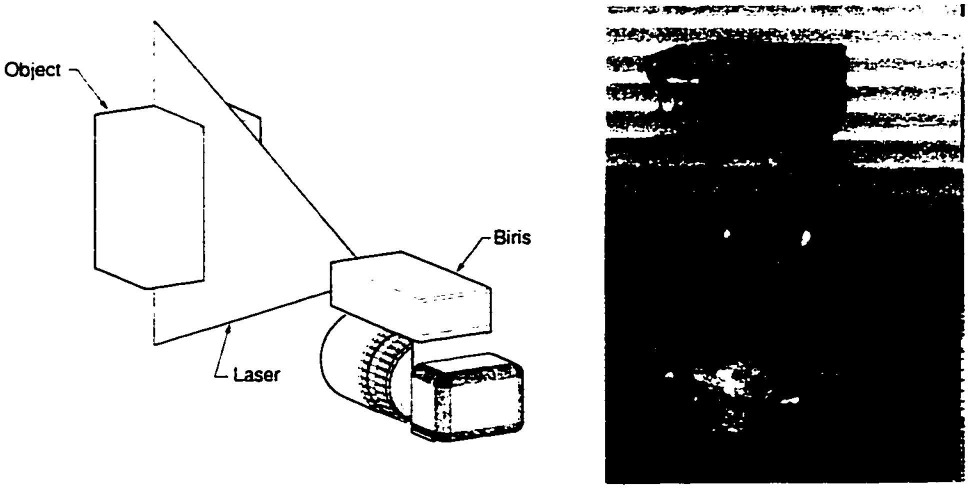


Figure 6.3: The BIRIS laser ranging system used for these experiments.

To acquire more data than that of a single acquisition the BIRIS sensor was mounted onto a pan and tilt unit, figure 6.3 (b). The ability to tilt the sensor is crucial in being able to acquire more data, since the field of view of the sensor is fairly limited.

When the Biris Laser Scanner is panned at a constant speed for a fixed tilt angle the result is a rectangular image of dimension 256 x N_{aq}, where N_{aq} is the number of acquisitions taken during the panning sequence. This process can be repeated for several different tilt angles so that a number of registered scans can be acquired. This thesis focuses primarily on the grouping of features taken from a single scan.

6.2 The BAHG Network: Detecting Corners and Continuity

For the experiments one network has been developed that models the detection of corners and continuity among planar surfaces. This BAHG network, depicted in figure 6.4, models the grouping of surfaces and their respective edges into corners and continuous surfaces.

This particular example is a consequence network depicting the belief values computed from evidence to hypotheses. This direction tends to be more natural for the application of the compatibility functions to the grouping process. Figure 6.4 is a perceptual grouping model depicting the existence of corners (**Cor**) and continuous surfaces (**Con**) from the existence of coplanar surfaces (**Cop**), parallel surfaces (**Pll**), vertices (**Ver**) , polygons (**Pol**), and finally a set of 3-D points (**3DPt**) and their respective edges (**Edg**). The 3-D points and edges are the only instantiated nodes in the network since they represent actual sensory data, the other nodes are hypothesized formations that contain belief values computed from the instantiated evidence.

This particular structure decomposes the formation of corners and continuous surfaces by relaxing one particular constraint among the grouping for each level in the network

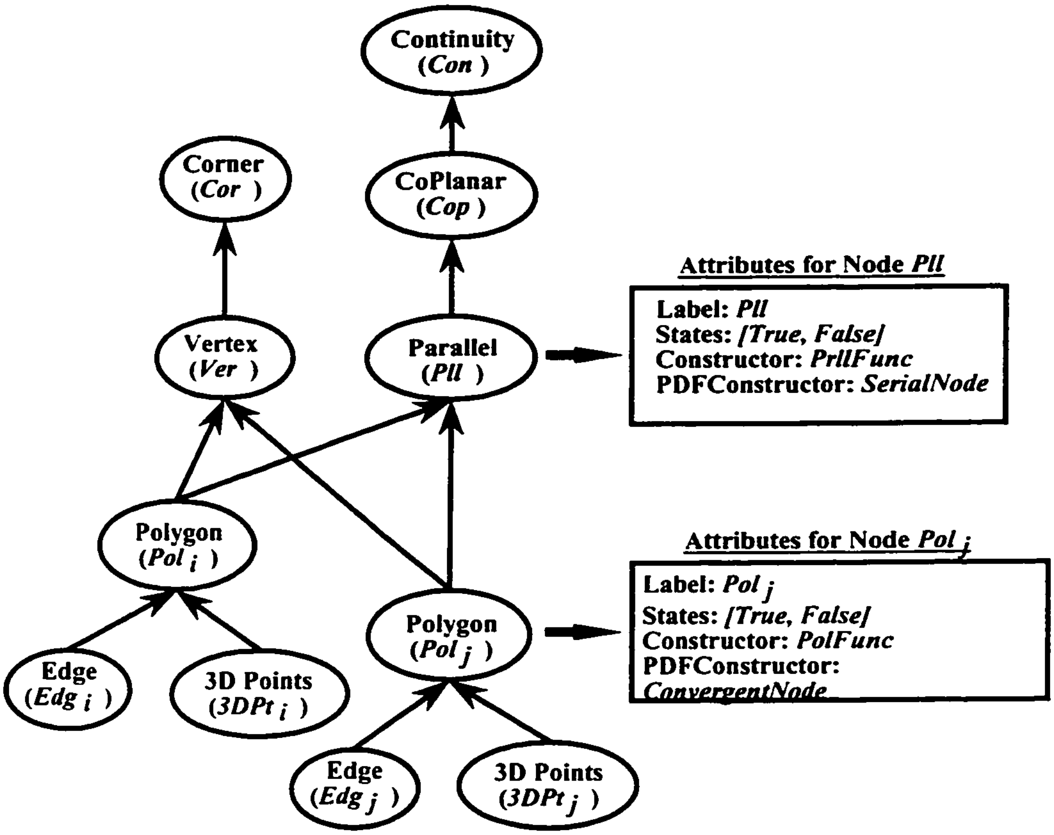


Figure 6.4: A Bayesian attributed hypergraph network for the detection of corners and continuity among planar surfaces.

as one traverses the network from top to bottom. This is a decomposition by parts operation with intermediate nodes that depict the relaxation of one single constraint at a time. This ideally leads to the use of one compatibility function for the computation of the conditional probability required at the node. In practice this is the case when similar geometrical features are being grouped, for example for the formation of parallel surfaces among two polygons one compatibility function (f_{prll}) is all that is required. For the grouping of dissimilar features, like the combination of 3-D points and their respective edges, two separate compatibility functions are required: one for the edges (f_{edge}) and the other for the formation of planes from the 3-D points (f_{plan}).

Vertices are the complementary to parallel surfaces so that they also use the parallel surface compatibility function to determine the conditionals. This particular formation is inserted so that all the conditional probabilities are computed based on the existence of the polygons. Some original instantiations of this network had the corners directly hypothesized from the absence of the parallel surface formation. The drawback of this assumption is that the absence of a formation can be caused by the absence of formations in the children. The result is an intuitively wrong result for the formation of corners from the absence of well formed polygons. This is an important point to consider when the network is being designed.

The coplanar formations apply the coplanar compatibility function (f_{copl}) to determine the conditionals for this particular node and finally, the conditionals required for the evaluation of continuity and corners are determined using the proximity (f_{prox}) compatibility functions.

As presented in section 3.3 the compatibility functions are first mapped to values bounded between 0 and 1 using the declining S-curve function $S(f_{PG}, 0, \beta, \gamma)$. Table 6.2 lists the values used for the arguments β and γ that were used in the declining S-curve mapping.

Compatibility	β	γ	Units
proximity	0.25	0.50	unitless
coplanarity	10	20	Rads
parallel	10	20	Rads
planarity	0.578	1.156	cm
edges	0.25	0.50	unitless

Table 6.2: Values for the arguments β and γ for particular compatibility functions.

These values were subjectively selected but the selection follows a certain reasoning. The proximity value of 0.25 represents the desire to consider surfaces with gaps whose areas are smaller than 1/4 the sum of the areas of the two neighbouring surfaces as proximal. anything beyond 0.5 is not proximal. This same argument can be applied to the edge compatibility value since it is also a ratio of areas. Surfaces are considered parallel if the surface normals are within 10 rads with respect to each other. and they are not parallel if beyond 20 rads. Coplanarity is similar to parallel surfaces and share the same arguments. The parameter values for the quality of the surface being considered as a plane was determined by computing the average of the variance of the data points of a typical set of points gathered from the BIRIS sensor.

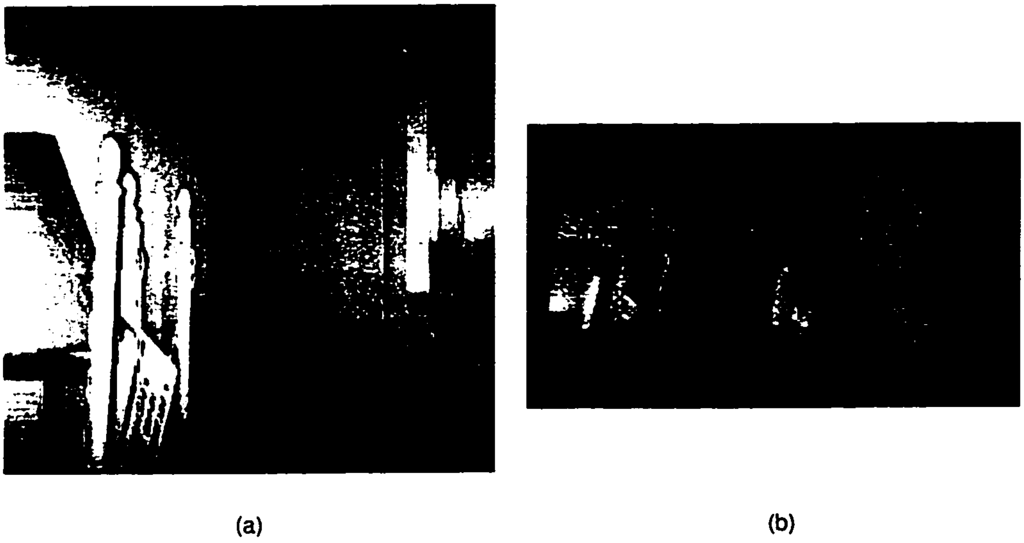
With this in mind the conditional probability equations are the following.

$$\begin{aligned}
P(Pol_i|Pt3D_i. Edg_i) &= \mathcal{S}(f_{plan}. 0, 0.578. 1.156)\mathcal{S}(f_{edge}, 0. 0.25. 0.50) \\
P(Pll|Pol_i. Pol_j) &= \mathcal{S}(f_{prll}. 0. 10. 20) \\
P(Ver|Pol_i. Pol_j) &= 1 - \mathcal{S}(f_{prll}. 0. 10. 20) \\
P(Cop|Pll) &= \mathcal{S}(f_{copl}. 0. 10. 20) \\
P(Cor|Ver) &= \mathcal{S}(f_{prox}. 0. 0.578. 1.156) \\
P(Con|Cop) &= \mathcal{S}(f_{prox}, 0. 0.578. 1.156)
\end{aligned} \tag{6.1}$$

The BAHG network in figure 6.4 is similar to a conventional Bayesian network interface except for the specification of the **Constructor** and **PDFConstructor** functions as well as the subscripted labels of the nodes **Edg**, **Pt3D**, and **Pol**. Each individually labeled node has a particular **Constructor** function that creates each node in the BAHG and their associated compatibility function values. This is depicted for the **Parallel** and **Polygon** nodes. In the following examples the compatibility values have been computed apriori using procedures developed from the algorithms described in chapter 4 and the results are stored on file. The **Constructor** functions read this information from a file instead of an attributed hypergraph. The **PDFConstructor** functions use results from the compatibility functions to compute the conditional probabilities. Since this network is a consequence network then these steps can be performed at the same time the node is created. Note that the **Parallel** node uses the generic **PDFConstructor** function *SerialNode* that is used for serial connections because only one compatibility function is used in this case, unlike the case for the **Polygon** node that uses the *ConvergentNode* **PDFConstructor** function.

6.3 Test Data Case 1: The Robot Lab

Figure 6.5 is an example of a typical scan taken by the BIRIS Laser Scanner of part of a robot laboratory room at the Institute for Information Technology.



(a)

(b)

Figure 6.5: Intensity image and 3-D data of a scan from the BIRIS sensor.

The layout of the room is approximately the shape of that shown in figure 6.6.

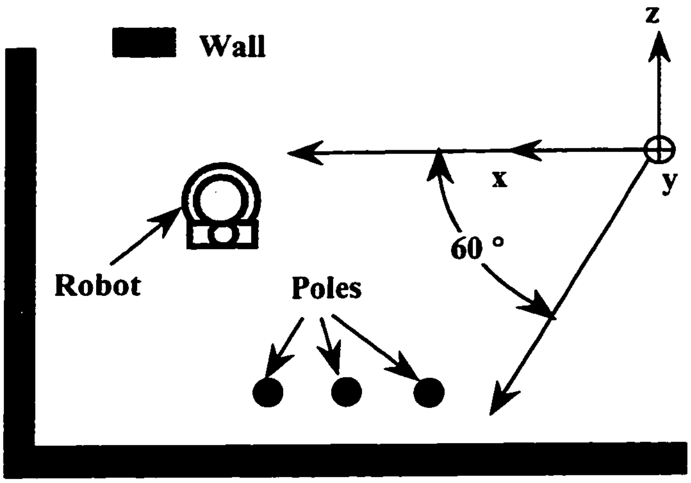


Figure 6.6: A schematic of the top view of the robot lab showing sensor position.

This image size is 256 x 256 pixels and was taken using a tilt angle of 0° and a pan angle of 60°. It is an ideal set of real test data because it contains possible corners and large planar surfaces.

6.3.1 The Planar Surfaces and Boundaries

The results of performing the segmentation on the range data in figure 6.5 are shown in figure 6.7 as both an intensity image and an orthographic projection of the 3-D data. The individual segments in the intensity image are shown using separate gray values to represent each segment. The advantage of maintaining the points in an image is that one can take advantage of the already established ordering in the 2-D image.

The surface boundaries and their respective high curvature points are depicted in figure 6.8 (a) and (b) respectively. The high curvature points are highlighted as white points while the edge is in gray pixels. Note that the representation of the surfaces in figure 6.8 (b) is a polygon created by using the high curvature points as the polygon

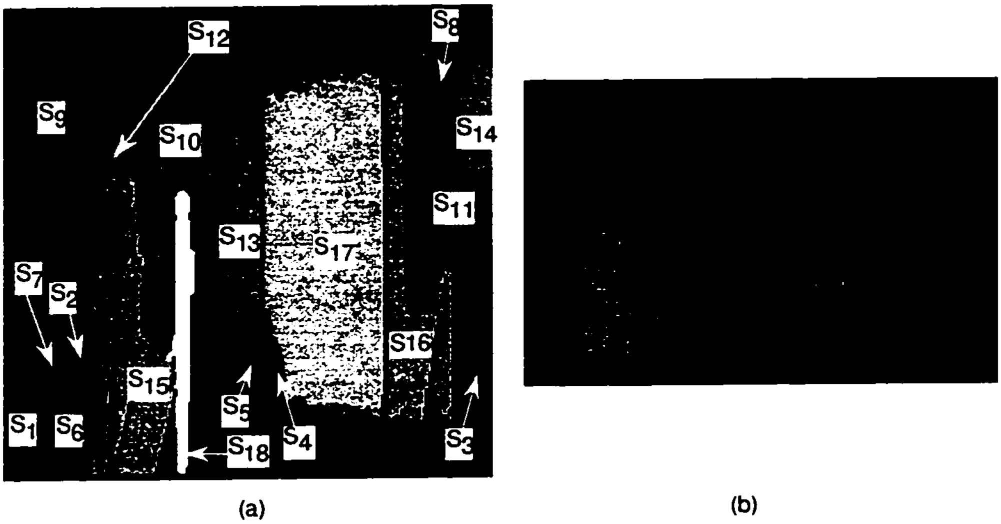


Figure 6.7: Surfaces extracted from the 3-D Points, label image (a) and orthographic projection (b).

points.

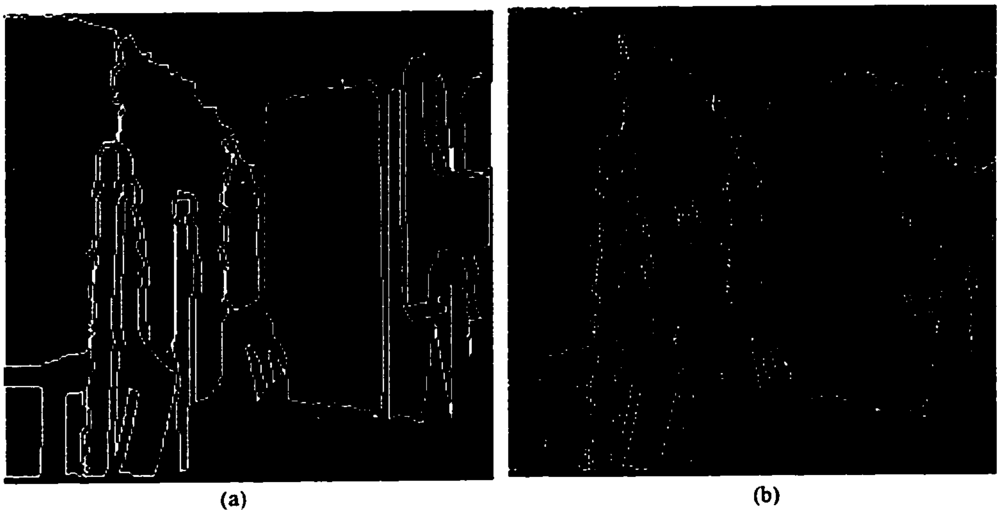


Figure 6.8: Boundaries extracted from the surfaces.

These results are represented as two separate attributed graphs in the code and saved as files for access to other processes when required.

6.3.2 Results

Results are presented in two manners: an example of a BAHG is shown of part of the sensory data showing some belief values and a table is presented depicting the belief in the formation of corners and continuous surfaces among neighbouring surfaces.

Reflecting back onto the labeled surfaces in figure 6.7 (a) a view depicting the grouping of surfaces S_{10}, S_{13}, and S_{17} is shown in figure 6.9. These particular surfaces are right in the center of the image and show both the formation of a corner and that of two continuous surfaces.

The Edg_i and $Pt3D_i$ hyperedges are instantiated as **TRUE** and therefore have belief values of 1.0 associated with them. Also the only common nodes shared among the Bayesian networks represented in this BAHG are the Pol_i nodes. In this case the corner and continuity formations labeled Cor_{22} and Con_{22} respectively correspond to the grouping of surfaces S_{13} and S_{17} while the corner and continuity formations labeled Cor_{15} and Con_{15} correspond to the grouping of surfaces S_{13} and S_{10}. These are arbitrary labels that correspond to the sequence in which the grouping was performed. In this example there are 25 total surface groupings of which the belief values are presented in table 6.3.

The majority of the belief values are close to 0. This is especially true for the continuity among surfaces. The principal reason for this is the stringent values for $\beta = 10$ and $\gamma = 20$ rads that result in very few parallel surfaces. In turn this will result in more corners being detected than continuous surfaces.

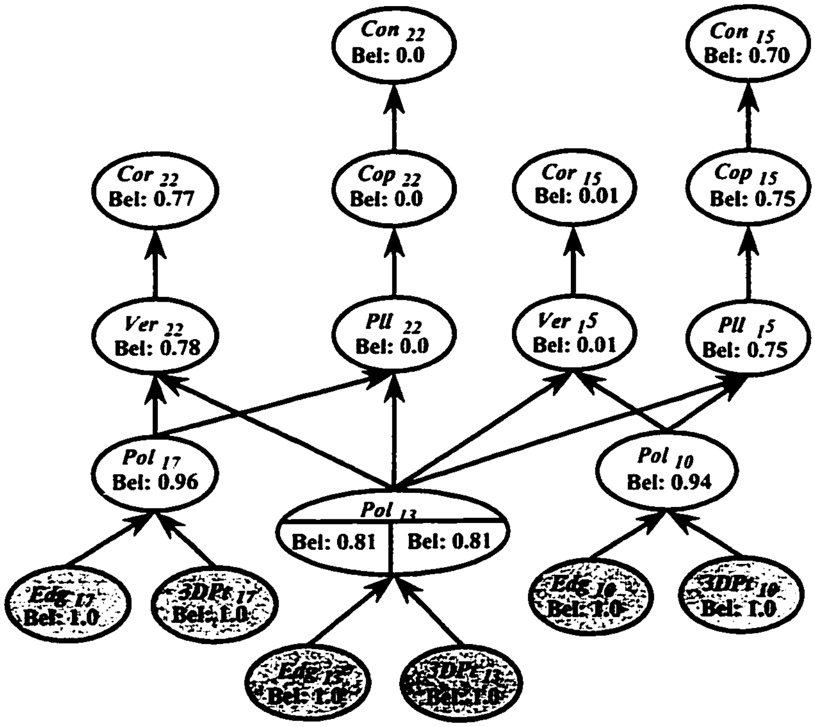


Figure 6.9: A partial view of the robot lab BAHG for surfaces S_{10}. S_{13}. and S_{17}.

6.3.3 Analysis of Results

The majority of the results are close to the expected formations if one was to look at the room and group the surfaces. The first important point is that the formation of a corner is counter to the formation of continuous surfaces. The belief values reflect this in the following manner. When the belief of the formation of a corner is high the belief in the formation of a continuous surface is low. and the converse is true. The exception to this are cases where both values are 0.0 which occur when the proximity between the surfaces is relatively large.

Using the BAHG it is rather simple to infer why these low values in both corners and continuity exists. it is simply a matter of searching through the graph looking for

S_i	S_j	Cor	Con	Comment
1	6	0.32	0.0	$S(f_{prox}$ low
1	9	0.87	0.0	**OK**
4	5	0.0	0.0	$S(f_{plan}(S_4)) \approx 0.0$
4	17	0.0	0.0	$S(f_{plan}(S_4)) \approx 0.0$
5	10	0.0	0.0	$S(f_{plan}(S_5)) \approx 0.0$
5	13	0.0	0.0	$S(f_{plan}(S_5)) \approx 0.0$
5	17	0.0	0.0	$S(f_{plan}(S_5)) \approx 0.0$
6	9	0.63	0.0	$S(f_{edge}(S_6))$ low
6	12	0.0	0.0	**OK** Jump Edge
8	11	0.0	0.0	$S(f_{plan}(S_8)) \approx 0.0$
8	14	0.0	0.0	$S(f_{plan}(S_8)) \approx 0.0$
9	10	0.87	0.0	**OK**
9	12	0.22	0.0	**OK** Jump Edge
10	12	0.32	0.0	**OK** Jump Edge
10	13	0.01	0.72	**OK**
10	15	0.0	0.0	**OK** Jump Edge
10	17	0.9	0.0	**OK**
10	18	0.0	0.0	**OK** Jump Edge
11	14	0.0	0.0	**OK** Jump Edge
11	16	0.0	0.0	**OK** Jump Edge
12	15	0.0	0.0	**OK** Jump Edge
13	17	0.77	0.0	**OK**
15	18	0.0	0.0	**OK** Jump Edge
16	17	0.02	0.27	$S(f_{plan}(S_{16})) \approx 0.0$
17	18	0.0	0.0	**OK** Jump Edge

Table 6.3: Belief values in surface corners and continuity in figure 6.7 (a).

occurrences in low compatibility values. Table 6.3 also lists a small comment for each grouping that mentions reasons why the belief values for particular surface formations are low. Low values for both corner and continuity do not necessarily signify bad results they can reflect situations where the edge shared by the two adjacent surfaces is in fact a jump edge and the surfaces cannot form either a corner or continuous surface. This is the case for surfaces that are adjacent to the robot's surface (S_{11}). those being surfaces S_8. S_{16} and S_{14}. The same holds true of the surfaces corresponding to the poles (S_{12}. S_{15} and S_{18}) where all of the surfaces adjacent to these have corresponding low belief values for being possible corners or continuous surfaces. These results suggest that another surface grouping, known as a discontinuity, should perhaps be introduced to the model. Discontinuities are generally not considered a perceptual grouping formation but certainly do add knowledge that can be used for interpreting a scene.

In other circumstances. in particular surfaces S_4 and S_5. the belief values in the formation of the polygons are so low that of course there cannot be any formation of corners and continuous surfaces. Some results are not so positive. It would have been desirable to have surfaces S_{16} and S_{17} to result in a high belief value of being a continuous surface. This was not so. the reason was the low planar surface compatibility value of surface S_{16}. This low certainty value is reflected throughout any other formations where surface S_{16} is involved. Therefore. low belief values in any formations suggest a need to search for the source of this low value. It is difficult. though. to determine if the low value is real or caused by a computational error without using redundant compatibility functions.

6.3.4 Environment Modeling

One of the strengths in using a BAHG representation is that continuity among the surfaces is still maintained as well as belief values in the formation of surface groupings. In this particular case a very simple procedure is to search through the BAHG and collect those hyperedges that share similar surfaces and significant belief values in continuity and corner values. This is easier presented to the user as an attributed graph where the edges represent possible corner or continuous surfaces and the attribute value is a tuple of the respective belief values in those formations.

For environment modeling. detecting continuity allows the joining of surfaces and more realistic models to be created. Figure 6.10 is an attributed graph representing the formation of corners and continuity among the planar surfaces that have been extracted from the BAHG in figure 6.9.

The edges connecting the nodes in figure 6.10 have been classified into 3 different types based on the belief values. The dotted edges are connections with belief values greater than 0.7 and are considered as connected surfaces. The dashed edges represent surfaces that are unconnected. clearly the belief values for both corner and continuity are 0. The solid edges represent an unknown relationship among the surfaces. The belief values are

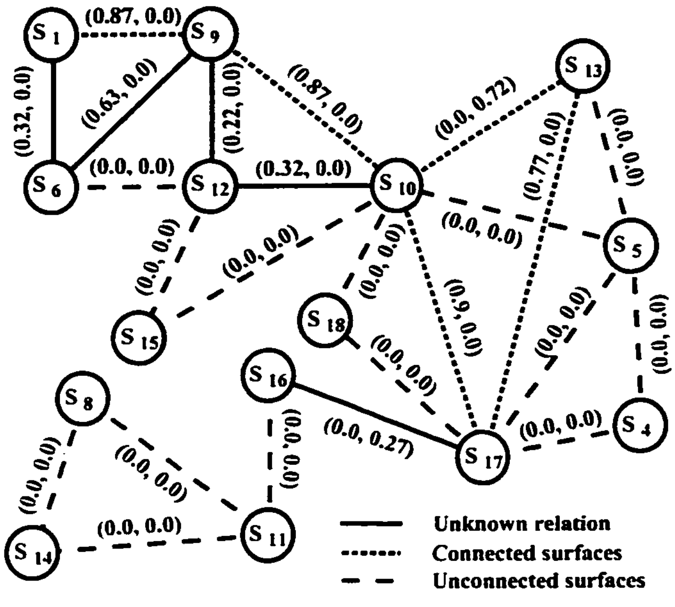


Figure 6.10: An attributed graph modeling the formation of corners and continuous surfaces.

somewhere between 0.2 and 0.7 and may have resulted in connected surfaces if the belief values had been high enough and therefore require further investigation. The results of the grouping procedure can also be summarized as an image, shown in figure 6.11 , which depicts those surfaces that are continuous and/or corners as one intensity, surfaces that are in the foreground in another intensity, and those surfaces with poor results in dark.

This is then an example of one of the simplest forms of reasoning that can occur using the BAHG and that would be harder to extract if simply multiple Bayesian networks had been maintained instead of the BAHG.

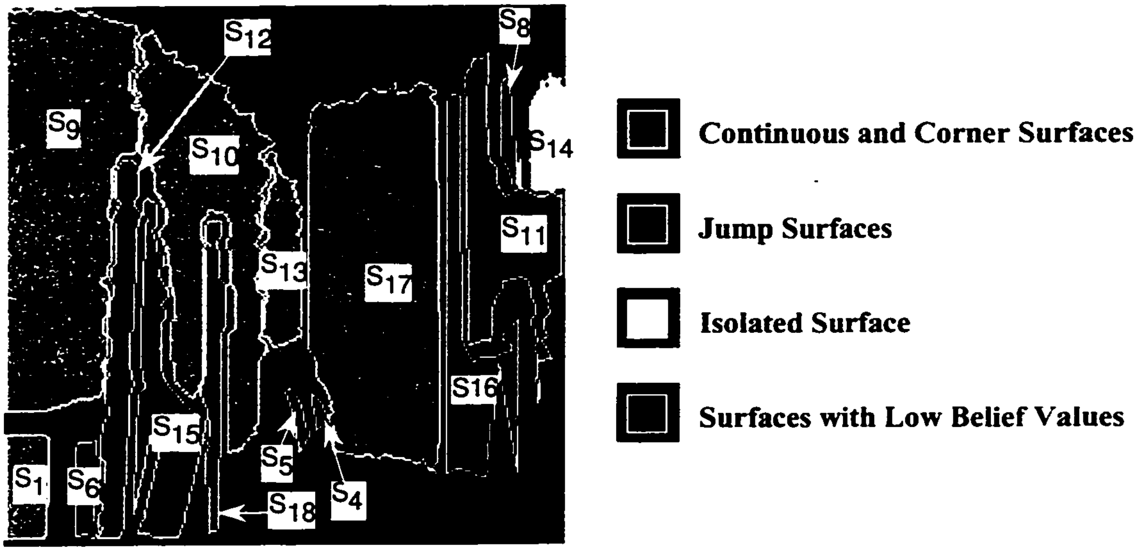


Figure 6.11: Grouping of surfaces with belief values ≥ 0.7.

6.4 Test Data Case 2: A Wall and Ceiling Junction

The next case is representative of a scan taken from the junction of a wall to a ceiling. Unlike a simple line drawing of a wall to ceiling junction. in this case part of the wall extends out like a column extending from the ceiling to the floor and the ceiling is populated by a number of light fixtures that are not detected by the sensor. Figure 6.12 is the intensity image from the camera that corresponds to the top scan of the composite image introduced in the introduction as 1.2 (a).

This particular example has an abundance of more continuous surfaces than that presented in case 1 and the surface quality is much higher. This had to do with a couple of reasons. the range data is not as far and the laser power source was improved.

This image size is 256 x 512 pixels and was taken using a tilt angle of 30° and a pan angle of 140°. Figure 6.13 is an orthographic projection of the 3-D data points taken for this acquisition.

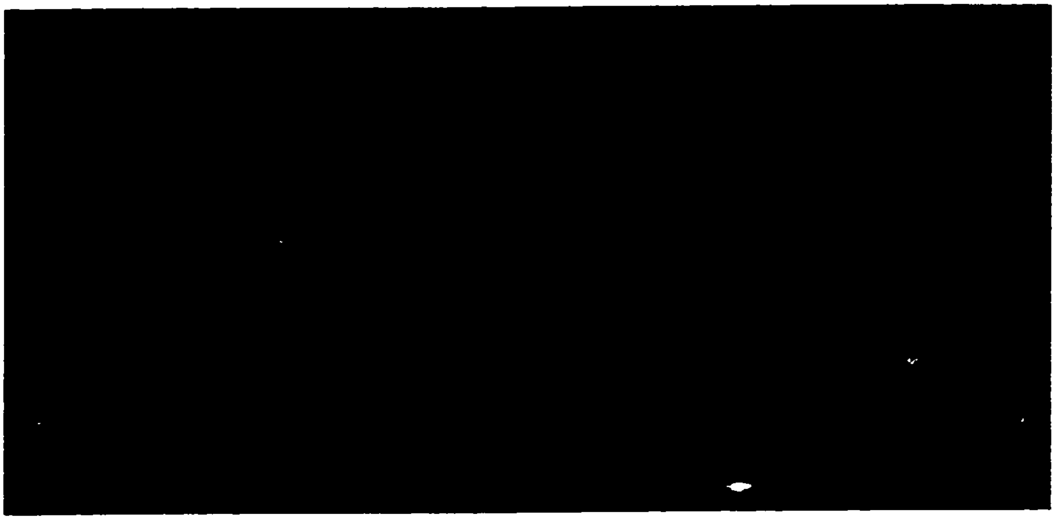


Figure 6.12: Intensity image of a wall and ceiling junction.

6.4.1 The Planar Surfaces and Boundaries

The results of performing the segmentation on the range data for the wall and ceiling junction are shown in figure 6.14 as an intensity image where one uniform intensity value represents a planar surface. For clarification the same segments can be displayed as a set of range values as depicted in figure 6.15.

The surface boundaries and their respective high curvature points are depicted in figure 6.16. The high curvature points are highlighted as red points while the edge is in a white color.

The boundary extractor was applied to the labeled image that was processed before hand with an erosion morphological operator. This can result in the further segmentation of the surfaces, as depicted in surfaces S_{22} and S_{25} in figure 6.16, resulting in the boundary tracking algorithm choosing only one of the segments and ignoring the other. This can be corrected by maintaining all boundaries that have been tracked about a surface instead

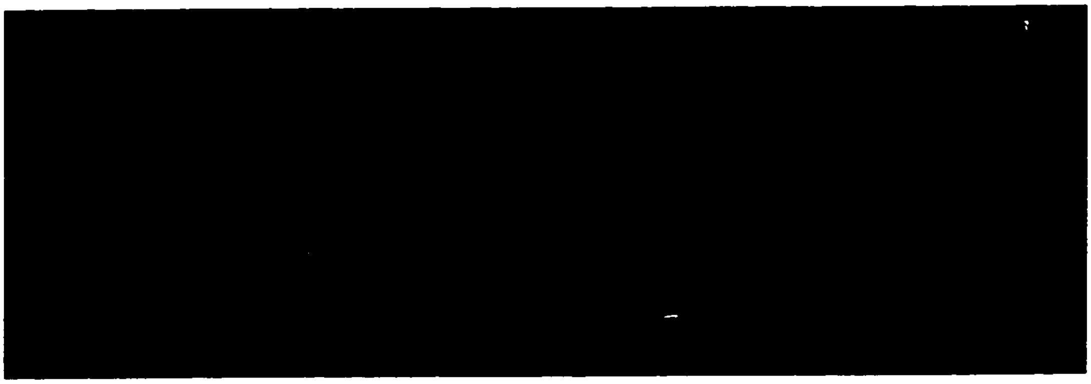


Figure 6.13: Orthographic projection of the 3-D points for the wall and ceiling junction.

of the first tracked boundary.

6.4.2 Results

The BAHG network depicted in figure 6.4 was applied to the attributes computed from the surface segmentation process and boundary extraction routines. Results are presented as a table since it is difficult to present the full BAHG without sacrificing readability. Again the grouping process is applied only to neighbouring surfaces reducing the combinatorial complexity among all surfaces dramatically. A total of 27 surfaces were extracted from the image resulting in 44 possible binary groupings of which 3 resulted in the inability of the proximity procedure to determine a common surface reducing the pairing combinations to 41. The belief values in the formation of corners and continuous surfaces are presented in table 6.4.

6.4.3 Analysis of Results

The comments in table 6.4 refer to particular reasons why combinations have low belief values. Those surface combinations that appear to have reasonable results have a com-

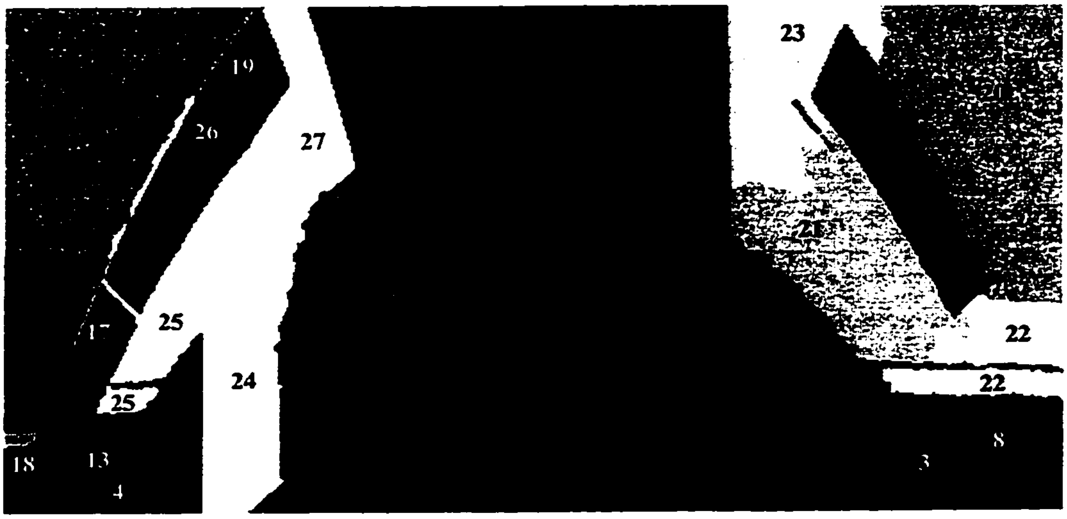


Figure 6.14: Labeled image of surfaces extracted from range values in figure 6.13.

ment of **OK** beside them. One of the first things to note is the low compatibility value in the formation of the edges for surfaces S_3. S_4. S_{22}. and S_{25} resulting in low belief values in the formation of "Corners" and "Continuous" surfaces. Surfaces S_3 and S_4 are small and lead to bad results in the calculation of the boundary. The further segmentation of surfaces S_{22} and S_{25} due to the erosion operation will have to be dealt with further by considering multiple boundaries for a surface or in some other fashion. Not all surface pairings that have low "Corner" and "Continuity" belief values are bad results they may be caused by legitimate "Jump Edges". The only manner to distinguish these two cases is to look at the belief value in the formation of a "Polygon". A high belief in the formation of a "Polygon" and low values for coplanar and proximity will signify a "Jump Edge".

6.4.4 Environment Modeling

Similar to the situation presented for the "Robot Lab" test case it is possible to look for continuity among surfaces with reasonable belief values as corners or continuous surfaces.

Figure 6.15: Orthographic projection of the surfaces depicted in figure 6.14.

This again can be shown as an attributed graph where the edges represent possible corner or continuous surfaces and the attribute value is a tuple of the respective belief values in those formations and shown in figure 6.17.

The gray edges in figure 6.17 are connections with belief values greater than 0.7 and are considered as connected surfaces that consist of both corners and coplanar continuous surfaces. There are a total of 14 of these surfaces out of 29 possible surface combinations that had good polygonal representations. Out of the left over 15 surface relationships. 9 have belief values in the range of 0.2-0.7 and are classified as unknown while 6 are definitely unconnected surfaces. Similarly the results of the grouping procedure can also be summarized as an image. shown in figure 6.18 . which depicts those surfaces that are continuous and/or corners as one intensity and surfaces with poor results in dark. In this case there are no surfaces that can be considered to be in the foreground.

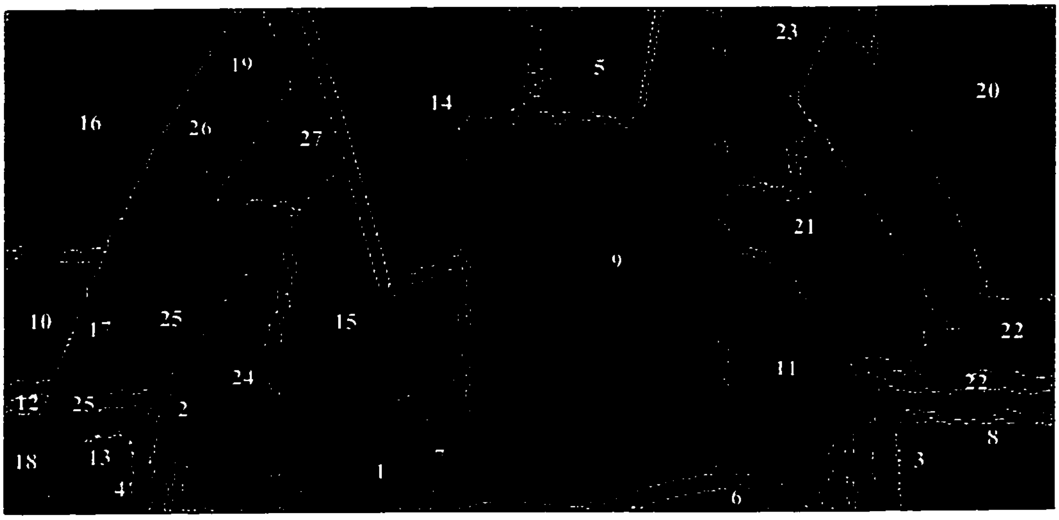


Figure 6.16: Boundaries extracted from the surfaces.

6.5 Summary of Experimental Results

In this chapter an example was presented of a BAHG network for the detection of corners and continuity among planar surfaces extracted from an ordered set of 3-D points. The example presented the design of the BAHG network as a decomposition of parts of connected surfaces into individual surfaces with particular constraints like parallel. coplanar. and vertices. The surfaces themselves are polygons and these are formed from a planar surface with an associated boundary. The five compatibility functions of proximity. parallel. coplanar. edges. and planar were used to determine the required conditional probabilities for computing the belief values in the corner and surface continuity formations. When the BAHG network is applied to a particular set of data a BAHG is formed that contains multiple instantiations of Bayesian networks between adjacent surfaces in the data.

The BAHG network example was applied to two sets of 3-D sensory data. one taken

S_i	S_j	Cor	Con	Comment
1	7	0.13	0.63	$S(f_{prll})$ low
1	15	0.22	0.57	$S(f_{prll})$ low
1	24	0.15	0.65	$S(f_{prll})$ low
2	4	0.0	0.0	$S(f_{edge}(S_4)) \approx 0.0$
2	10	0.80	0.0	**OK**
2	24	0.03	0.74	**OK**
2	25	0.0	0.0	$S(f_{edge}(S_{25})) \approx 0.0$
3	8	0.0	0.0	$S(f_{edge}(S_3)) \approx 0.0$
3	11	0.0	0.0	$S(f_{edge}(S_3)) \approx 0.0$
3	22	0.0	0.0	$S(f_{edge}(S_{22})) \approx 0.0$
5	9	0.82	0.0	**OK**
5	14	0.05	0.90	**OK**
6	9	0.42	0.0	$S(f_{edge}(S_6)) = 0.5$
6	11	0.21	0.0	$S(f_{edge}(S_6)) = 0.5$
7	9	0.16	0.0	**OK** Jump Edge
7	15	0.19	0.63	$S(f_{prll})$ low
8	22	0.0	0.0	$S(f_{edge}(S_{22})) \approx 0.0$
9	11	0.05	0.0	**OK** Jump Edge
9	14	0.95	0.0	**OK** Jump Edge
9	15	0.08	0.0	**OK** Jump Edge
9	21	0.65	0.0	**OK** Jump Edge
9	23	0.91	0.0	**OK**
10	12	0.06	0.25	$S(f_{edge}(S_{12}))$ low
10	16	0.01	0.92	**OK**
10	24	0.0	0.0	Bad Adjacency
11	22	0.0	0.0	$S(f_{edge}(S_{22})) \approx 0.0$
12	25	0.0	0.0	$S(f_{edge}(S_{25})) \approx 0.0$
13	25	0.0	0.0	$S(f_{edge}(S_{25})) \approx 0.0$
14	15	0.92	0.0	**OK**
14	27	0.02	0.0	**OK** Jump Edge
15	24	0.02	0.87	**OK**
15	27	0.89	0.0	**OK**
16	24	0.08	0.0	Bad Adjacency
16	25	0.0	0.0	$S(f_{edge}(S_{25})) \approx 0.0$
16	27	0.05	0.75	**OK**
20	21	0.01	0.91	**OK**
20	22	0.0	0.0	$S(f_{edge}(S_{22})) \approx 0.0$
20	23	0.01	0.79	**OK**
21	22	0.0	0.0	$S(f_{edge}(S_{22})) \approx 0.0$
21	23	0.0	0.89	**OK**
24	27	0.50	0.50	$S(f_{prox}) \approx 0.5$

Table 6.4: Belief values in surface corners and continuity in figure 6.15 (a).

of a robot laboratory and the other from the junction of the ceiling with a wall. The robot lab example is representative of a situation of several fragmented planar surfaces that were occluded by the robot and other objects in the environment. The wall ceiling example contained less fragmented surfaces and several examples of continuous and corner surfaces. For both types of data the same values for γ and β were utilized since they are invariant to the quality of the data. Grouped data surfaces from both test cases demonstrated expected results. Those surfaces that were not grouped. and should have been. were due to poor perceptual grouping values that were caused by noisy data or in some cases faulty tracking of a surface's boundary.

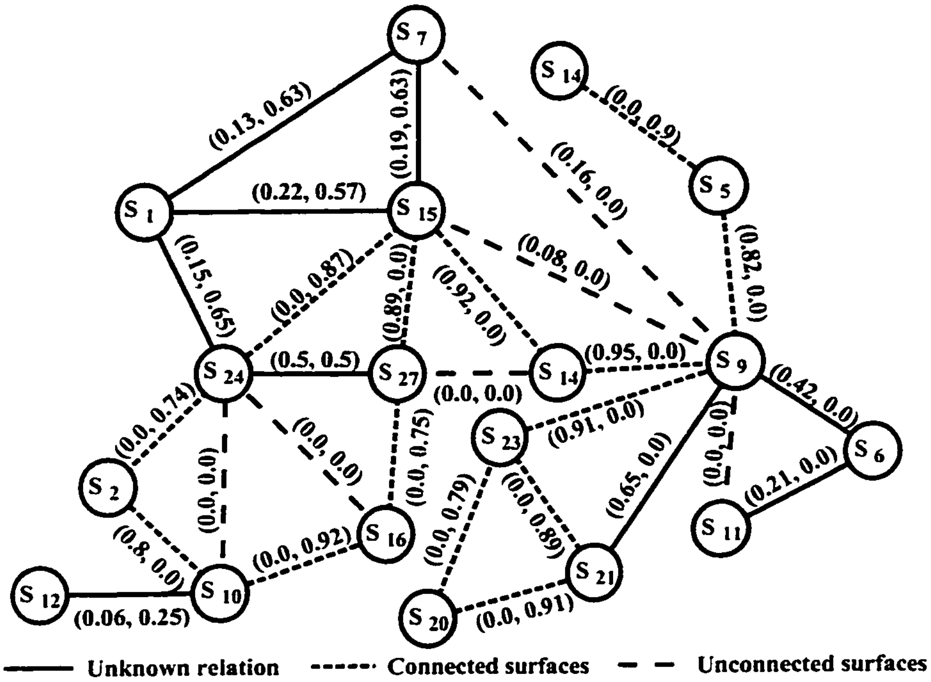


Figure 6.17: Attributed graph of the formation of corners and continuous surfaces.

It was shown that the BAHG can be used to infer the surfaces that should be connected either as coplanar or corner surfaces. These connected surfaces can then be used to create a more complete geometric model of the scanned environment. Even though it is possible to determine good candidates to be joined no algorithm was developed to actually join these surfaces. The difficulty in joining a set of surfaces, is determining what sections of the surface are to be extended and joined. A possible solution for this is to use the limits computed of the common surface between two neighbouring surfaces, see section 4.5, as a guideline in joining the surfaces. Surfaces to be connected can be extended within that common area until they intersect . A more challenging issue is to determine how these new hypothesized surface sections are to be represented in the BAHG since they are not composed of real 3-D points. These issues are all beyond the scope of this thesis.

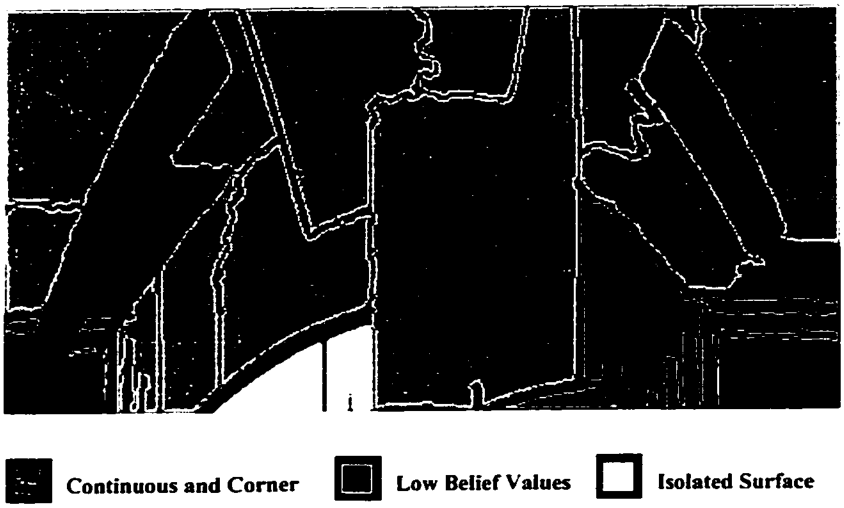


Figure 6.18: Grouping of surfaces with belief values ≥ 0.7.

A threshold belief value of 0.7 was used as to determine which surfaces are considered to be grouped. This value was arbitrarily chosen and is fairly low. This is a consequence of two factors: First, the approach taken for the combination of certainty values at a convergent node was the product rule which favours low certainty values and secondly the propagation of uncertainty in a Bayesian network is also a product of conditional probabilities and is sensitive to low belief values. So to achive relatively high belief values in the formations all the evidence must have high certainty values associated with them. The strength in the BAHG representation is the explicit manner in declaring the grouping process, using the BAHG network and compatibility functions, and in the modular decomposition of the grouping that allows for easier addition of new evidence.

Chapter 7

Conclusions and Future Work

In this chapter a summary of the work presented in the thesis is provided along with a review of the contributions to the perceptual grouping of 3-D surfaces for the purpose of environment modeling. This is followed by suggestions of future work based on problems and limitations that were identified throughout the period of this research.

7.1 Summary

The motivation of this thesis is to develop an approach for the detection of higher level formations among 3-D surfaces so that rational decisions can be made for the joining of surfaces. The main characteristics of sensory range data are: *(i)* They are susceptible to highly non-linear and non-gaussian levels of noise that lead to over segmentation of the data when surfaces are extracted, *(ii)* redundant information is common when a good signal is received from the sensor making it important to derive algorithms that can remove some of the redundant information, *(iii)* sparse surfaces are more often the norm due to the inability of the sensor to detect uniformly throughout a full acquisition. For indoor environments, occlusion among objects is common thus leading to irregular shaped boundaries for the surfaces. The ultimate goal is to develop an approach that could be used to bridge the gap between detailed sensory data and abstract labeled groupings among surfaces extracted from the sensory data.

In the attempt to meet this goal it became relevant that to simply create a set of procedures that could perform the grouping process would not benefit the research community on a whole. The general outcome of these systems are that procedural knowledge and representations are so tightly connected that it becomes difficult to separate the two forms of representation and re-use them. It is also difficult to determine how decisions were made by the system. In a similar fashion it is also important to maintain a belief value in the formation of groupings among the features extracted from the sensor data. This added information gives the user of the system a basis on the quality of the decision process.

This thesis presented an approach for the specification and management of procedural and declarative knowledge required in the grouping of 3-D surfaces for the modeling of large environments. The approach developed is a Bayesian attributed hypergraph (BAHG) which combines the benefits of attributed hypergraphs and Bayesian networks. Attributed hypergraphs have been shown to be a powerful tool for the representation of declarative knowledge in the form of attributes and relationships. On the other hand Bayesian networks offer a structured approach to reasoning with uncertainty. Their use of a network to represent independent relationships has a natural fit to the concept of decomposition by parts used for the automated modeling of 3-D sensory data. Procedural knowledge is still maintained as algorithms coded in a programming language. like C/C++. but now are packaged as construction functions for the BAHG that rely heavily on the use of compatibility functions. These also behave as measures in the quality of fit between the 3-D data and their respective models.

Certain intermediate goals were required in achieving the ultimate goal of grouping

3-D surfaces for environment modeling. These intermediate goals were the following:

1. Algorithms had to be developed to extract from the sensory data surfaces and their corresponding boundaries. For indoor environments this corresponds to planar surfaces and their respective polygonal boundaries.

2. The adaptation of an existing uncertainty management technique for the grouping of 3-D surfaces. This corresponded to the adaptation of Bayesian networks for the grouping of 3-D surfaces and the use of compatibility functions to compute the corresponding conditional probabilities.

3. The development of suitable perceptual grouping compatibility functions for 3-D planar surfaces.

4. The development of a unified representation for uncertainty and geometric modeling. This of course is the Bayesian attributed hypergraph.

The result was the implementation of a BAHG along with a graphical interface (a BAHG network) and several compatibility functions for measuring the planar. coplanar. and proximity quality between surfaces and the boundary and planar quality of the polygonal surfaces. The primary input is a set of 3-D range points and intensity values with the final output being a Bayesian attributed hypergraph (BAHG). The BAHG contains enough information about the 3-D polygonal surfaces so that a geometric model of the environment can be constructed. The BAHG was implemented using the Java object oriented language along with the BAHG Constructor.

This implementation was tested on the detection of corners and continuity among planar surfaces for actual 3-D range data of indoor environments. This particular data is prone to noise, fragmentation and occlusion and therefore grouping is rather difficult.

7.2 Research Contributions

While the domain of application for this work is in the modeling of indoor environments from 3-D sensory data the main contributions are in the management of uncertainty for the grouping of features extracted from this data. Much less effort was placed on the actual modeling component. Instead a framework was developed for the specification of the grouping of surfaces and management of uncertainty in that grouping process. The end result is a representation that maintains belief values in geometric formations that can be further used as knowledge for operations requiring geometric properties. i.e environment modeling. path planning. and/or virtual reality. In this section several contributions to the research community are highlighted and comparisons are made to other work in this domain. These contributions are primarily the contents of chapters 3. 4. and 5.

7.2.1 A Framework for Managing Uncertainty for the Grouping of 3-D Surfaces

A framework was developed for the management of uncertainty for the grouping of 3-D surfaces based on Bayesian networks. It was felt that Bayesian theory for uncertainty management was sufficiently developed to be applied to the domain of computer vision and 3-D surface grouping. The use of Bayesian networks to the computer vision domain is not unusual especially in the community for object recognition. What is of interest is the approaches used for constructing the network and the computing of the conditional probabilities. The particular framework presented uses a decomposition by parts approach to define the structure of the network and a set of compatibility functions for computing the conditional probabilities. It proposes that the network be designed from the top down and that each level of the network corresponds to a particular constraint among the features to be grouped. These constraints are directly related to a compatibility function that measures the quality of fit of a feature or set of features to the hypothesized model represented by the node. This is generally different from most conventional uses of Bayesian networks to the domain of computer vision. In most circumstances conditional probabilities are static and computed from statistical experiments that relate the probability of the formation of the child node given the parent nodes. For these types of systems evidence is entered at the leaf nodes and propagated throughout the network. It is not clear how one could introduce evidence to non-leaf nodes without instantiating those nodes and therefore removing all other circumstantial evidence via the principal of "d-separation". In the approach taken in this thesis the evidence is reflected as conditional probabilities and none of the internal nodes between the root and leaf nodes get instantiated. The consequence of this is that the network will not handle apriori probabilities and must be computed in a bottom-up fashion from evidence to hypotheses.

7.2.2 Compatibility Functions for 3-D Perceptual Grouping

Since compatibility functions are a crucial part of the construction process for a Bayesian network several compatibility functions were developed. In some circumstances these compatibility functions required new innovative approaches to be developed. in particular. the estimation of proximity. neighbourhood. and polygonal representations of the surfaces. These are all related to the surface's boundary. which comes as no surprise since it is the component of a surface that is the most difficult to detect using 3-D range data. The other compatibility functions. planar and coplanar. are derived from known geometric principles.

Estimating the surface's boundary as a polygon requires filtering of the edges and computing high curvature points along the edge as well as the replacement of continuous high curvature sections with straight lines. This leads to a fast algorithm with linear complexity $O(n)$, where n is the number of points in an edge.

The determination of a common region between surfaces is akin to a region growing operation with the exception that the complete edge is not grown but the line segments are translated until collisions with other surface boundaries are detected. This is ideally suited for determining common regions among surfaces while region growing would result in an exaggerated common region if the surfaces were relatively far apart. This type of determination of a common region among surfaces is not common in 2-D computer vision because there does not exist in that domain large gaps between surfaces. As well, in the 3-D computer vision community the majority of surface reconstruction has occurred with dense 3-D range data of small parts. Fragmentation is more common in data from large environments where the sensor's limitations are stretched to the limit.

7.2.3 A Unified Representation for 3-D Modeling and Uncertainty Management

From this research a unified representation integrating the positive attributes of attributed hypergraphs and those of Bayesian networks was developed. The driving force for this comes from the desire to integrate declarative knowledge into one representation along with belief values in the formation of groups among the data. The Bayesian attributed hypergraph (BAHG) was implemented and it incorporates the model of Bayesian networks for uncertainty management along with the model of hyperedges. The necessity of this comes from the concept of separating the process of gathering, compressing, and filtering sensory data from the actual processes that reason using this data. This idea was emphasized by Fayek [28] when he made the claim that it is rare that a system addresses both the sensing and reasoning components. It is not sensing and reasoning that are separated but the declarative knowledge acquired from the gathering, compressing, and filtering of sensory data and the procedural knowledge that uses that information to perform a particular action. For example, the BAHG maintains knowledge of the surfaces and their respective polygonal borders and the hypothesized groupings among those surfaces and the environment modeling module acts upon this knowledge to create coherent models of the environment. This knowledge is particular to certain types of applications, object recognition, path planning, dynamic object tracking, and surface reconstruction and the environment modeling module can be replaced by appropriate modules to perform those particular applications. Other systems that have used Bayesian networks in the computer vision domain have maintained a Bayesian network separate from the declarative knowledge gathered from the data. When some action has to be performed on that data then the application module must query both the Bayesian network and the declarative knowledge.

One of the greatest advantages of using a BAHG is the organizational nature of the BAHG. The use of a BAHG network to specify the grouping process extracts that knowledge from being embedded in some code. Any declarative knowledge is maintained in the BAHG and procedural knowledge used for the formation of the BAHG is packaged as construction functions at each node. This is as valuable as the computation of belief values but has not been emphasized greatly in the literature. It is now possible to use the structure to determine quickly how the grouping process occurred.

7.3 Conclusions

What commenced as an investigation into a representation and approach for the removal of detail generally common in 3-D data resulted in the development of a framework for the management of uncertainty for the grouping of 3-D surfaces. Eventually the outcome of this was a representation, the BAHG, that allowed a declarative specification of the grouping procedure with procedures to construct the BAHG. Along with this was the introduction of compatibility functions that compute a quality of fit of the sensory data to hypothesized perceptual grouping formations from the sensory data. The collected use of these compatibility functions lead to the formations depicted for each hyperedge of the BAHG. To use the compatability functions as viable conditional probabilities they are first mapped to certainty values, between 0.0 and 1.0, using a declining S-curve that requires the specification of two controlling parameters γ and β. These values are set intuitively and for most compatibility functions are invariant to the quality of the sensory data, therefore allowing the use of the same values over several test cases. These certainty values are used directly or combined with other certainty values to compute conditional probabilities. The BAHG with their belief values can be used to reason about the creation of models of indoor environments.

7.4 Suggestions for Future Work

Perhaps the most valuable contributions in any thesis are the recognition of faults and fundamental limitations to approaches that can be improved with new research. The approach presented in this thesis, like other systems, has its limitations that need to be addressed. These problems can either be fundamental research directions or in some cases practical issues that have come up many times but have not been addressed properly.

7.4.1 Use of Compatibility Functions in a Causal Network

The original directions of this research investigated the use of compatibility functions in causal networks. Causal networks are intuitively a better way to think about the existence of particular formations caused by hypothesized models. It also allows the use of a priori knowledge about the models to be represented as conditional probabilities, i.e a top-down specification. The problem is the introduction of evidence into the network.

Causal networks require that this introduction be placed at the leaf nodes. if not then the network may become separated due to the concept of "d-separation" and evidence below the "d-separation" point is ignored. In some applications this is a positive feature but in this example the desire is to maintain the uncertainty throughout the grouping process. A causal network would then require the leaf nodes to represent perceptual grouping constraints like parallel. coplanar. etc...and the internal nodes again hypothesized formations.

As was presented in section 3.2.2. compatibility functions are not well suited for computing the conditional probabilities in causal networks. These limitations just emphasize the fact that compatibility functions measure evidence and are not a measure of a priori knowledge of the existence of a particular formation. Other approaches must be used and the simplest is to experimentally measure the frequency of the occurence of particular formations given the existence of their parent formations. Another approach is to experimentally determine a probability in the existence of a perceptual grouping given the existence of a parent formation for particular children formations. This has some similarities to the work by Levitt et al. [47] except that the notion of compatibility functions was not exploited there instead patterns among the features in a particular camera view was used. This combination of compatibility functions and the use of a prior conditional probabilities has not been investigated properly.

7.4.2 Bayesian Networks for the Segmentation of 3-D Range Data

This thesis relied on a completely separate procedure for the extraction of surfaces from the 3-D range data. This algorithm, developed by Boulanger [14]. also leverages Bayes theory of probability to determine which points would make adequate surfaces. As was mentioned in section 2.1.3. the core work in 3-D range data is in the formation of surfaces from this data. This is commonly referred to as range data segmentation and in itself is a form of perceptual grouping of 3-D points. The concept is to seek the surface that would have generated a particular 3-D point. This same principal has been applied to the detection of boundaries in intensity images [18]. Recent extensions to Boulanger's work [14] have included color as well as range into the segmentation procedure. Eventually a Bayesian network can be defined that combines multiple evidence and desired groupings into the segmentation procedure. This would have to be a recursive algorithm maximizing the most probable surface given the evidence. In this manner, the segmentation process can be driven by particular evidence instead of simply the best fitting points to a surface.

7.4.3 A Library of 3-D Compatibility Functions

There is a necessity for a collection of 3-D compatibility functions. To my knowledge no one has yet composed a review of these functions for 2-D edges let alone for 3-D surfaces and edges. These functions can be used by many algorithms for 3-D computer vision. Again, these functions appear in nearly all types of work in 3-D object recognition, tracking, scene analysis, and environment modeling but no attempt has been made to collect them into one reference source.

7.4.4 Uniform Declarative Representation

Throughout this thesis the attributed hypergraph and Bayesian attributed hypergraph have been presented as a uniform approach for the representation of declarative knowledge in the grouping of sensory data. This thesis has not been one of the first attempts in using attributed hypergraphs as the mechanism for unifying representations across several processes. Ironically there still does not exist a data base facility that allows the instantiation of persistent attributed hypergraphs. If such a facility existed knowledge could be more easily shared across multiple procedures as well as testing and developing could be greatly enhanced. This lack of an attributed hypergraph database could possibly be attributed to the fact that attributed hypergraphs have been used more for representing knowledge and generally the artificial intelligence community have not been active in the data base and information storage and retrieval domain until recently.

Object oriented techniques are ideally suited for the design of attributed hypergraphs and was used in the implementation of the BAHG. Even though attributed graphs are used extensively for the representation of knowledge in computer vision and other domains there has been a limited effort placed on the development of persistent attributed graphs and little or no effort into the development of a generic persistent hypergraph library.

7.4.5 Extracting Structure for Environment Modeling

The original ambition of this thesis was to formulate an approach that would allow a user to enter context based information on the detail wanted in the grouping process. It is a recognized fact that one of the problems with range data is that all the detail that is possible to record is recorded. This leads to more data than is desired. Only for manufactured parts is this detail desired. For environment modeling, shape and structure are generally more desirable in particular for the creation of models for virtual reality. In that domain it is possible to use inaccurate models overlayed with texture maps to give a realistic appearance. The difficulty is how to specify the desired granularity and also the segmentation of components from 3-D range data. This is not a problem of having loose tolerances in the segmentation procedure but it is more an issue of the detection of continuity of a surface and an understanding of the environment. Several examples can be presented that bring the point across.

- **The Brick Wall:** A brick wall is composed of a number of planar surfaces separated by channels. How can this brick wall be modeled as one complete flat surface ignoring the channels between the bricks?

- **The Door Handle:** A door is composed of a number of parts. the door itself and the door handle. Can the door and handle be extracted as separate components without having holes in the door?

- **Windows in a Wall:** This is a similar problem to that of the door handle. Can the wall be extracted from the sensory data without the windows?

All of these questions require contextual knowledge to determine how to decompose the data extracted from the sensor. The challenge is to develop a generic approach that can address these types of problems.

Bibliography

[1] K. M. Andress and A. C. Kak. Evidence accumulation and flow of control in a hierarchical spatial reasoning system. In *Advances in Spatial Reasoning*, volume 1, chapter 4, pages 133–179. Ablex Publishing Corporation, Norwood, NJ, 1990.

[2] J-L. Arseneault, R. Bergevin, and D. Laurendeau. Grouping of straight line segments and circular arcs for scene analysis. In *Vision Interface '94*, pages 137–144, 1994.

[3] N. Ayache. *Artificial vision for mobile robots: stereo vision and multisensory perception*. MIT Press, Cambridge, MA, 1991.

[4] M. Barry, D. Cyrluk, D. Kapur, J. L. Mundy, and V-D. Nguyen. A multi-level geometric reasoning system for vision. In *Geometric Reasoning*, pages 291–332. The MIT Press, Cambrigde, MA, 1989.

[5] I. A. Beinlich and E. H. Herskovits. Ergo: A graphical environment for constructing Bayesian belief networks. In *Uncertainty in Artificial Intelligence*, Cambridge, MA, 1990.

[6] P. J. Besl and R. C. Jain. Three-dimensional object recognition. *ACM Comput. Surveys*, 17(1):75–145, March 1985.

[7] P. J. Besl and R. C. Jain. Invariant surface characteristics for 3-D object recognition. *Computer vision, Graphics, and Image Processing*, 33:33–80, 1986.

[8] P. J. Besl. *Surfaces in range image understanding*. Springer Series in Perception Engineering. Springer-Verlag, Amsterdam, 1986.

[9] I. Biederman. Human image understanding: recent research and theory. *Computer Vision, Graphics, and Image Processing*, 32:29–73, 1985.

[10] I. Biederman. Human image understanding. In *7th Scandinavian Conference on Image Analysis*, late paper, Aalborg, Denmark, 1991.

[11] T. O. Binford, T. S. Levitt, and W. B. Mann. Bayesian inference in model-based vision. In *Uncertainty in Artificial Intelligence 3*, pages 73–95. Elsevier Science, Amsterdam, 1989.

[12] T. O. Binford. Infering surfaces from images. *Artif. Intell.*, 17:205–244, 1981.

[13] F. Blais, M. Rioux, and J. Domey. Optical range image acquisition for the navigation of a mobile robot. In *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, volume 3, pages 2574–2580, Sacramento, CA., April 9-11 1991.

[14] P. Boulanger. Hierarchical segmentation of range and color images based on Bayesian decision theory. In *Proceedings of the Fifteenth International Workshop on Maximum Entropy and Bayesian Methods*, Santa-Fe, NM., July 31- August 4 1995.

[15] P. Boulanger. Knowledge representation and analysis of range data. In *International Conference on Recent Advances in 3-D Digital Imaging and Modeling*, Tutorial Notes, Ottawa, Canada, May 12 1997.

[16] P. Boulanger and F. Blais. Range image segmentation. free space determination. and position estimate for a mobile vehicle. In *SPIE Proceedings. Mobile Robots VII.* volume 1831. pages 444–455, Boston. MA. November 18-20 1992.

[17] P. Boulanger and V. Moron. Automatic inspection of industrial parts using 3-D optical range sensor. In *Proceedings of the SME Workshop on Non-contact 3-D Gaging.* pages 15–16. Detroit. MI.. April 16-17 1996.

[18] N. Bryson and C. J. Taylor. Boundary detection using Bayesian nets. *Image and Vision Computing.* 10(5):308–312. June 1992.

[19] C. Chen and A. C. Kak. A robot vision system for recognizing 3-D objects in low order polynomial time. *IEEE Trans. on Systems. Man. and Cybernetics.* 19(6):1535–1563. Nov/Dec 1989.

[20] E. Cox. *The fuzzy systems handbook.* AP Professional. 1994.

[21] F. Cozman. *JavaBayes Version 0.31: Bayesian Networks in Java.* Carnegie Mellon University. 1996.

[22] J. Dolan and R. Weiss. Perceptual grouping of curved lines. In *DARPA Image Understanding Workshop.* pages 1135–1145. Palo Alto. CA. May 1989.

[23] H. F. Durrant-Whyte. Uncertain geometry. In *Geometric Reasoning.* pages 447–481. A Bradford Book The MIT Press. Cambrigde. MA. 1989.

[24] A. M. Earnshaw. *3-D object synthesis in a robotic workcell.* Phd thesis. University of Waterloo, Waterloo, Ontario, Canada, 1991.

[25] S. Elgazzar. R. Liscano. F. Blais, and A. Miles. 3-D data acquisition for indoor environment modeling using a compact active range sensor. In *IEEE Instrumenta-*

tion. Measurement and Technology Conference (IMTC '97). pages 586–592. Otawa. Ontario. May 19-21 1997.

[26] T-J. Fan. G. Medioni. and R. Nevatia. Recognizing 3-D objects using surface descriptions. *IEEE Trans. on Pattern Analysis and Machine Intelligence*. 11(11):1140–1157. November 1989.

[27] O. D. Faugeras and M. Herbert. The representation. recognition. and locating of 3-D objects. *The Int. Journal of Robotics Research*. 5(3):27–52. Fall 1986.

[28] R. Fayek. *3-D Surface modeling using hierarchical topographic triangular meshes*. PhD thesis. University of Waterloo. Waterloo. Ontario. Canada. 1996.

[29] P. J. Flynn and A. K. Jain. Bonsai: 3-D object recognition using constraint search. *IEEE Trans. on Pattern Analysis and Machine Intelligence*. 13(10):1066–1074. October 1991.

[30] G. Foresti. V. Murino. C. S. Regazzoni. and G. Vernazza. A distributed approach to 3-D read scene recognition. *IEEE Trans. on Vehicular Technology*. 43(2):389–406. May 1994.

[31] Q-G. Gao and A. K. C. Wong. Curve detection based on perceptual organization. *Pattern Recognition*. 26(7):1039–1046. 1993.

[32] S. Gong. Bayesian net for functional integration of visual surveillance system. In *PRICAI94: 3rd Pacific Rim Int. Conf. on Artificial Intelligence*. volume 2. pages 850–856. Beijing. China. August 15-18 1994.

[33] O. Grau. A scene analysis system for the generation of 3-D models. In *International Conference on Recent Advances in 3-D Digital Imaging and Modeling*. pages 221–228. Ottawa, Canada, May 12-15 1997.

[34] W. E. L. Grimson. The combinatorics of object recognition in cluttered environments using constraint search. *Artificial Intelligence*. 44:121–165. 1990.

[35] J. Henokoff and L. Shapiro. Interesting patterns for model-based machine vision. In *IEEE Third International Conference on Computer Vision*, pages 535–538. Osaka. Japan. December 4-7 1990.

[36] J. Hochberg. Higher order stimuli and interresponse coupling in the perception of the visual world. In *Perception: Essays in Honour of James J. Gibson*. Cornell University Press. Ithaca. New York. 1974.

[37] J. Hochberg and V. Brooks. The psychophysics of form: reversible perspective drawings of spatial objects. *Am. J. of Psychology*. 73:337–354. 1960.

[38] R. Horaud. F. Veillon. and T. Skordas. Finding geometric and relational structures in am image. In *European Conf. on Computer Vision*, pages 374–384. Antibes. France. April 1990.

[39] J. Hoshino. T. Uemura. and I. Masuda. Region-based reconstruction of an indoor scene using an intergration of active and passive sensing techniques. In *IEEE Third International Conference on Computer Vision*. pages 568–572. Osaka. Japan. December 4-7 1990.

[40] S. A. Hutchinson. R. L. Cromwell. and A. C. Kak. Applying uncertainty reasoning to model based object recognition. In *IEEE Computer Vision and Pattern Recognition*, pages 541–548, San Diego, CA, July 4-8 1989.

[41] F. V. Jensen. B. Chamberlain, T. Nordahl, and F. Jensen. Analysis in hugin of data conflict. In *Uncertainty in Artificial Intelligence 6*, pages 519–528. North-Holland. Amsterdam. 1991.

[42] F. V. Jensen. *An introduction to Bayesian networks.* Springer-Verlag. New York. 1996.

[43] F. V. Jensen. H. I. Christensen. and J. Nielsen. Bayesian methods for interpretation and control in multi-agent vision systems. In *Proceedings of SPIE: Applications in Artificial Intelligence X: Machine Vision and Robotics*, volume 1708, pages 536–548, Orlando. FL. 1992.

[44] A. E. Johnson. R. Hoffman. J. Osborn. and M. Hebert. A system for semi-automatic modeling of complex environments. In *International Conference on Recent Advances in 3-D Digital Imaging and Modeling*, pages 213–220. Ottawa. Canada. May 12-15 1997.

[45] H-B. Kang and E. L. Walker. Perceptual grouping based on fuzzy sets. In *IEEE Int. Conf. on Fuzzy Systems*, pages 651–659. San Diego. CA. March 8-12 1992.

[46] F. Labonte. Applying the gestalt laws of perceptual grouping to 3-D image data. 4th Year Honours Thesis. Dept. of Psychology. Carleton University. Ottawa. Canada. April 1996.

[47] T. Levitt. T. Binford. and G. Ettinger. Utility-based control for computer vision. In *Uncertainty in Aritificial Intelligence 4*, pages 407–421. Amsterdam. 1990.

[48] T. S. Levitt. Model-based probabilistic inference in hierarchical hypothesis spaces. In *Uncertainty in Artificial Intelligence*, pages 347–356. Elsevier Science. Amsterdam. 1986.

[49] J. Liang. F. C. Jensen. and H. I. Christensen. A framework for generic object recognition with Bayesian networks. In *Intelligent Industrial Automation. IIA '96*, pages C9–15. Reading. UK. March 26-28 1996.

[50] R. Liscano, S. Elgazzar, and A. K. C. Wong. A proximity compatibility function among 3-D surfaces for environment modelling. In *5th IASTED International Conference on Robotics and Manufacturing*, pages 161–166, Cancun, Mexico, May 19-21 1997.

[51] R. Liscano, R. Fayek, A. Manz, E. Stuck, and J-Y. Tigli. Using a blackboard to integrate multiple activities and achieve strategic reasoning for mobile-robot navigation. *IEEE Expert*, 10(2):24–36, 1995.

[52] W. Liu, L. Li, and A. K. C. Wong. 3-D compound object representation using attrtibuted hypergraph. In *4th University of New Brunswick Artificial Intelligence Symposium*, pages 445–455, Fredericton, Canada, September 1991.

[53] D. G. Lowe. *Perceptual organization and visual recognition*. The Kluwer Int. Series in Eng. and Comp. Science. Kluwer Academic Publishers, 1985.

[54] D. G. Lowe and T. O. Binford. Segmentation and aggregation: an approach to figure ground phenomena. In *Proc. of DARPA Image Understanding Workshop*, pages 168–178, Palo Alto, CA, 1982.

[55] H. Q. Lu and J. K. Aggarwal. Applying perceptual organization to the detection of man-made objects in non-urban scenes. *Pattern Recognition*, 25(8):835–853, August 1992.

[56] D. Marr. *Vision*. W. H. Freeman, New York, 1982.

[57] J. D. McCafferty. *Human and machine vision: computing perceptual organisation*. Ellis Horwood Series in Digital and Signal Processing. Ellis Horwood, 1990.

[58] R. Mohan and R. Nevatia. Using perceptual organization to extract 3-D structures. *IEEE Trans. on Pattern Analysis and Machine Intelligence*. 11(11):1121–1139. November 1989.

[59] R. Munk-Fairwood. Recognition of geometric primitives using logic program and probabilistic network reasoning methods. In *Proceedings of SPIE: Applications in AI X: Machine Vision and Robotics*. volume 1708. pages 589–600. Orlando. FL. 1992.

[60] F. Nashashibi and M. Devy. 3-D incremental modeling and robot localization in a structured environment using a laser range finder. In *Proceedings of the 1993 IEEE International Conference on Robotics and Automation*. pages 20–27. Atlanta. GA. May 1993.

[61] J. Pearl. *Probabilistc reasoning in intelligent systems: networks of plausible inference*. Representation and Reasoning. Morgan Kaufman Publishers Inc.. San Mateo. CA. 1988.

[62] J. Princen. J. Illingworth. and J. Kittler. A hierarchical approach to line extraction. In *Computer Vision and Pattern Recognition*. pages 92–97. San Diego. California. June 4-8 1989.

[63] M. Rioux. G. Godin. P. Boulanger. G. Roth, and F. Blais. Electronic imaging of a 3-D shapes and its potential for the factory of the future. In *Proceedings of the 10th ISPE/IFAC International Conference on CAD/CAM. Robotics and Factories of the Future*. pages 691–698. Ottawa. Canada. August 21-24 1994.

[64] I. Rock. *An Introduction to Perception*. Macmillan Publishing Company. New York, 1975.

[65] E. Rosch. Principles of categorization. In *Cognition and categorization*, pages 27–48. Halsted Press, 1978.

[66] E. Rosch and C. B. Mervis. Family resemblances: studies in the internal structure categories. *Cognitive Pshychology*, 7:573–605, 1975.

[67] E. Rosch, C. B. Mervis, W. J. D. Gray, and P. R. Boyes. Basic objects in natural categories. *Cognitive Pshychology*, 8:382–439, 1976.

[68] S. Sarkar and K. L. Boyer. Integration, inference, and management of spatial information using Bayesian networks: perceptual organization. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(3):256–274, March 1993.

[69] R. Seidel. A simple and fast randomized algorithm for computing trapezoidal decompositions and for triangulating polygons. *Computational Geometry: Theory and Applications*, 1(1):51-64, July 1991.

[70] V. Sequeira, Joao G. M. Goncalves, and M. Isabel Ribeiro. 3-D scene modelling from multiple views. In *Proceedings of the SPIE: Videometrics IV*, volume 2598, pages 114–127, Philadephia, Pa., October 25-26 1995.

[71] C. G. Small. Techniques of shape analysis on sets of points. *International Statistical Review*, 56(3):243–257, 1988.

[72] F. Stein and G. Medioni. Structural indexing: efficient 3-D object recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 14(2):125–145, February 1992.

[73] M. Suk and S. M. Bhandarkar. *Three-dimensional object recognition from range images*. Computer Science Workbench. Springer-Verlag, New York, 1992.

[74] B. Tversky and K. Hemenway. Objects, parts, and categories. *Journal of Experimental Psychology*, 113(2):169–190, 1984.

[75] J. Wagemans, L. Van Gool, V. Swinnen, and J. Van Horebeek. Higher-order structure in regularity detection. *Vision Res.*, 33(8):1067–1088, 1993.

[76] E. L. Walker and M. Herman. Geometric reasoning for contructing 3-D scene descriptions from images. In *Geometric Reasoning*, pages 275–290. A Bradford Book The MIT Press, Cambridge, MA, 1989.

[77] S-H. Wang and Q-G. Gao. High-level decisions in belief networks based 3-D object recognition. In *IEEE Int. Conf. on Systems, Man, and Cybernetics*, volume 3, pages 2742–2747, Vancover, Canada, October 1995.

[78] R. Weiss and M. Boldt. Geometric grouping applied to straight lines. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 489–495, Miami Beach, FL, June 1986.

[79] Andrew P. Witkin and Jay M. Tenenbaum. On the role of structure in vision. In *Human and Machine Vision*, volume 8 of *Notes and Reports in Computer Science and Applied Mathematics*, pages 481–543. Academic Press, New York, 1983.

[80] A. K. C. Wong and W. Liu. Hypergraph representation for 3-D object model synthesis and scene interpretation. In *IARP The 2nd Workshop on Sensor Fusion and Environmental Modelling.*, pages 1–19, Oxford, UK, September 2-5 1991.

[81] A. K. C. Wong, S. W. Lu, and M. Riuox. Recognition and shape synthesis of 3-D objects based on attributed hypergraphs. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 11(3):279–290, March 1989.

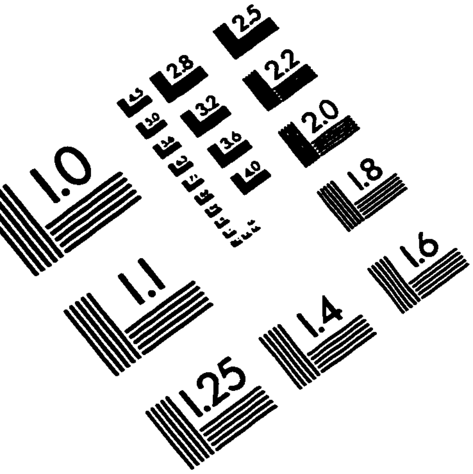
[82] E. K. Wong. Model matching in robot vision by subgraph isomorphism. *Pattern Recognition*, 25(3):287–303, March 1992.

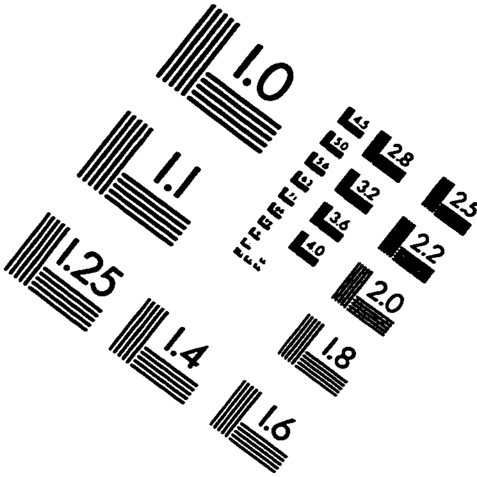
[83] K. C. Wong, J. Kittler, and J. Illingworth. Heuristically guided polygon finding. In *British Machine Vision Conference*, pages 400–407, Glasgow, UK, September 24-26 1991.

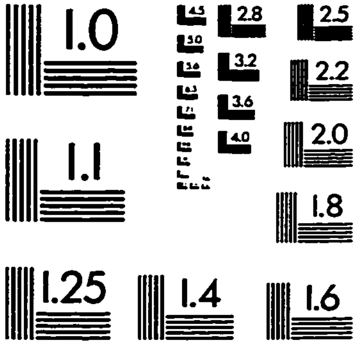
[84] N. L. Zhang and D. Poole. Exploiting causal independence in Bayesian network inference. *Journal of Artificial Intelligence Research*, pages 301–28, 1996.

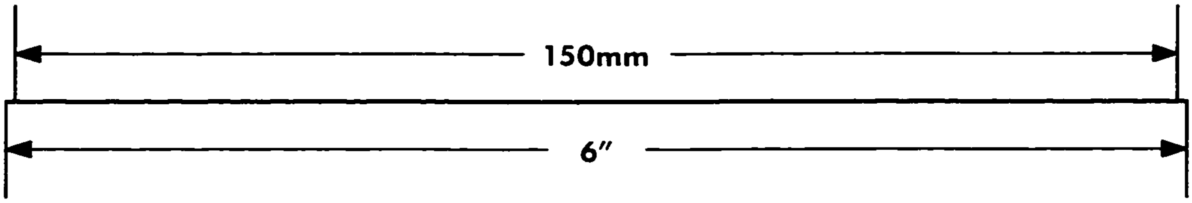
[85] Z. Zhang and O. D. Faugeras. *3-D Dynamic Scene Analysis: A Stereo Based Approach*. Springer Series in Information Sciences. Springer-Verlag, Berlin, 1992.

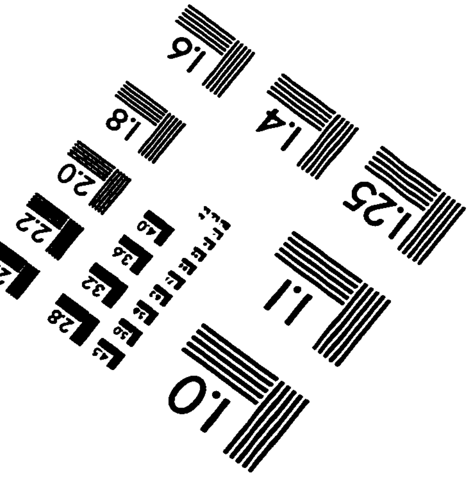
IMAGE EVALUATION TEST TARGET (QA-3)

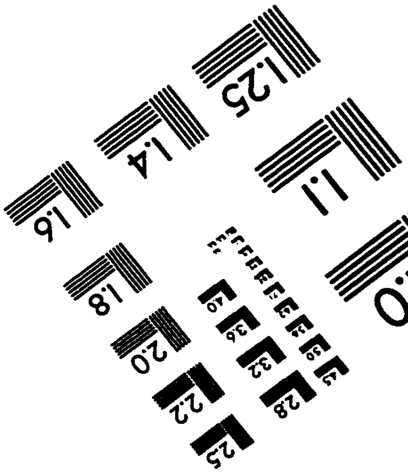












APPLIED IMAGE, Inc
1653 East Main Street
Rochester, NY 14609 USA
Phone: 716/482-0300
Fax: 716/288-5989