NOTICE WARNING CONCERNING COPYRIGHT RESTRICTIONS:
The copyright law of the United States (title 17, U.S. Code) governs the making
of photocopies or other reproductions of copyrighted material. Any copying of this
document without permission of its author may be prohibited by law.
AN OBJECT-CENTERED THREE-DIMENSIONAL
MODEL BUILDER

CLAYTON ALBERT DANE III

A DISSERTATION
in
Computer and Information Science

Presented to the Graduate Faculties of the University
Pennsylvania in partial fulfillment of the
requirements for the degree of Doctor of Philosophy.

1982
ABSTRACT

AN OBJECT-CENTERED THREE-DIMENSIONAL MODEL BUILDER

CLAYTON DANE

SUPERVISOR: DR. RUZENA BAJCSY

A method of building a three-dimensional model of a rigid object using information from many views is described. Planar and quadric surface primitives describe the object's surface in an object-centered reference frame. The extent of a primitive is defined by the intersection of the primitive with its neighbors. An edge graph defined by these intersections implicitly expresses spatial relationships between surface primitives.

The model builder's input consists of groups of data points corresponding to different views. Each data point contains spatial and orientation information about the object's surface at a discrete location. A set of registered arrays is used to summarize input information in local areas. Mathematical principles from differential geometry are applied to determine local surface properties. A region-growing technique is applied to this information to identify data points which are then represented by a surface.
primitive. Edges and corners are computed based on intersections of surface primitives. The results from analysis of the various views are transformed to a common arbitrary reference frame for integration into a global model. The final object-centered reference frame established based on the center of gravity and moments of inertia of the object as determined from the complete model.

The goal of model building has applications in fields of pattern recognition, computer vision, robotics, computer-aided design and computer-aided manufacturing. A model using surface primitives appears as a natural fi rst step in describing an object because surfaces are obvious visual features. The strengths and weaknesses of the surface model are explored.

Keywords:

AN OBJECT-CENTERED THREE-DIMENSIONAL MODEL BUILDER

CLAYTON ALBERT DANE III

A DISSERTATION

in

Computer and Information Science

Presented to the Graduate Faculties of the University of Pennsylvania in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

1982

------------------------
Supervisor of Dissertation

------------------------
Graduate Group Chairperson
ACKNOWLEDGEMENT

This work is a reflection of me and indirectly a reflection on the world I live in. Without the understanding and contributions of the people in that world, I could not exist. I stood on the shoulders of giants for support while doing this work. For fear of failing to acknowledge everyone justly, I offer to all a collective, simple thank you.

This work was funded in part by grants from the National Science Foundation and the National Institutes of Health.
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>INTRODUCTION</td>
</tr>
<tr>
<td>0</td>
<td>BACKGROUND</td>
</tr>
<tr>
<td>.1</td>
<td>Data Acquisition</td>
</tr>
<tr>
<td>1.1</td>
<td>Orientation Information</td>
</tr>
<tr>
<td>1.2</td>
<td>Spatial Information</td>
</tr>
<tr>
<td>1.3</td>
<td>Object Representation</td>
</tr>
<tr>
<td>1.4</td>
<td>Knowledge Driven Systems</td>
</tr>
<tr>
<td>0.1</td>
<td>A SURFACE MODEL</td>
</tr>
<tr>
<td>.1</td>
<td>Surface Primitives</td>
</tr>
<tr>
<td>.2</td>
<td>The Edge Graph</td>
</tr>
<tr>
<td>.3</td>
<td>The Model Structure</td>
</tr>
<tr>
<td>.4</td>
<td>An Example</td>
</tr>
<tr>
<td>0.2</td>
<td>BUILDING A SURFACE MODEL</td>
</tr>
<tr>
<td>.1</td>
<td>Input Data</td>
</tr>
<tr>
<td>.1.1</td>
<td>Sources Of Data</td>
</tr>
<tr>
<td>.1.2</td>
<td>A View</td>
</tr>
<tr>
<td>.2</td>
<td>Analysis Of A Local View</td>
</tr>
<tr>
<td>.2.1</td>
<td>Determining Surface Primitives</td>
</tr>
<tr>
<td>2.1.1</td>
<td>Structuring Input Data</td>
</tr>
<tr>
<td>.2.1.2</td>
<td>Types Of Properties</td>
</tr>
<tr>
<td>.2.1.3</td>
<td>Growing Groups Of Data Points</td>
</tr>
<tr>
<td>.2.1.3.1</td>
<td>Quadric Surfaces</td>
</tr>
<tr>
<td>.2.1.3.1.1</td>
<td>Orientation Continuity</td>
</tr>
<tr>
<td>.2.1.3.1.2</td>
<td>Depth Continuity</td>
</tr>
<tr>
<td>.2.1.3.2</td>
<td>Planar Surfaces</td>
</tr>
<tr>
<td>.2.1.4</td>
<td>Surface Fitting</td>
</tr>
<tr>
<td>.2.1.4.1</td>
<td>Fitting Criteria</td>
</tr>
<tr>
<td>.2.1.4.1.1</td>
<td>Spatial Criterion</td>
</tr>
<tr>
<td>.2.1.4.1.2</td>
<td>Orientation Criterion</td>
</tr>
<tr>
<td>.2.1.4.2</td>
<td>Minimization</td>
</tr>
<tr>
<td>.2.2</td>
<td>The Local Edge Graph</td>
</tr>
<tr>
<td>.3</td>
<td>View Integration</td>
</tr>
<tr>
<td>.3.1</td>
<td>Transformation To A Common Coordinate System</td>
</tr>
<tr>
<td>.3.2</td>
<td>Surface Identification</td>
</tr>
<tr>
<td>.3.2.1</td>
<td>The Types Of Surface</td>
</tr>
<tr>
<td>.3.2.2</td>
<td>Measure Of Surfaces' Similarity</td>
</tr>
<tr>
<td>.3.2.3</td>
<td>The Role Of The Edge Graph</td>
</tr>
<tr>
<td>.3.3</td>
<td>Description Updating</td>
</tr>
<tr>
<td>.4</td>
<td>The Final Reference Frame</td>
</tr>
</tbody>
</table>
IMPLEMENTATION

Input Data

Sources Of Data

Artificial Data

Real Data

A View

Analysis Of A Local View

Determining Surface Primitives

Structuring Input Data

Types Of Properties

Growing Groups Of Data Points

Quadric Surfaces

Orientation Continuity

Depth Discontinuity

Planar Surfaces

Surface Fitting

Fitting Criteria

Minimization

The Wrong Point In The Right Place

The Right Point In The Wrong Place

A Substitute For Better Resolution

Transformation To A Common Coordinate System

Surface Identification

Description Updating

The Final Reference Frame

RESULTS

Presentation Of Input Data

Presentation Of Surface Primitives

Test Cases

Flattened Sphere

Cube And Ellipsoid

Cylinder And Sphere

Cylinder And Negative Sphere

Cube And Negative Cylinders

Ice Cream Cone

Barbell

Mug

Telephone Handset

CONCLUSIONS AND FUTURE WORK

Specific Conclusions

General Conclusions

Future Work

INDEX

REFERENCES
4-1 / Observed Properties

4-2 / Derived Properties
# TABLE OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>A View of an Object</td>
<td>34</td>
</tr>
<tr>
<td>-2</td>
<td>Model Structure</td>
<td>35</td>
</tr>
<tr>
<td>-1</td>
<td>Local Coordinate System</td>
<td>42</td>
</tr>
<tr>
<td>-2</td>
<td>Similar Surface Orientation</td>
<td>50</td>
</tr>
<tr>
<td>-3</td>
<td>Similar Spatial Location</td>
<td>50</td>
</tr>
<tr>
<td>-4</td>
<td>A Study in Curvature</td>
<td>52</td>
</tr>
<tr>
<td>-5</td>
<td>Decision Space</td>
<td>65</td>
</tr>
</tbody>
</table>

**Latten Sphere**

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.3.1-1</td>
<td>View 1 / Shaded Image</td>
<td>93</td>
</tr>
<tr>
<td>6.3.1-2</td>
<td>View 1 / Input Data</td>
<td>94</td>
</tr>
<tr>
<td>6.3.1-3</td>
<td>View 1 / Surface 1</td>
<td>94</td>
</tr>
<tr>
<td>6.3.1-4</td>
<td>View 1 / Surface 2</td>
<td>95</td>
</tr>
<tr>
<td>6.3.1-5</td>
<td>View 1 / Surface 3</td>
<td>95</td>
</tr>
<tr>
<td>6.3.1-6</td>
<td>View 2 / Shaded Image</td>
<td>96</td>
</tr>
<tr>
<td>6.3.1-7</td>
<td>View 2 / Input Data</td>
<td>96</td>
</tr>
<tr>
<td>6.3.1-8</td>
<td>View 2 / Surface 1</td>
<td>97</td>
</tr>
<tr>
<td>6.3.1-9</td>
<td>View 2 / Surface 2</td>
<td>97</td>
</tr>
<tr>
<td>6.3.1-10</td>
<td>View 2 / Surface 3</td>
<td>98</td>
</tr>
<tr>
<td>6.3.1-11</td>
<td>View 3 / Shaded Image</td>
<td>98</td>
</tr>
<tr>
<td>6.3.1-12</td>
<td>View 3 / Input Data</td>
<td>99</td>
</tr>
<tr>
<td>6.3.1-13</td>
<td>View 3 / Surface 1</td>
<td>99</td>
</tr>
<tr>
<td>6.3.1-14</td>
<td>View 4 / Shaded Image</td>
<td>100</td>
</tr>
<tr>
<td>6.3.1-15</td>
<td>View 4 / Input Data</td>
<td>100</td>
</tr>
<tr>
<td>6.3.1-16</td>
<td>View 4 / Surface 1</td>
<td>101</td>
</tr>
</tbody>
</table>

**Ibe and Ellipsoid**

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.3.2-1</td>
<td>View 1 / Shaded Image</td>
<td>103</td>
</tr>
<tr>
<td>6.3.2-2</td>
<td>View 1 / Input Data</td>
<td>103</td>
</tr>
<tr>
<td>6.3.2-3</td>
<td>View 1 / Surface 1</td>
<td>104</td>
</tr>
<tr>
<td>6.3.2-4</td>
<td>View 1 / Surface 2</td>
<td>104</td>
</tr>
<tr>
<td>6.3.2-5</td>
<td>View 2 / Shaded Image</td>
<td>105</td>
</tr>
<tr>
<td>6.3.2-6</td>
<td>View 2 / Input Data</td>
<td>105</td>
</tr>
<tr>
<td>6.3.2-7</td>
<td>View 2 / Surface 1</td>
<td>106</td>
</tr>
<tr>
<td>6.3.2-8</td>
<td>View 2 / Surface 2</td>
<td>106</td>
</tr>
<tr>
<td>6.3.2-9</td>
<td>View 2 / Surface 3</td>
<td>107</td>
</tr>
<tr>
<td>6.3.2-10</td>
<td>View 3 / Shaded Image</td>
<td>107</td>
</tr>
<tr>
<td>6.3.2-11</td>
<td>View 3 / Input Data</td>
<td>108</td>
</tr>
<tr>
<td>6.3.2-12</td>
<td>View 3 / Surface 1</td>
<td>108</td>
</tr>
<tr>
<td>6.3.2-13</td>
<td>View 3 / Surface 2</td>
<td>109</td>
</tr>
<tr>
<td>6.3.2-14</td>
<td>View 3 / Surface 3</td>
<td>109</td>
</tr>
<tr>
<td>6.3.2-15</td>
<td>View 3 / Unused Data</td>
<td>110</td>
</tr>
<tr>
<td>6.3.2-16</td>
<td>View 4 / Shaded Image</td>
<td>110</td>
</tr>
<tr>
<td>6.3.2-17</td>
<td>View 4 / Input Data</td>
<td>111</td>
</tr>
<tr>
<td>6.3.2-18</td>
<td>View 4 / Surface 1</td>
<td>111</td>
</tr>
<tr>
<td>6.3.2-19</td>
<td>View 4 / Surface 2</td>
<td>112</td>
</tr>
<tr>
<td>6.3.2-20</td>
<td>View 4 / Surface 3</td>
<td>112</td>
</tr>
</tbody>
</table>
Cylinder and Sphere

6.3.3-1 / View 1 / Shaded Image
6.3.3-2 / View 1 / Input Data
6.3.3-3 / View 1 / Surface 1
6.3.3-4 / View 2 / Shaded Image
6.3.3-5 / View 2 / Input Data
6.3.3-6 / View 2 / Surface 1
6.3.3-7 / View 2 / Surface 2
6.3.3-8 / View 2 / Surface 3
6.3.3-9 / View 3 / Shaded Image
6.3.3-10 / View 3 / Input Data
6.3.3-11 / View 3 / Surface 1
6.3.3-12 / View 3 / Surface 2
6.3.3-13 / View 3 / Surface 3
6.3.3-14 / View 3 / Surface 4
6.3.3-15 / View 4 / Shaded Image
6.3.3-16 / View 4 / Input Data
6.3.3-17 / View 4 / Surface 1
6.3.3-18 / View 4 / Surface 2
6.3.3-19 / View 4 / Surface 3

Cylinder and Negative Sphere

6.3.4-1 / View 1 / Shaded Image
6.3.4-2 / View 1 / Input Data
6.3.4-3 / View 1 / Surface 1
6.3.4-4 / View 1 / Surface 2
6.3.4-5 / View 2 / Shaded Image
6.3.4-6 / View 2 / Input Data
6.3.4-7 / View 2 / Surface 1
6.3.4-8 / View 2 / Surface 2
6.3.4-9 / View 3 / Shaded Image
6.3.4-10 / View 3 / Input Data
6.3.4-11 / View 3 / Surface 1
6.3.4-12 / View 3 / Surface 2
6.3.4-13 / View 4 / Shaded Image
6.3.4-14 / View 4 / Input Data
6.3.4-15 / View 4 / Surface 1
6.3.4-16 / View 4 / Unused Data

Cube and Negative Cylinders

6.3.5-1 / View 1 / Shaded Image
6.3.5-2 / View 1 / Input Data
6.3.5-3 / View 1 / Surface 1
6.3.5-4 / View 1 / Surface 2
6.3.5-5 / View 1 / Surface 3
6.3.5-6 / View 1 / Surface 4
6.3.5-7 / View 2 / Shaded Image
6.3.5-8 / View 2 / Input Data
6.3.5-9 / View 2 / Surface 1
<table>
<thead>
<tr>
<th>Section</th>
<th>View</th>
<th>Image Type</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.3.5-10</td>
<td>View 2</td>
<td>Surface 2</td>
<td>139</td>
</tr>
<tr>
<td>6.3.5-11</td>
<td>View 2</td>
<td>Surface 3</td>
<td>140</td>
</tr>
<tr>
<td>6.3.5-12</td>
<td>View 3</td>
<td>Shaded Image</td>
<td>140</td>
</tr>
<tr>
<td>6.3.5-13</td>
<td>View 3</td>
<td>Input Data</td>
<td>141</td>
</tr>
<tr>
<td>6.3.5-14</td>
<td>View 3</td>
<td>Surface 1</td>
<td>141</td>
</tr>
<tr>
<td>6.3.5-15</td>
<td>View 3</td>
<td>Surface 2</td>
<td>142</td>
</tr>
<tr>
<td>6.3.5-16</td>
<td>View 3</td>
<td>Surface 3</td>
<td>142</td>
</tr>
<tr>
<td>6.3.5-17</td>
<td>View 3</td>
<td>Unused Data</td>
<td>143</td>
</tr>
<tr>
<td>6.3.5-18</td>
<td>View 4</td>
<td>Shaded Image</td>
<td>143</td>
</tr>
<tr>
<td>6.3.5-19</td>
<td>View 4</td>
<td>Input Data</td>
<td>144</td>
</tr>
<tr>
<td>6.3.5-20</td>
<td>View 4</td>
<td>Surface 1</td>
<td>144</td>
</tr>
<tr>
<td>6.3.5-21</td>
<td>View 4</td>
<td>Surface 2</td>
<td>145</td>
</tr>
<tr>
<td>6.3.5-22</td>
<td>View 4</td>
<td>Surface 3</td>
<td>145</td>
</tr>
<tr>
<td>6.3.5-23</td>
<td>View 4</td>
<td>Unused Data</td>
<td>146</td>
</tr>
<tr>
<td>6.3.6-1</td>
<td>View 1</td>
<td>Shaded Image</td>
<td>147</td>
</tr>
<tr>
<td>6.3.6-2</td>
<td>View 1</td>
<td>Input Data</td>
<td>148</td>
</tr>
<tr>
<td>6.3.6-3</td>
<td>View 1</td>
<td>Surface 1</td>
<td>148</td>
</tr>
<tr>
<td>6.3.6-4</td>
<td>View 1</td>
<td>Surface 2</td>
<td>149</td>
</tr>
<tr>
<td>6.3.6-5</td>
<td>View 2</td>
<td>Shaded Image</td>
<td>149</td>
</tr>
<tr>
<td>6.3.6-6</td>
<td>View 2</td>
<td>Input Data</td>
<td>150</td>
</tr>
<tr>
<td>6.3.6-7</td>
<td>View 2</td>
<td>Surface 1</td>
<td>150</td>
</tr>
<tr>
<td>6.3.6-8</td>
<td>View 2</td>
<td>Surface 2</td>
<td>151</td>
</tr>
<tr>
<td>6.3.6-9</td>
<td>View 3</td>
<td>Shaded Image</td>
<td>151</td>
</tr>
<tr>
<td>6.3.6-10</td>
<td>View 3</td>
<td>Input Data</td>
<td>152</td>
</tr>
<tr>
<td>6.3.6-11</td>
<td>View 3</td>
<td>Surface 1</td>
<td>152</td>
</tr>
<tr>
<td>6.3.6-12</td>
<td>View 3</td>
<td>Surface 2</td>
<td>153</td>
</tr>
<tr>
<td>6.3.6-13</td>
<td>View 4</td>
<td>Shaded Image</td>
<td>153</td>
</tr>
<tr>
<td>6.3.6-14</td>
<td>View 4</td>
<td>Input Data</td>
<td>154</td>
</tr>
<tr>
<td>6.3.6-15</td>
<td>View 4</td>
<td>Surface 1</td>
<td>154</td>
</tr>
<tr>
<td>6.3.6-16</td>
<td>View 4</td>
<td>Surface 2</td>
<td>155</td>
</tr>
<tr>
<td>6.3.7-1</td>
<td>View 1</td>
<td>Shaded Image</td>
<td>156</td>
</tr>
<tr>
<td>6.3.7-2</td>
<td>View 1</td>
<td>Input Data</td>
<td>157</td>
</tr>
<tr>
<td>6.3.7-3</td>
<td>View 1</td>
<td>Surface 1</td>
<td>157</td>
</tr>
<tr>
<td>6.3.7-4</td>
<td>View 1</td>
<td>Surface 2</td>
<td>158</td>
</tr>
<tr>
<td>6.3.7-5</td>
<td>View 1</td>
<td>Surface 3</td>
<td>158</td>
</tr>
<tr>
<td>6.3.7-6</td>
<td>View 2</td>
<td>Shaded Image</td>
<td>159</td>
</tr>
<tr>
<td>6.3.7-7</td>
<td>View 2</td>
<td>Input Data</td>
<td>159</td>
</tr>
<tr>
<td>6.3.7-8</td>
<td>View 2</td>
<td>Surface 1</td>
<td>160</td>
</tr>
<tr>
<td>6.3.7-9</td>
<td>View 2</td>
<td>Surface 2</td>
<td>160</td>
</tr>
<tr>
<td>6.3.7-10</td>
<td>View 2</td>
<td>Surface 3</td>
<td>160</td>
</tr>
<tr>
<td>6.3.7-11</td>
<td>View 3</td>
<td>Shaded Image</td>
<td>161</td>
</tr>
<tr>
<td>6.3.7-12</td>
<td>View 3</td>
<td>Input Data</td>
<td>162</td>
</tr>
<tr>
<td>6.3.7-13</td>
<td>View 3</td>
<td>Surface 1</td>
<td>162</td>
</tr>
<tr>
<td>6.3.7-14</td>
<td>View 3</td>
<td>Surface 2</td>
<td>163</td>
</tr>
<tr>
<td>6.3.7-15</td>
<td>View 3</td>
<td>Surface 3</td>
<td>163</td>
</tr>
<tr>
<td>6.3.7-16</td>
<td>View 4</td>
<td>Shaded Image</td>
<td>164</td>
</tr>
<tr>
<td>6.3.7-17</td>
<td>View 4</td>
<td>Input Data</td>
<td>164</td>
</tr>
</tbody>
</table>
CHAPTER ONE

INTRODUCTION

A method of building a model of a rigid object is described. It utilizes information from many views covering the complete surface of the object. The model is three-dimensional in nature and is expressed in an object-centered reference frame. Planar and quadric surface primitives are used in conjunction with an edge graph to describe the object's surface structure. There are other ways to represent an object. However, a surface model appears as a natural first description because surfaces are the most obvious visual features. The goal of modeling has applications in the fields of pattern recognition, computer vision, robotics, computer-aided design and computer-aided manufacturing. The strengths and weaknesses of this surface model are explored.

The purpose of a model is to organize or structure information to facilitate the solution of a problem. There are many varieties of models from which to choose when considering three-dimensional objects. At present, there is no universally "best" model for representation of three-dimensional object. Models are divided into the
There are many aspects of representation which can affect the solution to a problem. If a surface or volume representation uses primitives, the number of different primitives and their properties can affect the usefulness of the representation. Consider the following two volume primitives. The first primitive is a simple sphere. It is rotationally invariant and has been used successfully to model the human body [O'ROURKE/BADLER79]. Algorithms for manipulating a model of spheres are relatively simple because all the instances of primitives can be handled the same way. A single primitive can be a weakness, also, for
A planar object is difficult to represent with spheres, as there are no alternatives in this representation. In contrast, the use of the class of generalized cylinders as primitives permits a large variety of volumes to be expressed. However, algorithms dealing with these primitives are more complex because of the added variety and increased complexity of individual primitives. Another aspect of representation involves the method used to compose the object into its primitive parts. Is the result of the decomposition unique? Are the primitives lique? Are they permitted to overlap? How are the boundaries defined? Are they implicitly or explicitly stated? What are the costs and benefits of the various options? All these aspects of representation emphasize the need to study methods of representation.

The purpose of this work is to develop a computer algorithm which automatically builds models of rigid three-dimensional objects. The algorithm is not intended to help one build a model from the mind's eye. Rather, it is intended that three-dimensional data obtained from a real object be input to the algorithm. The final description is intended for use in display, manufacture, recognition and further analysis by man or machine. This goal is a task, too large for a single dissertation. A number of guidelines help concentrate the effort into a problem.
reasonable size for a single dissertation.

The intended uses of the model are many and varied. All appear to share a common need for geometrical information about the shape and structure of the object. Since the model's intended use is varied, it should preserve as much information as possible and avoid transformations that are not reversible. The proposed model is classified as low level because the details of the object can be constructed from the model. This fact is necessary if the model is to be used for display or manufacture.

In a real environment, complete information is seldom available instantaneously. People compensate for the lack of information by utilizing previously determined models. In this work, no supporting information, such as models of similar objects, is available nor is all the information about the current object available at once to the model builder. The model builder has a finite capacity to retain and actively analyze raw input data, but may make an unrestricted number of requests for information about specific local areas of the object during the analysis. The information provided in response to a request is often partial, much like our human view of things.
Real world objects of interest are three-dimensional: structure. They may have flat or curved surfaces. They may locally convex or concave in shape. They may be classified into simple to complex. The objects used here are motivated by the desire to model man-made objects from the off-line environment. It is desirable to have a modelling system that handles objects in a real environment. However, this is very difficult to achieve and not essential to the presentation problem. In order to simplify the situation, single objects in isolation are considered. This permits a concentration of effort on representation problems. Other problems such as separating several unknown objects are not of interest and are not considered here.

The model building process described uses four primary steps as a general survey of the object. For each view information about the location and orientation of points on the surface is summarized and analyzed. Based on the results of the analysis, groups of data points are formed or representation by a planar or quadric surface. A least-squares fit of the data points in the group determines the coefficients defining the surface. Once a data point is used to determine a surface, it is removed from further consideration. The process of summarizing and grouping is repeated until there are no unused data points remaining. Once all the remaining data points cannot be grouped. Once all the
possible surfaces in a view have been extracted, the intersections can be computed mathematically and the existence of edges verified in the input data. This process of determining edges was not implemented because of the similarity to work done previously by others [LEVIN76]. The formation of a local edge graph completes the analysis of the view. The results of the local analyses are integrated into a global description. This process requires a change from the local viewer-dependent coordinate system to an arbitrary global or world coordinate system. Given the change of coordinates, the integration process must recognize when two surface primitives from different views represent the same underlying surface. If the proposed edge graph were available, this decision could be made based on surface shape, number of edges, and shape of adjacent surfaces. As implemented, the decision is based on surface shape alone.

An object-centered coordinate system is one where the position of the origin and the orientation of the axes are fixed relative to the object. An object-centered coordinate system is important if the description is to be used for recognition from any view. The world coordinate system used during integration is viewer-independent but it is not object-centered. It is proposed that the final coordinate system used to describe the model have its origin at the
center of gravity and its axes aligned with the principal moments of inertia. There are many other possible object-centered coordinate systems, but this one is chosen above the others because it appears feasible to compute its location based on the global object description.

The motivation for studying how to build a description of a three-dimensional rigid object has been presented. In chapter two the major issues of modelling are raised and confronted. Results from other investigations are cited in order to help resolve them. Chapter three describes a proposed surface model without regard to use. It highlights the organization and structure of the model. Chapters four and five share a common structure of topics. Chapter four describes a method of building an instance of the model. Chapter five goes a step further by providing implementation details about the method described in chapter four. Chapter six reports the results of testing some of the key ideas presented. An approximation of a telephone handset is the most challenging object tested. The results for less complex, artificial objects are presented also in order to highlight the strengths and weaknesses of the method. Finally, chapter seven presents conclusions and ideas for future work.
CHAPTER TWO

BACKGROUND

The goal of object description by a machine has been pursued over the years with varying degrees of success. In a classic paper by Roberts [ROBERTS65], many fundamentals required for the analysis of three-dimensional objects are reported. Complex planar objects are presented using a combination of basic three-dimensional volume primitives: cube, wedge and hexagonal prism. Homogeneous coordinates are used to facilitate the expression of projective camera transformations from three-dimensional model data to the two-dimensional image data. The location of object vertices in the image is determined based on extracted edge information. It is assumed that the two-dimensional vertex information of the image is translated from the three-dimensional vertex information of a primitive by a perspective projection. The primitive associated with the transformation having the least error is selected to model that part of the object. Given such a transformation, only a scale factor remains to be determined in order to specify completely the object's position, orientation and size. This final scale factor
This work is determined based on the camera's height. This work of Roberts was the starting point from which the computer vision field developed.

This chapter is divided into two major sections. The first section discusses methods of three-dimensional data acquisition for static scenes. This topic is not of direct concern to the work. However, a survey of methods is presented in order to establish the feasibility of obtaining three-dimensional data. The second section discusses representational schemes. It raises major representational issues and reviews previous research for possible solutions.

1.1 Data Acquisition

Methods of three-dimensional data acquisition may be grouped using various criteria. One criterion is the type of information obtained: spatial information about depth or orientation information about shape. However, this criterion is not useful in all cases because some methods may provide both spatial and orientation information. A better criterion may be the property or feature used in the process of obtaining the results. Three different approaches for obtaining orientation information are examined. One approach depends on the photometric properties of the surfaces present and the lighting source. The second approach depends on the interpretation...
two-dimensional data assuming a three-dimensional source in order to obtain orientation information. A third approach depends on directly "feeling" or sensing the surface using a tactile sensor. Four different approaches for obtaining spatial information are examined. The first approach depends on correlating intensities of pixels in a stereo image pair. A second approach depends on detecting and matching edges of various strengths in a stereo image pair. A third approach depends on identifying artificially created features in an image. Finally, the fourth approach directly measures the spatial information using a tactile sensor.

1.1 Orientation Information -

The term orientation information refers to the information about the local orientation of a surface. The direction of the surface normal at a particular location expresses the local surface orientation in quantitative terms. In contrast, shape information describes the surface's behavior over a larger area. Horn pioneered the development of methods to determine a surface's shape and orientation from observed shading [HORN75]. The initial method utilizes constraints imposed by the reflectivity function of the object's surface and the location of the camera and the lighting source. The reflectivity functions used models lambertian reflectance. Methods that make u
this basic idea are referred to as reflectance

this basic idea are referred to as reflectance ma

the initial method, a more refined and

梢 sophisticated technique has been developed [HORN77].

f addition, techniques that take advantage of specia

cons 
stRAINTS, such as the availability of multiple images,

ave been developed [WOODHAM77] 

HORN/WOODHAM/SILVER78]. Generally, these methods work be

hen the environment is controlled so that the assumption of

ambertian reflectance is true.

A two-dimensional projection of three-dimensiona

formation retains many clues that can be used to

reconstruct or infer the original three-dimensional shape of

an object. The next five works examined utilize various

cles to infer the original orientation information. The
ole that texture and contours play in visual perception

urface shape has been explored [STEVENS79]. The idea that

relationship between a contour generator and the

resulting contour on a surface can be used to reconstruct

ther, knowing the other under certain constraints,

 studied. The use of contour constraints is develop

ther to infer surface shape from image contours

WITKINS80]. The idea that contours are a combination

shape information and projective transformation distortion

both of which are regular in behavior, is advanced.

method for surface reconstruction based on this idea whi
explains the contour best and which produces a smooth surface is used to obtain a surface and its orientation.

The use of texture elements is another way to observe perspective distortion and to estimate surface orientation. Kender showed that the identification of similar texture elements at different orientations is feasible [KENDER77]. Ikeuchi confirmed the method's validity by recovering the shape of a golf ball using the texture of small circles present on its surface [IKEUCHI80]. The use of this method is limited by the need for a consistent texture over the surface of the object.

Kanade identifies geometric assumptions which permit systematic recovery of three-dimensional shape from two-dimensional images [KANADE79]. The idea of "skewed symmetries" is introduced formally as a two-dimensional linear affine transformation of a traditional real symmetry in three-dimensions. The work of Stevens [STEVENS77] presents evidence to support this concept but does not utilize it. A technique to recover surface orientation based on mapping regularities in the image, like parallel lines and "skewed" symmetries, into constraints on shape is demonstrated.
A tactile sensor provides a relatively direct method of obtaining orientation information by observing the pressure differences between various sites on the sensor. These pressure differences can be used to produce an accurate estimate of local surface orientation. However, the size of the sensor is small when compared to the whole area of the object's surface. This fact is a major disadvantage because the sensor needs to be moved physically to many positions in order to obtain a representative sample of orientation information on the object. The development of tactile sensors for use with computers is in progress [WOLFELD81] [HILLIS81].

1.2 Spatial Information

Stereo images can be used to obtain spatial information about the three-dimensional location of surface points. One of the major problems in using stereo is the correspondence problem. The correspondence problem involves identifying the same feature in the two images. Once this problem is solved, photometric techniques can be applied to triangulate the location [WOLF74]. Solutions to the correspondence problem have been demonstrated using pixel information directly [HANNAH74] [GENNERY79]. Such methods depend on correlation techniques to tell when a match has been found. The methods work best when the picture is composed of diverse areas. When areas are similar, these methods
roduce less impressive results. A typical example of situation where these methods may perform poorly is finding corresponding points along an edge formed between the textured leaf surface of a tree and a background of sky. The reason for the difficulty is that there are many local matches that appear equally acceptable along such an edge. Additional global information is necessary to improve the results.

Recently, a theory of human vision was proposed [MARR/POGGIO77]. In this theory, the matching is done on edges instead of directly on intensity. A computer vision system has been developed and implemented to support the feasibility of the theory [GRIMSON80]. The accuracy of the resulting three-dimensional data depends in part on how well the edges can be located. A hierarchy of edges is defined by a measure of edge strength. This hierarchy is used in a sequential process to build incrementally a stereo disparity map. The method works best in scenes containing many edges or texture.

Methods that employ artificial means of creating features have been used to obtain spatial information. An early method that creates features by projecting light patterns has been reported in the literature [WILL/PENNINGTON72]. More recently, a similar method has been reported that projects a grid pattern of light onto
ject to create artificial features [FREEMAN/POTMESIL79]
le projection of the intersection of two lines of the gr:
xms a feature on the surface of the object. Such featu:
e easily found in two images of a stereo image pair ai:
ished. Each feature permits the location of one point t:
le surface to be determined. These surface points then a:
used to generate a surface patch which represents tl
ject. In this case, the grid of light projected need m
known precisely because it is not used directly jasure the geometric properties of the object. Rather, 5 used only to make the solution of the corresponden
problem easier. If a grid and its projection are kno:
recisely, then spatial information can be computed from Lngle image. This computation uses two rays of light, o:
through the camera lens and the other through the lig
source, to triangulate the position of the surface at t
intersection of the two rays. A scanning laser sens:
ystem uses such a computation to determine range da
NITZAN/BRAIN/DUDA77]•

A tactile sensor, in addition to supplying orientati:
information, can be considered to supply spati:
information. A major disadvantage remains the requirem:
to move the sensor physically to many positions. Th:
mechanism used to position the sensor is the real source:
the spatial information. However, in any practical tacti
stem, the sensor and the positioning mechanism are integrated and operate together.

2 Object Representation

From Roberts' early work followed many works which interpreted lines derived from images as edges in a three-dimensional world. Typical of the achievements in this "line" research are Guzman's efforts [GUZMAN68]. Valuable three-dimensional object interpretations are derived from two-dimensional regions present in a single image. To achieve this, Guzman considers evidence suggested by the structure of image regions and their relationships with each other. Recognition of an object is done without references to estimates of three-dimensional measurements of coordinates. Many of Guzmán's techniques are ad hoc, based on observation of what appears to work most consistently. Huffman [HUFFMAN71] [CLOWES71] developed rules to label and describe accurately line drawings. Waltz [WALTZ75] enhanced and refined the performance of this type of analysis by approaching the problem in a systematic manner. Ambiguities used by cracks, obscure edges or shadows are no longer a source for gross mis-interpretation. A catalog of possible line/junction interpretations guides the analysis. The scenarios that Huffman, Clowes and Waltz deal with are limits to what is not representative of the real world. It is assumed
At perfect line drawings of solid planar objects where every corner is formed by exactly three planar surfaces are available. Kanade's origami world expands the domain of planar objects handled by considering constraints imposed by surfaces as well as edges [KANADE78]. It effectively deals with line drawings that are less perfect and more realistic. The extension of line drawing interpretation to include curved surfaces is another important step in understanding scenes. One representative work of this type is reported by Chien and Chang [CHIEN/CHANG74]. As the scenes handled become more realistic, applications to industrial assembly line tasks seem more feasible. However, all the works on line drawings presented here produce qualitative descriptions of shape and use a single view only. They lack the quantitative description present in engineering drawings and are not suitable for CAD/CAM systems. They are just the initial step in understanding three-dimensional shape description. The next step is to investigate representations that are more quantitative in nature.

Object representations based on volumes have been investigated. The use of a generalized cylinder as a volume primitive was suggested [BINFORD71]. A generalized cylinder is characterized roughly as the volume created by sweeping a cross sectional area along an axial curve. The generalized cylinder has been used to model objects like a torus, a co...
More recently, the generalized cylinder has been used to model the three-dimensional structures found in biomedical data [SOROKA79]. The use of spheres to represent three-dimensional objects has been reported also [D'ROURKE/BADLER79]. One great advantage of using the sphere as a primitive is that it is invariant to rotation.

This work is motivated by the medial axis transform [BLUM67]. The transform produces a skeleton-like representation of an object or figure by determining the centers of maximal spheres. The maximal spheres' radii determine the "thickness" of the object.

A major drawback to the medial axis is that singular changes in the boundary produce "spurs" in the resulting skeleton. Attempts to minimize such behavior using smoothing and a relaxed definition of the transform have produced some success [BADLER/DANE79]. An alternative approach to the medial axis transform has been reported [MOHR81]. In this work, non-overlapping spheres are packed into the volume. A skeleton like representation is obtained by connecting the centers of adjacent tangent spheres. The representation can assume a hierarchical structure based on the radius of the spheres. If a coarse model is desired, the skeleton is formed by connecting only tangent spheres with a relatively large radius. As greater detail
required, the radius restriction is relaxed to produce a model of greater detail.

One unanswered question in a representation using primitives is how many primitives are enough? Most of the representations prefer to have too many. Another representation proposed has just three primitives [SHAPIRO/et.al.81]. The three primitives are sticks, plane and blobs. Each instance of a primitive is modified with specific description values. The representation is used for three-dimensional object models. The goal is to study the similarity of objects based on relational distance measures.

The question of object representation is not unique to image analysis. A volume representation was used by MAGI to produce "computer generated perspective views of three-dimensional objects" [GOLDSTEIN/NAGAL71]. There were nine primitive volumes which could be combined in an algebraic manner to form complex objects. An alternative approach for representing an object as a network of surface patches was reported by [BRAID75]. Using this method to represent complex objects has the difficulty of computing and processing the resulting intersecting surfaces. The work has been applied to CAD/CAM [WO077]. The goal of this work is to study the roles of positive and negative solid in creating cavities needed to link the volumetric design approach with existing numerically controlled tools.
Modelling three-dimensional objects using polyhedra is proposed [O'ROURKE81]. An algorithm for determining such a polyhedron given a set of vertices is described and results are presented. However, a method for determining the set of vertices and the sensitivity of the method to different sets of vertices is not addressed.

The use of spatial information in conjunction with reflectance data has been reported [DUDA/NITZAN/BARRETT79]. A scene segmentation procedure for finding planar surface patches is described. It is intended for use in the recognition of objects modeled as polyhedra. Many of the low-level operations are applicable to both spatial and orientational information.

Another method of representation using cubic B-splines and Coons surface patches is reported [ZORK/HANSON/RISEMAN81]. The method is capable of modelling both polyhedral and curved objects. A layered network of entities is used to structure the model. Instances of objects have been designed interactively and a method of matching has been tested. No method of automatically building such a model from a real object is advanced.

The desirability of a viewpoint independent model has been pointed out. Such a model is only one step toward a canonical representation reported [HINTON81]. A method
determining a canonical, object-based reference frame is outlined. The method independently chooses the reference frame and generates a description, making it an ideal candidate for implementation as a parallel computation.

The Gaussian image is a representation used to organize local surface orientation information. It is formed by ignoring spatial information and concentrating the unoriented surface normals at the origin. An extension to this representation called the extended spherical image has been proposed [SMITH79]. In it, a single representative normal whose length is proportional to the surface area at that orientation, replaces coincidental normals of similar direction. This representation is informative because certain classes of surfaces can be identified by the extended spherical images. For example, planar surfaces appear as isolated normals of large length. Cylinders appear as arcs of great circles in the spherical image.

Work describing many three-dimensional objects in a single scene using planar and quadric surfaces has been reported [OSHIMA/SHIRA79]. The three-dimensional coordinates of the surface points in a regular pattern are recovered. Overlapping surface elements formed by fitting planar surfaces to groups of eight by eight surface points are defined. Using a region growing process, adjacent elements are merged into larger elementary regions which are
approximately planar. The resulting regions are classified as planar, curved or undefined based on the variance of the surface element's normals in the region and the size of the region. A second region growing process merges adjacent curved regions into curved global regions if they are connected locally and smoothly. Quadric surfaces are fitted to represent the curved global regions using the original underlying data points. Once the global regions are established, regions' properties and relationships between regions are determined. The method of building a scene description deals with multiple objects in a single scene. It should be noted that the curved surfaces are developed based on statistical parameters that indirectly reflect the surface in a qualitative fashion rather than on quantitative geometric properties like surface curvature. In addition, single views appear to be considered in isolation. An attempt is made to move from a viewer-centered reference frame to an object-centered reference or to combine information from several views. A second paper describing the use of the scene description in recognition has appeared [OSHIMA/SHIRA81].

The use of information from multiple views in the analysis of static scenes has been explored to a very limited extent. The analysis of solid planar convex objects in isolation is reported [UNDERWOOD/COATES75]. The method
squires input in the form of accurate line drawings. The description of each view is two-dimensional in nature and consists of a set of edge segments. A description of faces, edges and junctions is developed. A ratio of line segments' lengths that is invariant under rotational, scaling and translation transformations is computed from two intersecting lines. It is assumed that no significant perspective distortion exists in the views. The two intersecting lines are defined by their reference points determined by junctions of edges. The invariant ratio is used to identify the same surface in different views. No explicit knowledge of slating the different views is needed. However, the restrictions on the sequence of views used in learning the object are imposed. The first restriction is that two or more surfaces in the new view must match a known surface. This restriction eliminates the need to merge two disjoint descriptions of an object by insuring a single connected description. The second restriction is that a new view must contain some new information to be learned. The results of this method are qualitative in nature much like Waltz's results [WALTZ75].
Another method of analysis that uses multiple views for solid bodies bounded by quadric or planar faces has been reported [SHAPIRA/FREEMAN77]. The input for each view is a drawing description also. There are several restrictions, such as corners which are formed by exactly three surfaces and a general camera position, that simplify this complex problem. Unlike the method of Underwood andmates, parameters relating the different views are known and used. This information is needed to identify the same corner or junction in different views. The resulting description is in the form of face groups. Each face group describes the boundary of a single face in three-dimensional terms. The results are more quantitative in nature but do not capture the shape of the surface between the boundaries.

A great variety of representational schemes have been proposed for many purposes. It is only natural to compare schemes, their properties and their uses. A pragmatic comparison of representational schemes based on the operations that can be performed using them and the capability and cost of converting between them is presented [ADLER/BAJCSY78]. The general categories of volume and surface models are used to help structure and clarify the relationships between the various schemes. Many of the issues raised are motivated by problems found in both the computer graphics and computer vision fields.
Marr and Nishihara examine representational constraint imposed by the application and by the computational problem related to processing retinal images [MARR/NISHIHARA77; they identify three criteria as being useful in judging representational scheme. The accessibility criterion is used to judge if a representation can express the required information in a usable form. The scope and uniqueness criterion addresses the issue of the domain of objects that can be represented and the number of possible descriptions for the same object. The stability and sensitivity criterion measures the continuity and resolution of presentation. Aspects of the representation including the coordinate system used to express representation, primitives and their organizational structure are studied. The desirability of an object-centered, modular description utilizing volume primitives is expressed. The basic processes of using such a representational scheme for building models and recognizing objects are presented. The significance of the paper is not in the specific model advanced, but rather it is the identification of representational properties that contribute to finding problem's solution.

Another paper looks at representation from the industrial computer-aided design and manufacturing viewpoint with its need for designing more reliable and versatile...
systems [REQUICHA80]. It provides a summary of important representational issues, compares known schemes of representation and presents a design for a geometric modelling system. In the paper, the study of representation motivated by specific applications is advocated.

3 Knowledge Driven Systems

One of the earliest knowledge driven systems found silhouettes of the human head [KELLY71]. The edges were first found in an image of reduced resolution. This information then was used as a guide in finding the edges in the original image. This work used two ideas which will be seen again: planning and the data pyramid.

More recently, a knowledge driven system using regions was reported [FREUDER76]. In low level vision, the use of absolute threshold values can be disastrous because of the great variation possible in different images. To avoid this problem, Freuder's work used relative thresholds based on the currently known regions.

Sloan created a knowledge driven system to analyze outdoor scenes [SLOAN77]. It used a production system which varied the techniques applied based on the available current knowledge. The behavior of the production system was determined by a set of rules and the current state
knowledge.

Other knowledge driven systems similar to Kelly's system have been developed for use with aerial scenes [ALLARD/BROWN/FELDMAN78] [ROSENTHAL78]. In both cases certain features were being searched for in the images, knowledge about where the features normally appeared was used to limit the search area. For example, if one were looking for a car, then one looked on roads or parking lots, not in an open field. Rosenthal's work used the data pyramid to good advantage to reduce computation also.
A SURFACE MODEL

A detailed description of the static nature of the proposed surface model is presented. The basic element of the model is the surface primitive. A model may use a number of surface primitives to describe the object's surface. These elements are cemented together by an edge graph. The resulting model captures the three-dimensional nature of a real object better than either the surface primitives or the edge graph individually can.

1. Surface Primitives

The surface primitive is a basic element of the model; each primitive is a planar or quadric surface. An instance in the model may use one or more primitives to describe the object's surface. Each primitive represents a finite area of the surface of the object. The area of the primitive could be infinite in theory. In reality, the extent of the primitive is defined by the intersection of the primitive with its neighbors.
The use of planar and quadric surfaces as primitives is motivated by two reasons. Many man-made objects can be modelled accurately using only planar and quadric surfaces. NPT, a language for numerically controlling machine tools, includes planar and quadric surfaces in its surface definitions. Also, the ease of mathematical manipulation when compared to higher order surfaces is a factor in using planar and quadric primitives. The complexity of the surfaces to be fitted affects the process of fitting surfaces to the groups of data points, and this process is an integral part of the model builder. A least squares method of fit is used to determine the coefficients from the raw data. As surface primitive complexity increases from planar through quadric toward higher order surfaces, the number of coefficients required to define the primitive increases and so does the size of the least squares problem. In addition, a model composed of quadric surface primitives offers an advantage in determining object symmetries. A quadric surfaces have at least one plane of symmetry. Some quadric surfaces, like ellipsoids, have three planes of symmetry.

A surface primitive can be expressed as an implicit equation of the form

\[ f(x,y,z) = 0 \]

where \( f \) is a scalar function of order two or less in the
variables x, y, and z. The location of a point in \( \mathbb{R}^3 \) Cartesian space is represented by the three-tuple \((x, y, z)\). The point lies on the surface if the equation \( f(x, y, z) = 0 \) is satisfied. The surface divides the space into two half spaces. In one half space the function \( f \) is always positive, and in the other half space the function \( f \) is always negative. A surface defined by an implicit form \( f(x, y, z) \) is unique because if \( f(x, y, z) \cdot 0 \) then \( c \cdot f(x, y, z) \cdot 0 \) where \( c \) is any real constant. It is necessary to add a constraint to insure that a surface has only one form.

Efficiently testing whether a point is on a surface and uniquely defining surfaces are two advantages in using parametric surfaces. Consider a surface defined by a parametric form such as

\[
F(u, v) = P
\]

where \( F \) is a vector function of rank three and order two less in the parametric variables \( u \) and \( v \). \( P \) is a point on the surface determined by the values of \( u \) and \( v \) in the restricted domain. One can express the same surface in many different ways using a parametric form. In general, there is no systematic method for determining the equivalence of two equations. In addition, there is no simple method to test whether a given point lies on a parametric surface, except for the fact that these facts, the implicit form of the equation can be used to represent the surface primitive.
The edge graph is essential for the accurate representation and reconstruction of the object by the derived model. A surface primitive expresses the basic shape of part of the object. However, a primitive may specify implicitly a surface that is larger than intended. For example, four coefficients define a plane of infinite area. Only a small portion of that plane is a valid representation for a planar face of a finite object. The edge graph contains explicit information about boundary curves which define the valid extent of each surface primitive.

The information contained in the edge graph is defined by the intersections of adjacent surface primitives. Computation of the intersection of two or three quadric surfaces has been investigated [LEVIN79]. The curve resulting from the intersection of two quadric surfaces lies on the surface of a ruled quadric and can be expressed in canonical parametric equation. An edge is represented by the equation that describes the X, Y, and Z coordinates as a parameter is varied over a range of values. An endpoint or corner of an edge is determined by the intersection of the surfaces that meet there. In order to compute corner locations, a trace of the sequence of neighboring surface primitives encountered along the
boundary of the primitive is required. The intersection of two sequential neighboring surface primitives from the trace and the original surface primitive determine a corner. Once the corners are found, the model need record only the coefficients of the edge equation and the extreme points of the range in order to reproduce the boundary of the surface primitive.

3 The Model Structure

There are four units or records of information that are combined to form an instance of the model.

Each object model has one object record. It contains global information about the object such as the number of surface primitives, the number of edges, and the number of corners. In addition, it contains a list of pointers to the surface primitive records.

Each surface primitive is represented by a separate record. The surface primitive record contains a set of coefficients which define the surface. The coefficients of the surface equation are defined such that an outward pointing normal is obtained by partial differentiation. In addition, there are three sequences of pointers: one for neighboring surface primitives, another for edges and the last for corners. A sequence differs from a list in that
Sequence implies a specific order. The reason for requiring an ordering is that there is a correspondence among neighbors, edges and corners. For example, the intersection of the i-th neighboring primitive and the current primitive forms the i-th edge. Also, the i-th edge begins at the i-th corner and ends at the i+1-th corner.

An edge record describes an edge between two surface primitives. It contains edge coefficients, parametric limit values and pointers to the associated corner records. The coefficients define a parametric vector equation. This equation defines the edge in spatial coordinates. The parametric limit values represent the extreme range of parametric values for the edge equation. In addition, each edge record contains pointers to the associated corner records.

A corner record expresses explicitly the spatial location of the intersection of three or more surface primitives. This information is implicitly available via the parameter limit values and edge equation of the edge record.
4 An Example

Figure 3-1 shows an object that may be described as a sphere with two flattened planes. Figure 3-2 shows the record structure of the proposed model in this instance.

A View of an Object

Figure 3-1
Model Structure

Figure 3-2
A method for building an instance of the surface model is described. An outline of the method is presented in a set of stylized procedures. Words in procedure titles which are prefixed with a denote parameters specified by the use of the procedure. Discussion of the method includes a description of the desired and its source, the local analysis of data from a single view, the integration of results from analyses into a global description, and the transformation to an object-centered reference frame. Chapter seven provides in corresponding sections additional detail in implementation.

The procedure "build surface model" describes the low level processing of the method.
PROCEDURE: BUILD SURFACE MODEL

FOR each primary view

- Input data for the current view
- Analyze the current view

IF a partial global description exists

THEN

- Integrate the current results into the global description

ELSE

- Make the current results into the global description

END-IF

END-FOR

END-PROCEDURE

1. Input Data

The input consists of groups of data points. Each such group, like a photograph, contains only partial information from a specific point of view. The information in each group is expressed in its own local coordinate system. There are many possible groups corresponding to different views. A group is specified by "viewing" parameters in arbitrary but fixed global reference frame. This global reference frame is the bridge that links together the local coordinate systems of different views. These input
assumptions are consistent with methods of data acquisition of three-dimensional information.

1.1 Sources Of Data -

There are many methods of obtaining three-dimensional data about objects. For developmental purposes, it is desirable to use a method that produces "clean" data with little or no error. Also, the ability to generate repeatedly the exact same data is useful for debugging programs. An artificial data generation program is needed to produce both spatial and orientation information about points on the surface of an object. In the field of computer graphics, programs that generate shaded images of modelled objects are required to generate similar information [GOLDSTEIN/NAGAL71]. As an expedient solution to constructing an artificial data generator, such a shading image algorithm is used as the basis for the computation of the input data.

The topic of three-dimensional data acquisition has been discussed generally. There appear to be two practical methods of obtaining the desired input data from real world situations. The first method obtains data from pairs of stereo images. A second method involves the use of a tactile sensor under computer control.
1.2 A View -

The basic unit of input is the point. Each point in a group is assumed to lie on the surface of the object. In addition, it is assumed that there are no intervening interfaces present between the surface point and some fixed point in space. This assumption permits one to think of the group of data points as appearing in a single photograph taken by a camera located at the fixed point in space. Hence, the use of the word "view" to describe a group of data points. Also, like a camera, the resolution of the data is a function of the distance between the fixed point and the center of the object.

Each group or view expresses its information in its own local coordinate system. The origin of this system is located at the camera position. The orientation of the system is such that the object is located along the negative z-axis (see Figure 4-1). For each point, the X, Y, and Z coordinates and the local surface orientation at the point are known. The orientation of the local coordinate system with respect to the camera position and the object guarantees that the Z components of all the surface normals are always positive. So, it is possible to express the surface orientation in terms of a unit surface normal with only two numbers. The two numbers are interpreted as the X and Y components of the unit surface normal.
The data points in each group are assumed to be uniformly scattered relative to the local X-Y plane. Intuitively, this assumption minimizes local blind spots due to sampling. However, it does not guarantee their total absence. No systematic spatial inter-point relations among data points, such as a regular grid pattern of points, is assumed. This fact permits a greater variety of input resources to be used. Without this systematic relationship, there is no easy way to determine a given point's neighbors.

The analysis has available, for use at any one time, a group of data points corresponding to a single view. However, there is no restriction placed on the number of views or the points of view used. Initially, data from four primary views are investigated. Currently, the model builder processes views sequentially. Since the primary views are intended as a general survey of the object, the local analysis of the primary views could proceed in parallel. It is expected that features may be discovered that require supplementary exploration.

Each primary view corresponds to an image seen by a camera at a vertex of a tetrahedron, with the earner pointing toward the center of the tetrahedron. It is intended that the views overlap a small amount in order to guarantee the model builder sees complete information.
eventually. Complete coverage of the surface is necessary in order to obtain closure of the object description and prevent "loose ends". An arbitrarily oriented global reference frame whose origin is near the object is used to specify the viewing parameters. These parameters may be interpreted as a camera position and a camera orientation in space. If the information from a primary view is used to generate a shaded image, the object would appear to fill roughly ninety percent of the image, and it would be entered in the field of view.

In a primary view, the object is framed against the background. In a supplemental view, there is no requirement that the field of view include the whole object. Data from supplemental views may be requested dynamically as the analysis proceeds. The need for supplemental views arises when a primary view contains ambiguous or insufficient information about a local area. Therefore, it is expected that a supplemental view contains information about a limited part of the object. The analysis of supplemental views occurs after the primary view's analysis is complete. No evidence exists as to whether it is better to merge the supplemental results into the primary results before integration into the global description or to integrate the supplemental results directly.
4.2 Analysis Of A Local View

The local analysis is described by the process of "analyze local view". There are two major tasks performed in the analysis of a single view. First, the analysis forms subgroups of data points and determines surface primitives which adequately represent them, and the second task is to compute the location of edges indicated from the intersections of surface primitives and their existence in the data.
PROCEDURE: ANALYZE THE *local VIEW

- Obtain data about the *local view
- Determine the *local surface primitives
- Determine the *local edge graph

IF the *local description is not complete internally
THEN

REPEAT

- Determine a supplemental view
- Analyze the supplemental view
- Integrate the supplemental results into
  the *local description

UNTIL the *local description is complete internally
OR maximum resolution is obtained

END-IF

END-PROCEDURE

4.2.1 Determining Surface Primitives -

The procedure "determine the local surface primitives" describes the process of finding surface primitives that represent groups of data points.
PROCEDURE: DETERMINE THE LOCAL SURFACE PRIMITIVES

REPEAT

- Summarize the unused data points
- Group the data points based on the curvature and depth continuity reported in the data summary

FOR each group found

- Determine a surface primitive by a least squares fit of the original data points
- Remove the used data points from further consideration

END-FOR

UNTIL there are no unused data points remaining OR no new primitives are found

END-PROCEDURE

4.2.1.1 Structuring Input Data -

In order to organize the input data, an array structure is utilized. Each property of interest is represented in a separate, two-dimensional array. Corresponding elements in a set of registered arrays contain different information about the same volume in the domain represented. Use of a stack of registered arrays has been proposed as a parallel computational model [BARROW/TENENBAUM79B]. Its use here for sequential processing, and there is no immediate modification of existing values in the various arrays.
The input data provided has no systematic structure; each permits it to be expressed using the array structure directly. To facilitate the use of an array structure, the number of regularly spaced rectangular volumes is defined, which side of these volumes is orthogonal to one of the three coordinate axes. The front of the volume is the plane \( Z = +00 \) and the back of the volume is the plane \( Z = -00 \). The projection of the volumes onto a local X-Y plane produces a 2-dimensional grid which serves as a basis for mapping the volumes into an array structure. Each array element presents all the data points within the corresponding cellular volume.

The use of an array to summarize local property determined by several data points is an interesting use of hierarchical data structure. The use of such a data structure is not new to vision systems. See [ROSENTHAL7] for additional details. There are three primary purposes for using the array structure as the second level of the hierarchy. First, it allows for a systematic way to determine a cell's closest neighbor. The implicit knowledge of the array's structure makes this operation possible. Second, a cell value represents information about several data points. This reduction of data saves memory space and...
chosen carefully, the reliability of the data may be increased. It is desirable to utilize properties that depend on all the values, not just one. Two examples of statistical properties that depend on all the data points are the average value and standard deviation. In contrast, the statistical properties of minimum and maximum value may be affected adversely by a single bad data point. During the local analysis, decisions based on single points are avoided.

2.1.2 Types Of Properties –

The properties represented by arrays are divided into two groups: observed properties and derived properties. Table 4-1 describes the observed properties and Table 4 describes the derived properties. The observed properties are statistics computed directly from the original input data. The derived properties require the use of some special knowledge about geometry in order to compute their value.

<table>
<thead>
<tr>
<th>Observed Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of data points in local area</td>
</tr>
<tr>
<td>Average and standard deviation of local Z values</td>
</tr>
<tr>
<td>Average and standard deviation of X component of local surface normals</td>
</tr>
<tr>
<td>Average and standard deviation of Y component of local surface normals</td>
</tr>
</tbody>
</table>

Table 4-1
Derived Properties

- Local curvature in an X-Z plane
- Local curvature in a Y-Z plane
- Surface orientation continuity
- Surface depth continuity

Table 4-2

2.1.3 Growing Groups Of Data Points -

One of the parts of the local analysis groups data points for representation by surface primitives. There are two methods for doing this task. A general purpose method is described first. It works for both planar and quadric surfaces. A specialized method for only planar surfaces is described after the general, but computationally more expensive, method. In both methods, evidence that indicates the presence of the same type of surface in local adjacent cells is sought in order to grow groups of data points.

2.1.3.1 Quadric Surfaces -

The general method is based on two assumptions. The first assumption is that as a single surface is traversed in any direction, the sequence of surface normals should change smoothly. For surfaces of uniform curvature, the components of the normal vary linearly. This fact has been observed before and used to reconstruct spherical or cylindrical surfaces [BARROW/TENENBAUM79A] Also, the change in the
Surface normals should be consistent. For example, consider traversing the curve formed by the intersection of a sphere and an X-Z plane in the positive X direction defined by the local coordinate system. The surface is not a full sphere but rather a hemisphere because of the partial data available to a viewer. The planar curve of intersection is part of a circle and has two endpoints if the degenerate case of a plane tangent to a sphere is not allowed. Starting at the negative X end of the curve, the X component of the surface normal is largely negative. As the X value increases as the curve is traversed, the X component of the normal increases in value toward the positive. The final normal has the most positive value of the X component for all the surface normals on the curve. The value of the component of the surface normal changes smoothly as the curve is traversed, and the change is consistently in the same direction. It should be noted that the plane intersection used to determine the curve examined is viewer-centered rather than object-centered. Therefore, the curve is not one of intrinsic importance to describing the shape of the object. This expedient approach is taken however, because it is assumed that nothing is known of the object’s shape. It works because the goal to identify "sameness" is modest. If a more ambitious goal identification of surface type is selected, this simple approach would not be sufficient.
The second assumption is that a single surface should have a smooth surface as reflected in the depth or Z values of the data points*. It is assumed that the underlying surface is not changing greatly in a small local area allowing the range of surface Z values in such an area can be estimated successfully knowing the average surface orientation. Evidence of the presence of more than one surface is indicated when the observed range of Z values significantly exceeds the estimated range. Again, the evidence collected seeks to identify "sameness".

These two assumptions are complementary in nature. The first deals with orientation information, and the second deals with spatial information. Either, by itself, may fail to detect the presence of two surfaces. Figure 4-2 shows an image with two surfaces in different spatial locations where the first assumption fails because the surfaces have similar orientation. Figure 4-3 shows an image with two different surfaces where the second assumption fails because the surfaces are located close together.
Similar Surface Orientation

Figure 4-2

Similar Spatial Location

Figure 4-3
2.1.3.1.1 Orientation Continuity -

The first assumption is implemented as a local shape labelling process. The goal is to label each cell in the array structure with a name that is characteristic of the shape of the surface within the local area and to identify larger areas of local shape continuity. Labels like convex, concave, flat or unknown are not adequate, and a richer set of labels is needed. As an illustration of this need, consider the difference in shape between the sphere and the cylinder of Figure 4-4. Both objects may be described by the label convex, yet there is a significant difference in shape. By examining the curvature of two curves determined by the perpendicular planes X=0 and Y=0, the difference is shown. The careful selection of the planes contributes significantly to the example's clarity. If the planes X+Y=0 and X-Y=0 had been selected, a single pair of planes would not be sufficient to show clearly the difference in shape. Fortunately, the local analysis has information about many planes parallel to the two selected on which to base inferences.
1.3.1.2 Depth Continuity -

The second assumption is implemented in two steps. The first step checks for surface continuity within a local 1. An estimate of the range of Z values under the assumption that only a single surface is present is computed. When the observed range of Z values significantly exceeds the estimated range, the presence of more than one surface is indicated. The second step checks for continuity between adjacent cells. When the Z values in the two adjacent cells differ significantly, the presence of more than one surface is indicated.
2.1.3.2 Planar Surfaces -

A simpler method for planar surfaces has been implemented based on the fact that planar surfaces are flat. This flatness property is reflected directly in the range of values the components of the surface normals assume. All the normals from a single planar surface point in a single direction. Ideally, any cell of the array containing data points from a single planar surface has no variation, and the range of values for the components of the surface normals is zero.

Adjacent cells with similar average values for the components of the surface normals and with zero ranges are considered for representation by a single planar surface. However, the evidence reflects only orientation information, it is necessary to check that two surfaces of similar orientation are not present. This situation is checked transforming the data points belonging to the region of interest into a new coordinate system where the surface normals are aligned with the new Z axis. If only one planar surface is present, then there should be only one common value for the transformed Z coordinate of the data points.
2.1.4 Surface Fitting -

After a group of data points has been selected, the surface is fitted. The growing process provides the initial group of data points to be represented and the general type of primitive required: planar or quadric. The data points are drawn from a limited area of the total surface area because of the conservative nature of the growing process. This fact makes the accurate estimation of the surface parameters more difficult. Errors in the data points further complicate the problem. Two criteria, spatial and orientation, are combined to determine the value of the surface parameters.

2.1.4.1 Fitting Criteria -

A least squares fit of the observed data points is used to estimate the underlying mathematical surface. Two criteria are used: spatial and orientation. Consider each type separately and independently. Ideally, for spatial information, the intuitive geometric idea of minimizing the sum of the square of the distance between the data point and the surface is desirable. For orientation information, the angular difference between the observed and computed surface normals should be minimized. Nothing is known about the surface, so it is difficult to implement these criteria directly.
Instead, less intuitive criteria are used. In the limiting case where there is no error, both sets of criteria lead to the same estimation of surface parameters. However, with imperfect data, no such claim can be made.

2.1.4.1.1 Spatial Criterion -

Let a quadric surface be expressed implicitly as

\[ ax^2 + by^2 + cz^2 + dxy + eyz + fzx + gx + hy + jz + k = 0 \]

for \( Q(X,Y,Z) = 0 \) for short. For an individual point, the spatial criterion chosen to be minimized can be expressed as

\[ Q(X,Y,Z)^2 \]

If there is no error in the data and its source is a quadric surface, then the minimum value of Expression 4-1 is zero; however, there is no direct intuitive geometric interpretation of the error in cases where the minimum value is greater than zero.

The spatial criteria used for fitting surfaces selected to simplify the mathematics involved with the least squares fit. However, the errors associated with the fit cannot be used directly to determine the goodness of fit because the error measure is affected by the value of the surface coefficients.
absolute terms, such as Euclidean distance, is important. A bound on the Euclidean distance error measure using an error criterion of Expression 4-1 is developed.

Let \( P_0 \) be the location of a point on the surface \( Q \) that \( Q(P_0) = 0 \). Assume that an error in position, \( \Delta \), is introduced during the observation process. Let \( P_1 \) be the observed location corresponding to \( P_0 \). Assume that \( P_1 \) is not on the surface, so that \( Q(P_1) \neq 0 \). Now, consider a first order approximation of \( Q(P_0) \) obtained by a Taylor’s series expansion of the function \( Q \) about the point \( P_1 \),

\[
Q(P_0) \approx Q(P_1) + \Delta P \cdot \nabla Q(P_1).
\]

Since \( Q(P_0) = 0 \),

\[
Q(P_1) \approx -\Delta P \cdot \nabla Q(P_1).
\]

Substituting the definition of the dot product, the expression is

\[
Q(P_1) = -\|\Delta P\| \|\nabla Q(P_1)\| \cos \Theta,
\]

and squaring both sides yields

\[
Q(P_1)^2 = \|\Delta P\|^2 \|\nabla Q(P_1)\|^2 \cos^2 \Theta.
\]

Since

\[
0 \leq \cos \Theta \leq 1,
\]

\[
\|\Delta P\|^2 \|\nabla Q(P_1)\|^2 \geq Q(P_1)^2
\]
This last inequality shows that the spatial error criterion of Expression 4-1 can be expected to be a consistent estimate of the error as long as the value of $\nabla Q(P_1)$ and the angle $\theta$ are relatively constant over the range of data points. In addition, the estimate of the Euclidean error distance can be used to place the residual errors of different primitives on a common scale for comparison of the accuracy of the underlying data points.

2.1.4.1.2 Orientation Criterion -

Let $A$ be a vector representing the actual surface normal and let $B$ be a vector representing the observed surface normal. The observed surface normal is a unit vector, so $||B|| = 1$. The normal of the fitted surface, which is derived from $Q$ by differentiation, and its length is necessarily one. Consider the expression

$$N(X,Y,Z) = ||A||^2 - (A \cdot B)^2$$

(Expression 4-2)

Expanding the expression using the definition of dot product...
Lelds

\[ N(X,Y,Z) = \|A\|^2 \cdot (\|A\|^2 \|B\|^2 \cos^2 \theta) \]

where \( \theta \) is the angle between the observed and actual normals. Further simplification leads to

\[ N(X,Y,Z) = \|A\|^2 \|B\|^2 (1 - \cos^2 \theta) \]

or

\[ N(X,Y,Z) = \|A\|^2 \sin^2 \theta . \]

This expression obtains a minimum value of zero when the two normals are parallel and is positive otherwise. Also, note that the magnitude of \( A \), \( \|A\| \), can be written as \( \|\nabla Q(P)\| \) at any point \( P \).

2.1.4.2 Minimization -

The error criterion used is of the form

\[ \sum_{i=1}^{n} Q(X,Y,Z) + N(X,Y,Z) \]

values for the coefficients \( a, b, c, d, e, f, g, h, j \), and \( k \) are desired which minimize this expression. There are many solutions to this problem since any surface defined \( (X,Y,Z) = 0 \) is defined equally well by \( 1 \cdot Q(X,Y,Z) = 0 \), where
\[a + b + c + d + e + f + g + h + j + k = 1\]

is introduced in order that a unique solution may be determined.

The problem is now a constrained minimization which may be solved using the theory of Lagrange multipliers [FULK69]. The theory guarantees that a function \(F\) takes on a local extreme value subject to a constraint function \(G=0\) when

\[\nabla F - \lambda \nabla G = 0 \tag{Equation 4-3}\]

Setting

\[F = \sum_{i=1}^{n} Q(X, Y, Z)^2 + \sum_{i=1}^{n} N(X, Y, Z)^2\]

and

\[G = a + b + c + d + e + f + g + h + j + k - 1\]

permits the theory to be applied to this problem. The solution associated with the minimum lambda is the desired one. The vector equation of 4-3 can be expressed as ten scalar equations. Each one of the scalar equations contains partial derivatives with respect to one of the ten unknown coefficients. Implementation of a method of solution facilitated by observing that finding a solution to Equation 4-3 is an eigenvalue problem. Methods for determining eigenvalues are well known and standard software packages...
exist which produce acceptable solutions. If a different constraint equation is chosen, this fact may not be true.

2.2 The Local Edge Graph -

After all the primitives have been determined, the second task of the local analysis is to determine the edge and corners. The procedure "determine local edge graph" describes briefly this process. Ideally, the intersection of adjacent surface primitives defines the boundary extent of the primitives. This determination is more accurate than a direct estimation from the original sampling of the three-dimensional data. Once an edge's location is computed, it can be verified in the input data and can be recorded explicitly in the object description.

In a local view, the appearance of adjacency based on the X and Y location does not insure that two surface primitives form a real edge. The depth continuity property array discussed previously can provide evidence to refute the existence of a common boundary. When two primitives form an edge, it is desirable to express their common boundary in terms of a parametric equation and a pair of parameter values denoting the range of the boundary. This information is determined for each pair of surface primitives, a graph of edges and corners can be built.

Also, knowing the exact extent of the surface primitives,
trmits the reconstruction of an accurate shaded image <
le object from the description.

Surface primitive boundary closure is important. Oi
ght assume that after having found all the edges that ti
suiting boundaries of the surface primitives in the loc
would be closed. In general, this is not tru<
xrface primitives adjacent to depth discontinuities
cking an adjacent neighbor along part of their bounda
cause of local perspective do not have closed boundarie
le missing part of the boundary can be filled in
formation from another view. Also, the corners associat
ith edges adjoining a missing boundary may not be tr
orners, but virtual corners resulting from the loc
erspective. It is important to include some knowled
bout these virtual edges and corners in the local analys
order to avoid mistaking them for the real thing. Th
art of the edge graph where a virtual edge or corner see
o "appear" should be marked as incomplete.
PROCEDURE: DETERMINE THE LOCAL EDGE GRAPH

FOR each primitive

WHILE "walking" around the boundary of the primitive in the x-y space
- Compute the intersection of the primitive with its neighbor
- Verify the edge exists in the original data
- Update the edge graph with information about the "real" edge just found

END-WHILE

END-FOR

END-PROCEDURE

4.3 View Integration

After the local analysis of a view is complete, results are integrated into a global description of the object. This integration is characterized by several distinct steps: transformation of information from a coordinate system to a global coordinate system, computation of a measure of similarity between two surfaces, identification of identical surfaces, and the modification of surface parameters based on new information. The procedure "integrate the local results into the next-description" describes briefly this process.
PROCEDURE: INTEGRATE THE *local RESULTS INTO THE *next-level DESCRIPTION

- Transform the *local results into the *next-level's reference frame
- Identify the *local primitives in the *next-level's results based on primitive similarity and edge information
- Update the *next-level's description to reflect the matched information and append new information

ND-PROCEDURE

3.1 Transformation To A Common Coordinate System -

The original data and information derived from the local analysis is expressed in the local coordinate system. There is some global reference frame or coordinate system used to specify the different camera positions. The actual global coordinate system used is less important than the relationships between the various views. Knowing these relationships, transformations that map information from each of the local coordinate systems into a common global coordinate system may be computed. The change of reference frame from the local, viewer-centered one to an arbitrary global one is the first critical step toward achieving a description that is viewer independent.
3.2 Surface Identification -

A major issue in view integration is the determination of whether or not a surface primitive from the local view has been seen before. Intuitively, the location, orientation and basic shape of the surface are factors to consider. In addition, information provided by the edge graph is of value.

3.2.1 The Types Of Surface -

Each surface primitive is categorized by the local analysis as planar or quadric. In the global description, the category of quadric is refined into ellipsoid, hyperboloid of one sheet, hyperboloid of two sheets, elliptic paraboloid, hyperbolic paraboloid, cone, and cylinder. Membership in a refined category is based on numerical properties of the surface. The coefficients of the surface are mapped into a continuous decision space. The set of hypersurfaces divides the space into regions defining the categories. Each surface receives the label of the region into which it maps. For additional details about determining a surface's type see [LEVIN76].

The refinement process for surface types transforms the description by abstraction, and it can be misleading to consider the three surfaces depicted in Figure 4-
surfaces \( s_i \) and \( s_2 \) lie in the same region and are separated by a relatively large distance. In contrast, surfaces \( s_2 \) and \( s_3 \) lie in different regions but are relatively close; while \( s_i \) and \( s_2 \) share a common label, \( s_2 \) and \( s_3 \) are more similar geometrically. For this reason, generic type not used as a measure of two surfaces' similarity.

\[ \begin{array}{c}
s_i \\
\hline
s_2 \quad \text{Decision Space} \\
\hline
s_3
\end{array} \]

\[ \text{Figure 4-5} \]

3.2.2 Measure Of Surfaces' Similarity -

In order to decide if a new surface matches an existing surface in the description, a measure of similarity is computed. Such a measure is computed between the new surface and each existing surface in the object description. If no measure falls below a predefined minimum threshold value, then the new surface is assumed to be unique and added to the object description. Otherwise, the old surface with the lowest similarity value is assumed to match the new
surface. The new information can be used to modify the existing surface description. The measure of similarity is computed as the weighted sum of the square of the differences of the corresponding surface parameters.

The measure of similarity used is an unsophisticated first attempt which lacks an intuitive, geometric interpretation. A more sophisticated measure which explicitly considers the shape of the surface and its location and orientation in space is seen as the next step. For quadric surfaces, there are methods of extracting and separating these pieces of information from the surface coefficients [LEVIN76]. Given this new measure, it should be possible to predict the sensitivity and robustness of the measure in geometric terms.

3.2.3 The Role Of The Edge Graph -

The identification of the same surface primitive from different points of view does not depend on the measure of surface similarity alone. During each local analysis, an edge graph can be developed which contains information relating adjacent surface primitives. Once tentative identification is made based on similarity, the adjacent information is checked for consistency. The ability to merge the local and global edge graphs in the absence of conflict provides additional support for the current
3.3 Description Updating -

Integration of a new local analysis result should improve the global description. If a local primitive has not been seen before, the global description is expanded to include it. If a local primitive is judged to exist in the global description already, it still may be necessary to modify the global description. The shape of the surface primitive may differ between the local result and the original global description. The question of how best to combine these two pieces of information in order to create a more accurate shape description is not addressed in this work.

The information contained in a single view is often partial. In general, several views are required in order to obtain a complete description. The global description is built incrementally. After a local analysis is completed, the derived surface primitives and the edge graph are integrated into the global description. While it is not required that the global and local descriptions being integrated share some common features, it is desirable. If they do, conflicts in descriptions can be detected and resolved immediately. The new global description is then one of a single, connected surface area rather than the disjoint areas. Having disjoint surface areas within
bal description is not fatal because, at sometime, a nev:
al view will provide information linking the two areas.
ever, an integration of a view that joins two disjoint
as is more complex and difficult than an integration that
Is a view to a single area description.

i The Final Reference Frame

The final form of the object description uses --<
ference frame whose origin lies at the center of gravity
the object and whose axes are aligned with the principal
ents of inertia. The center of gravity and moments of
artia can be obtained by several methods. One method uses
a surface primitives as the basic unit of mass t<
proximate these values. Another method requires th<
aversion of the surface model to a voxel representation
uses the voxel as the basic unit of mass to approximati
the center of gravity and moments of inertia. Once th<
liter of gravity and the moments of inertia are known, th<
final transformation from the arbitrary global referenc
frame to the object-centered reference frame can be compute"d
plied to the model. The last transformation result
the final reference frame of the description being tie
the structure of the object rather than to the local
pective of the viewer or some arbitrary reference frame
CHAPTER FIVE

IMPLEMENTATION

This chapter provides additional information which is required in a practical implementation but is not relevant directly to the understanding of the basic method of the model builder. Many of the comments presented here are the direct result of experience gained in implementing or using computer programs to test the ideas presented. However, some comments are based indirectly on results and may be speculative in nature.

1.1 Input Data

Implemented computer programs use input data that is processed by groups or views. Each group is limited to 6-bit words of memory. A data point consists of five real numbers: three numbers expressing spatial information and two numbers expressing orientation information. Therefore, each data point requires ten words of memory, and a group is limited to a maximum of 409 data points.
1.1 Sources Of Data -

Artificially generated data is obtained from a data generation program based on a graphics system called JADRICS. Real input data can be obtained from the analysis of pairs of stereo images or a tactile sensor. Currently, there are no sources providing information about real objects available locally for use with this work, but they are reported in the literature.

• 1.1.1 Artificial Data -

The QUADRICS system is a constructive geometric modeling system that permits the production of shaded images. In the course of constructing the images, the type of three-dimensional information desired here as input is aerated. The system models objects using volume primitives whose surfaces are quadric or planar. The surfaces are represented in the program in an implicit conjunctional form. The volume primitives are defined by the intersection of half spaces associated with these surfaces! The user is not concerned with surfaces but rather with primitive volumes. The volumes may be combined using boolean-like operators of NOT and OR to produce convex and concave objects. However, the valid grammar for combining volumes is restricted because primitive volumes may not
The original QUADRICS system may be found in [STRAUSS80]. Additional information about similar modeling systems is available in [GOLDSTEIN/NAGEL71].

A data generation program is needed to produce both spatial and orientation information about points on the surface of an object. While QUADRICS and the data generation program needed here have different goals, they share many similar requirements. For example, both need to compute surface normal information for a given point on the surface. The QUADRICS program generates such information by computing the partial derivatives of the surface from an implicit second order equation. As an expedient solution in constructing a data generation program, QUADRICS was borrowed and modified. Both programs share a common external form of model representation. The shaded image generation algorithm is the basis for the computation of the spatial and orientation information. In the case of the data generation program, this computation runs under program control rather than human direction, and the results are numeric rather than graphic. The effect of different views is obtained by transforming groups of primitives. The original QUADRICS system is used to generate a model of the test object under human direction. The data generation program reads a file created by QUADRICS and generates a group of three-dimensional data points from a view specification.
r the analysis program.

1.1.2 Real Data -

Methods of obtaining three-dimensional data have been viewed in chapter two. The purpose of this section is to estimate the quality of the data obtained from these methods. In the case of stereo, it is assumed that the coordinate system expressing the data is oriented such that the Z axis is parallel to the average of the two optics associated with the stereo pair of images. Each three-dimensional data point is determined by the intersection of two lines of sight, one from each earner; if these lines are close to being parallel, then the range of Z value is expected to contain the major portion of the error. The physical layout of the data acquisition system will determine the allowable camera positions; hence, will affect the accuracy of the data. Section 10.6 [DUDA/HART73] presents an error analysis for stereoscopic perception. See [DERISI81] for additional details related to the implementation of a stereo algorithm.

The tactile method is capable of producing data of greater accuracy. Assuming the tactile sensor is not constrained in its orientation for a given position and that it is free to make the best use of its abilities, the error
values. This method appears to be able to produce the type and quantity of information required more easily than the stereo method. The tactile sensor offers a unique opportunity for the interleaving of data acquisition and analysis because the rate of acquisition is limited by the ability to move the sensor quickly. The sequential analysis of data as it becomes available and the ability to change the acquisition strategy in progress to take advantage of the new information remain large unresolved problems. These are not considered here because of their size and complexity. See [WOLFELD81] for additional details on the tactile sensor.

Obtaining orientation information directly is difficult. Orientation or shape has been successfully recovered from intensity data in a controlled environment [HORN75]. In an uncontrolled environment, other methods must be used. An alternative approach for obtaining orientation information estimates the surface normals mathematically from depth information. This procedure involves fitting surface patches to local areas and computing partial derivatives from the patches to estimate the normals. A patch may be planar, quadric, bi-cubic, or any other one that is computed easily. The planar patch is favored since minimal effort is desirable. The accuracy of the normal estimate depends on the location and error of the
points used in fitting the patch. In order to insure the independence of the spatial and orientation errors, point used to determine the normal estimate should not be used also as other actual data points in the group. This fact makes it necessary to over sample the surface in order to compensate for the lack of orientation information. The rate of sampling depends on the accuracy of the orientation information desired.

1.2 A View -

Artificial data is obtained from the data generation program. The underlying process involves the computation of spatial and orientation information used to form a 64 by 64 pixel shaded image. The data points generated are many in number and regularly spaced in a grid. In fact, the 40 potential available data points are more than the memory of the PDP11/60 can hold practically at one time. Some potential data points are not realized because the location corresponds to the background in the image. Order to conform to the input requirements mentioned above, approximately 400 data points are selected at random from the ones available. The system-supplied random number generator, RANDU, is used as the basis of the selection process. The same initial seeds are used on every view RANDU to insure reproducible input data for debugging.
The ability to examine a part of the whole is an invaluable tool in the analysis of objects. Previously, in the description of the model builder, the use and purpose of primary and supplemental views was discussed. The implemented computer programs consider only primary views; the local analysis generates requests for supplemental views, but they are not honored. The incomplete results of the local analysis are integrated into the global description. In many cases, redundant information from other local analyses fills in the gap.

2.2 Analysis Of A Local View

The analysis of a single view has been implemented partially. The second task of determining the edges and intersections of surfaces has not been attempted because of the similarity to work done by Levin [LEVIN76].

2.2.1 Determining Surface Primitives

The ability to determine efficiently surface primitives is a key step in the analysis.
2.1.1 Structuring Input Data -

The original data is divided into local areas by a regular grid of cells. Registered arrays organize the data into summary information for systematic access. Each array contains information about a different observed or computed property. All the elements of an array refer to the same observed or computed property but for different local areas. Corresponding elements in different arrays refer to the same local area. Cell boundaries are defined so that a cell contains the data points on the average. In practice, a cell may contain a variable number of points because of the data point distribution assumption. However, cells with two or less points are removed from consideration by the local analysis.

2.1.2 Types Of Properties -

Properties are recorded in arrays of byte-sized information; the information is coded into numerical values with a maximum range of 256.

2.1.3 Growing Groups Of Data Points -

The implementation of algorithms to grow groups of data points revealed many unexpected cases that required special consideration.
2.1.3.1 Quadric Surfaces -

The grouping of data points generated by underlying quadric surfaces is relatively straightforward. However, if the underlying surface is of higher order, it is a much more difficult problem to find "reasonable" groups for presentation by a quadric surface.

2.1.3.1.1 Orientation Continuity -

The goal is to characterize the shapes of local areas and identify larger areas of local shape continuity. The use of shape labels like convex, concave, flat and unknown have been shown by the example of Figure 4-4 to be inadequate. A richer set of labels which depends on the array structure has been developed. The shape label attached to a local cell depends on the properties of the local cell and its four-connected neighbors. At the lowest level, an estimate of the curvature of a curve on the surface of the object connecting the center of two adjacent cells is desired. Such a curve is defined by the intersection of the object's surface with either an X-Z or Z-Z plane. An estimate of the curvature at the mid-point boundary between the adjacent cells is computed as the difference of the average surface normals of the two cells. If implemented, the curvature is labelled as positive, zero or negative. The zero label is attached when the difference
within a tolerance of true zero. In addition, the label of unknown is required because a cell may have less than the four neighbors due to its location in the array or due to lack of sufficient data which disqualifies a cell.

By combining two estimates of curvature from opposite sides of a cell, an idea of the shape of the curve formed by the intersection of the surface with the plane is obtained. Labels reflecting estimates of the shape of two curves formed from the intersection of orthogonal planes with the surface are determined for each cell. A label representing the shape of the surface in the local area is assigned based on these two shape estimates. It should be noted that the labelling process is done conservatively. That is, the cell is labelled as mixed if any doubt exists about its shape and is removed from consideration in the growing process. Adjacent cells with similar labels are grouped together for possible representation by a single surface primitive.

2.1.3.1.2 Depth Discontinuity -

The estimate for the range of Z values within a single cell is based on the assumption that a single planar surface is present. The estimate is a function of the cell size, the average value of the Z component of the surface normal, and the range of the Z components of the surface normal. This estimate is approximate and subject to error because
surfaces are not limited to planar surfaces.

Discontinuity between adjacent cells is indicated when the magnitude of the difference of the two average Z values significantly exceeds the average of the two range values of Z. When both tests for discontinuity are used together, a reliable indication of discontinuity is obtained.

2.1.3.2 Planar Surfaces -

Identification of data points for representation by planar surface depends on finding points whose surface orientations are the same. In practice, some error is expected. Therefore, points with similar, not identical surface orientations are considered. Here, the criteria for similar is a small range of values in the range of values of the surface orientation data.

2.1.4 Surface Fitting -

The criterion for the fitting of surfaces appears at first. It is an expedient solution. However, when examined in greater detail, the mystery of why it works will can be explained.
2.1.4.1 Fitting Criteria -

The criteria for fitting surfaces has been stated. Two independent criteria for fitting the surface primitives have been developed and combined. However, the issue of the relative importance of the spatial versus the orientation information has not been addressed. The expression

\[ ||\Delta P||^2 \times ||Q||^2 \]

has been developed as an approximation to the spatial part of the quantity minimized, where \( \Delta P \) is the Euclidean distance error. Also, the expression

\[ ||Q||^2 \times \sin \theta \]

has been developed as the orientation part of the quantity minimized, where \( \theta \) is the angle between the observed and actual surface normals. The expression

\[ ||Q||^2 \times (||\Delta P||^2 + \sin \theta) \]

represents an approximation of the quantity minimized. However, the combination is questionable because the two parts are not expressed in the same units of measure. The criteria used implies an arbitrary equivalence between one unit of linear measure and one unit of angular measure. Justification for this mix of spatial and orientation information is offered. Rather, it is presented in order how explicitly the mix used. Experiments using parti...
Data of known surfaces in isolation without error show that the use of both spatial and orientation information produced better results than just spatial information alone. Small changes in the relative weights of the spatial and orientation information appear to have affected the results little.

2.1.4.2 Minimization -

The solution to the minimization problem may be imputed using a standard eigenvalue subroutine from any of many scientific subroutine libraries available. The eigenvector associated with the minimum eigenvalue is the desired solution to Equation 4-3.

2.1.4.2.1 The Wrong Point In The Right Place -

The estimation of the surface parameters is only good as the data points used in the computation. Should a number of "bad" data points be included, the result of the estimation of the surface parameters would be poor. Since points are grouped based on the average properties of each, "bad" points may be selected because a majority of "good" points mask their presence. The error associated with an individual point does not indicate absolute whether it is good or bad. However, if there are only a few bad points, the set of points with large errors includes the
it of "bad" points. By removing a subset of data points with large errors from the original set, the number of "bad" data points can be reduced, possibly to zero. This assumption assumes that the number of "bad" data points is relatively small. This strategy has two major disadvantages. Even under ideal conditions, some good data points are discarded in an attempt to remove "bad" data points. Also, an additional interface fitting is required to determine the primitive. See SCHLIER/BOLLES81 for additional ideas on how to handle similar problems.

In a practical implementation of the above strategy, there are two questions of importance to be considered. If there are no "bad" data points, how good should the fit be? The answer to this question depends mainly on the source of the three-dimensional data. Each type of sensor introduces some noise or error in the data acquisition process. A decision to apply the above strategy can be made based on the observed fit error as compared to the expected fit error. An estimate of the expected fit error may be derived from theoretical analysis or from empirical evidence. If it is necessary to apply the strategy, how many "bad" data points are there in the original set? It is assumed that the method of selecting the original set has limited the number of "bad" data points to a relatively small percentage. The estimate of expected error may be used as
side to removing points. Another approach is to assume that a fixed percentage of data points should be removed. This latter approach is the one implemented. This goal is achieved by comparing a data point's error to a threshold value. In either case, however, it is difficult to predict how the fit and residual errors are affected by the removal of data points without refitting.

In the process described above, a threshold is used to determine when to remove a "bad" point. This threshold is based on observed error computed using the error measure equation 4-1. This error measure is not absolute. However, this fact has little adverse effect so long as the data points are within the region around the origin where the threshold value was developed for use. If the data are outside this region, a different threshold value must be determined.

2.1.4.2.2 The Right Point In The Wrong Place -

The initial process used to define surfaces produces groups of data points, and each group is represented by a single surface. The groups are chosen conservatively in order to minimize the probability of points from different actual surfaces being placed in the same group. As a result, many data points near the boundaries where surfaces meet are not used in the original surface parameter.
Once a surface's parameters have been estimated, a second pass through the data points is made to find these undiscovered points. New points are included in the group if they meet the following criteria:

\[ Q(X,Y,Z)^2 < STOL \quad \text{and} \quad N(X,Y,Z)^2 < OTOL \]

where STOL is an error tolerance based on the original surface fit and OTOL is an angular error tolerance of fixed magnitude. The STOL tolerance is subject to the same problems discussed in the previous section. To insure a single connected surface, only points from cells adjacent to cells with known members are checked for new members. After the expansion is complete, a new estimation of the surface parameters is computed based on all the members of the group.

2.1.4.2.3 A Substitute For Better Resolution -

Many times, one surface may mask the presence of another surface. The underlying masked surface may not contribute a significant number of data points because only a small part of the surface is visible in the view. As with humans, a second look is helpful. This idea is implemented as an iterative process. After a surface is fitted, the points represented by it are removed from further consideration. When all the initial surface candidates have
been checked, new values for properties are computed based on the unresolved points. The removal of resolved points may clarify the type of surface to use for representing the remaining points. This new information is analyzed for additional surface candidates.

This iterative approach makes possible the identification of underlying surfaces that do not extend over a large number of cells in the registered arrays. The same results could be obtained by increasing the number of cells in the registered arrays. This increase in resolution would require a corresponding increase in the number of data points. A tradeoff between the increased processing time of the iterative approach versus the requirement of more input data can be made to achieve a given effective resolution.

3.3 View Integration

Implementation of the view integration is limited because the edge graph information is not generated by the local analysis.

3.3.1 Transformation To A Common Coordinate System -

Integration of local view information requires a change in the coordinate reference frame. This change is accomplished by expressing information in homogeneous
3.2 Surface Identification -

In the previous discussion on similarity, the need for a more intuitive measure which explicitly considers shape, orientation and location was identified. The following discussion attempts to motivate a method of determining canonical forms of quadric surfaces to achieve this goal. The canonical forms are the same as those in solid geometry [DRESDEN64]. A summary of one method is reported in [LEVIN76]. In that method, a surface defined by \( Q(X,Y,Z) \) is expressed in matrix form by

\[
p \times Q \times \text{transpose}(p) = 0
\]

where \( p \) is a point in homogeneous coordinates of the form \((X,Y,Z,1)\), and \( Q \) is a four by four symmetric matrix defined by the original coefficients as
\[
Q = \begin{bmatrix}
  a & d/2 & f/2 & g/2 \\
  d/2 & b & e/2 & h/2 \\
  f/2 & e/2 & c & j/2 \\
  g/2 & h/2 & j/2 & k \\
\end{bmatrix}
\]

The canonical matrix \( C \) is derived from \( Q \) by factorizing and translational and translational information specific to the distance of the surface into explicit matrix multipliers. The original form of

\[
p^* Q^* \text{transpose}(p)
\]

can be expressed as

\[
>^* R^* T^* C^* \text{transpose}(T)^* \text{transpose}(R)^* \text{transpose}(p)^*
\]

where \( C \) is the canonical form of \( Q \), \( R \) is a rotation matrix, and \( T \) is a translation matrix.

3.3 Description Updating -

If all surface primitives do not have closed boundaries reflected in the edge graph at the conclusion of the integration of the four primary views and their subordinated supplemental views, then closure of the description has not occurred, and some detail has been missed. To complete the description, supplemental views can be requested based on the lack of boundary closure.
4 The Final Reference Frame

The center of gravity and moments of inertia need to be determined from the input data or the global description. An approximation to this information can be computed easily. Given a voxel representation, each full voxel is considered as a point mass of unity at the center of the voxel. The computation involves sums of products. Two possible methods of obtaining a voxel representation are described.

At present, the input data for the surface description builder is obtained artificially from a modified graphics algorithm. This algorithm with additional modification can serve as the basis for generating voxel data from the global description. During the shaded image generation, a depth buffer for each pixel in the image is computed. The depth buffer is an ordered list by Z value of surface intersections with a ray parallel to the Z axis. Each pixel uses a different ray. In the case of the graphics system and the artificial data generation program, only the first entry in the depth buffer is of interest. All the entries are of interest when generating voxel data. It is possible to determine which parts of the ray are inside the volume by examining the depth buffer's entries sequentially. Initially, all the voxels are considered as empty. Each depth buffer will supply the information required to determine which voxels pierced by the ray should be filled.
By carefully selecting the size of the image and modelling space viewed, a voxel representation of desired resolution may be computed.

Another method of obtaining a voxel representation is to create it directly from the input data of the surface description builder. Initially, all the voxels in the space are considered full. As input data from a view considered, evidence is obtained that certain voxels are empty. Specifically, voxels enclosing a ray connecting camera position with a visible surface point are empty. Also, voxels enclosing a ray starting at the camera and intersecting the object are empty. Simply, each supplies information to cut away matter much like a sculptor carving a statue. The resolution in the voxel space depends on the sampling density of the input data.
The key ideas proposed have been implemented and tested. The results of that effort are presented here. The programs implemented in support of these ideas are written in FORTRAN. A 16-bit mini-computer, a PDP11/60 using an operating system, serves as the test bed. The ideas implemented and tested include the finding of groups of data points for representation by surface primitives, the fitting of surfaces to these groups via a least squares technique, the transformation of information from the viewer-dependent reference frame to a viewer-independent reference, and the identification via shape in that reference frame of the same surface from different points of view.

1 Presentation Of Input Data

The input data for each local view is presented in two forms. A pair of pictures showing the same data points in the two forms is shown in a single figure. Both forms represent a data point as a dot. A point's local X and spatial values determine its position in the picture.
brightness of the dots in the left, or upper, picture depends on the Z value of the point. Points closer to the viewer appear brighter. This method of intensity modulation is called depth cueing. The brightness of the dots in the right, or lower, picture depends on the orientation of the surface normal. Points with surface normals pointing toward the viewer appear brighter. This method of intensity modulation is referred to as orientation cueing.

2.2 Presentation Of Surface Primitives

The results of the grouping process are presented in a similar fashion to the input data. Each picture in a figure shows a group of data points that is represented by a single primitive surface. Points not in the group appear as dimmed dots. This type of display permits one to maintain a sense of perspective and structure on the whole. In addition, each picture contains a set of lines forming a grid. The grid denotes the approximate boundaries that define the local regions or cells in the registered arrays.

3.3 Test Cases

During the testing of nine objects, the programs made one error in the analysis of the thirty-six local views. There were many views with unused data points and requests for supplemental information. These requests were not...
nored, and the view integrations were done using tli
icomp
result
ticomp
result
missing informa
seen and recorded in the results of another loc
analysis and a complete global description was developed.

Requests for supplemental information resulted when tli
Deal analysis was not able to find new surface primitive
it there were remaining unused data points. There were
reasons found to cause this problem. The first reason
as that there were too few data points remaining to fit a
urface. These data points came from a surface that
peared to cover a small area in the view. They were
clustered together in one or two cells of the summary repo
and could not be identified by the shape labelling process.
The second reason was that the remaining data points were
sufficient number to define a primitive but were so spread
out over several cells in the summary report as to create
storted and inaccurate picture. The shape labelling process
was unable to identify a consistent surface. The
third reason was related to a constraint of the analysi
to prevent parts of cylinders from being misrepresented
planar surfaces that were long and narrow, these surfaces
were rejected as primitives. It seemed better to request
plementa
and look at the area in great
tail before committing to a primitive.
3.1 Flattened Sphere

The test object labelled "flattened sphere" is a sphere with two adjoining planar sections removed. The purpose of the object is to test the ability to recover simple, relatively large planar and quadric surfaces and integrate them into a consistent global description. Analyses of the four primary views were completed without difficulty. The integration of the results of these local analyses produced an accurate description of the object. The final global description consisted of one spherical primitive and two planar primitives.

Flattened Sphere / View 1 / Shaded Image

Figure 6.3.1-1
Flattened Sphere / View 1 / Input Data

Figure 6.3.1-2

Flattened Sphere / View 1 / Surface 1

Figure 6.3.1-3
Flattened Sphere / View 1 / Surface 2

Figure 6.3.1-4

Flattened Sphere / View 1 / Surface 3

Figure 6.3.1-5
Flattened Sphere / View 2 / Shaded Image

Figure 6.3.1-6

Flattened Sphere / View 2 / Input Data

Figure 6.3.1-7
Flattened Sphere / View 2 / Surface 3

Figure 6.3.1-10

Flattened Sphere / View 3 / Shaded Image

Figure 6.3.1-11
Flattened Sphere / View 3 / Input Data

Figure 6.3.1-12

Flattened Sphere / View 3 / Surface 1

Figure 6-3.1-13
Flattened Sphere / View 4 / Shaded Image

Figure 6.3.1-14

Flattened Sphere / View 4 / Input Data

Figure 6.3.1-15
Flattened Sphere / View 4 / Surface 1

Figure 6.3.1-16
3.2 Cube And Ellipsoid -

The test object labelled "cube and ellipsoid" is a cut Lth half an ellipsoid protruding from one of the cube's Lanar faces. The purpose of the object is to test the capability to grow surface primitives across narrow necks while presence of other surfaces such as in view one. The analyses of views one, two and four were completed without difficulty. In view three, a request was made for supplemental information about the area of the ellipsoid were seventeen cells covering the unidentified area and this should have been enough for the labelling process function. However, five of the cells had only one underlying data point and were discarded. In addition, further cells had only two underlying data points. These cells contained information of a doubtful nature. The shape labelling process produced several candidates for surface primitives. However, they all contained too few data points to fit a primitive accurately. The integration of the results of the local analyses produced an accurate description. The unidentified surface of view three was seen and recorded in other local results. The final global description consisted of six planar primitives and an ellipsoid primitive.
Cube and Ellipsoid / View 1 / Shaded Image

Figure 6.3.2-1

Cube and Ellipsoid / View 1 / Input Data

Figure 6.3.2-2
Cube and Ellipsoid / View 1 / Surface 1

Figure 6.3.2-3

Cube and Ellipsoid / View 1 / Surface 2

Figure 6.3.2-4
Cube and Ellipsoid / View 2 / Shaded Image

Figure 6.3.2-5

Cube and Ellipsoid / View 2 / Input Data

Figure 6.3.2-6
Figure 6.3.2-7

Cube and Ellipsoid / View 2 / Surface 1

Figure 6.3.2-8

Cube and Ellipsoid / View 2 / Surface 2
Cube and Ellipsoid / View 2 / Surface 3

Figure 6.3.2-9

Cube and Ellipsoid / View 3 / Shaded Image

Figure 6.3.2-10
Cube and Ellipsoid / View 3 / Input Data
Figure 6.3.2-11

Cube and Ellipsoid / View 3 / Surface 1
Figure 6.3.2-12
Cube and Ellipsoid / View 3 / Surface 2

Figure 6.3.2-13

Cube and Ellipsoid / View 3 / Surface 3

Figure 6.3.2-14
Cube and Ellipsoid / View 3 / Unused Data

Figure 6.3.2-15

Cube and Ellipsoid / View 4 / Shaded Image

Figure 6.3.2-16
Cube and Ellipsoid / View 4 / Input Data

Figure 6.3.2-17

Cube and Ellipsoid / View 4 / Surface 1

Figure 6.3.2-18
Figure 6.3.2-19

Figure 6.3.2-20
Cube and Ellipsoid / View 4 / Surface 4

Figure 6.3.2-21
3.3 Cylinder And Sphere -

The test object labelled "cylinder and sphere" is composed of a cylinder with one of its ends adjoined to a hemisphere. The purpose of the object is to test the ability to distinguish the change from one quadric surface to another quadric surface. The analyses of views one, two, and four was completed without difficulty. In view three an error occurred in the analysis of the planar surface that forms the bottom of the cylinder. On the first pass over the data points, the bottom of the cylinder was detected as a long, narrow planar surface and was rejected because of the cylinder restriction. The extent of the surface was not truly so but appeared in the summary as such because the adjacent cylinder masked its presence in the adjoining cells. After finding the cylinder and sphere, it attempted immediately to fit a quadric surface to the underlying planar surface points. This resulted in only half the data points being used. In the next pass after the used data points were removed and a new summary computed, it found the planar surface based on the remaining data points. The integration of the local results produced an inaccurate description with internal inconsistencies.
Cylinder and Sphere / View 1 / Shaded Image

Figure 6.3.3-1

Cylinder and Sphere / View 1 / Input Data

Figure 6.3.3-2
Cylinder and Sphere / View 1 / Surface 1

Figure 6.3.3-3

Cylinder and Sphere / View 2 / Shaded Image

Figure 6.3.3-4
Cylinder and Sphere / View 2 / Input Data

Figure 6.3.3-5

Cylinder and Sphere / View 2 / Surface 1

Figure 6.3.3-6
Cylinder and Sphere / View 2 / Surface 2

Figure 6.3.3-7

Cylinder and Sphere / View 2 / Surface 3

Figure 6.3.3-8
Cylinder and Sphere / View 3 / Shaded Image

Figure 6.3.3-9

Cylinder and Sphere / View 3 / Input Data

Figure 6.3.3-10
Cylinder and Sphere / View 3 / Surface 1

Figure 6.3.3-11

Cylinder and Sphere / View 3 / Surface 2

Figure 6.3.3-12
Cylinder and Sphere / View 4 / Shaded Image

Figure 6.3.3-15

Cylinder and Sphere / View A / Input Data
Figure 6.3.3-17

Figure 6.3.3-18
Cylinder and Sphere / View 4 / Surface 3

Figure 6.3.3-19
3.4 Cylinder And Negative Sphere -

The test object labelled "cylinder and negative sphere" is a cylinder with half a sphere removed from one end. The purpose of the object is to test the ability to identify the large concave surface of a simple object. The analyses of the first three views were completed without difficulty. View four, a request for supplemental information about the bottom area of the cylinder was made. The analysis detected and rejected a long, narrow region because of the cylindrical restriction. It failed to make an error similar to the error in the cylinder and sphere example because the planar surface was represented by a smaller number of data points. The integration of the results of these four local analyses produced an accurate description of the object. The surface not identified in view three was seen and recorded in other local results. The final global description consisted of one spherical primitive, one cylindrical primitive and one planar primitive.
Cylinder and Negative Sphere / View 1 / Shaded Image

Figure 6.3.4-1

Cylinder and Negative Sphere / View 1 / Input Data

Figure 6.3.4-2
Cylinder and Negative Sphere / View 2 / Surface 1

Figure 6.3.4-7

Cylinder and Negative Sphere / View 2 / Surface 2

Figure 6.3.4-8
Cylinder and Negative Sphere / View 3 / Shaded Image

Figure 6.3.4-9

Cylinder and Negative Sphere / View 3 / Input Data

Figure 6.3.4-10
Cylinder and Negative Sphere / View 3 / Surface 1

Figure 6.3.4-11

Cylinder and Negative Sphere / View 3 / Surface 2

Figure 6.3.4-12
Cylinder and Negative Sphere / View 4 / Surface 1

Figure 6.3.4-15

Cylinder and Negative Sphere / View 4 / Unused Data

Figure 6.3.4-16
3.5 Cube And Negative Cylinders -

The test object labelled "cube and negative cylinders" is composed of a cube with three negative cylinders aligned with the faces. The negative cylinders remove more than one quarter of the cube's volume. The purpose of the object is to test the ability to identify concave surfaces in a complex object. The analyses of the first two views were completed without difficulty. In view three, a request was made for supplemental information in the center area containing the cylindrical surface. The surface was represented by too few data points to be identified. In view four, a request was made again for supplemental information. The area of interest contained three cylindrical surfaces but their presences could not be resolved in the cell labelling process. The integration of the results produced an accurate description of the object. The surfaces not identified in views three and four were seen and recorded in other local results. The final global description consisted of six planar primitives and three cylindrical primitives.
Cube and Negative Cylinders / View 1 / Shaded Image

Figure 6.3.5-1

Cube and Negative Cylinders / View 1 / Input Data

Figure 6.3.5-2
Cube and Negative Cylinders / View 1 / Surface 1

Figure 6.3.5-3

Cube and Negative Cylinders / View 1 / Surface 2
Cube and Negative Cylinders / View 1 / Surface 3

Figure 6.3.5-5

Cube and Negative Cylinders / View 1 / Surface 4

Figure 6.3.5-6
Cube and Negative Cylinders / View 2 / Shaded Image

Figure 6.3.5-7

Cube and Negative Cylinders / View 2 / Input Data

Figure 6.3.5-8
Cube and Negative Cylinders / View 2 / Surface 1

Figure 6.3.5-9

Cube and Negative Cylinders / View 2 / Surface 2

Figure 6.3.5-10
Cube and Negative Cylinders / View 2 / Surface 3

Figure 6.3.5-11

Cube and Negative Cylinders / View 3 / Shaded Image

Figure 6.3.5-12
Cube and Negative Cylinders / View 3 / Input Data

Figure 6.3.5-13

Cube and Negative Cylinders / View 3 / Surface 1

Figure 6.3.5-14
Cube and Negative Cylinders / View 3 / Surface 2

Figure 6.3.5-15

Cube and Negative Cylinders / View 3 / Surface 3

Figure 6.3.5-16
Cube and Negative Cylinders / View 3 / Unused Data

Figure 6.3.5-17

Cube and Negative Cylinders / View 4 / Shaded Image

Figure 6.3.5-18
Figure 6.3.5-19

Cube and Negative Cylinders / View 4 / Input Data

Figure 6.3.5-20

Cube and Negative Cylinders / View 4 / Surface 1
Cube and Negative Cylinders / View 4 / Surface 2

Figure 6.3.5-21

Cube and Negative Cylinders / View 4 / Surface 3

Figure 6.3.5-22
Cube and Negative Cylinders / View 4 / Unused Data

Figure 6.3.5-23
3.6 Ice Cream Cone -

The object labelled "ice cream cone" is composed of a cone with its base adjoined to a hemisphere. The purpose of the object is to test the ability to identify the smooth change from one quadric surface to another quadric surface. The analyses of the four primary views were completed without difficulty. The integration of the results produced an accurate description of the object. The final globe description consisted of one spherical primitive and one conic primitive.

Ice Cream Cone / View 1 / Shaded Image

Figure 6.3.6-1
Figure 6.3.6-2

Figure 6.3.6-3
Ice Cream Cone / View 1 / Surface 2

Figure 6.3.6-4

Ice Cream Cone / View 2 / Shaded Image

Figure 6.3.6-5
Ice Cream Cone / View 2 / Input Data

Figure 6.3.6-6

Ice Cream Cone / View 2 / Surface 1

Figure 6.3.6-7
Ice Cream Cone / View 3 / Input Data

Figure 6.3.6-10

Ice Cream Cone / View 3 / Surface 1

Figure 6.3.6-11
Ice Cream Cone / View 3 / Surface 2

Figure 6.3.6-12

Ice Cream Cone / View 4 / Shaded Image

Figure 6.3.6-13
Ice Cream Cone / View 4 / Surface 2

Figure 6.3.6-16
3.7 Barbell -

The test object labelled "barbell" is composed of three connected parts. The two end parts are spheres, and the center part is a circular cylinder. The purpose of this object is to test the integration process with similar taped primitives in different locations. The analyses of the four primary views were completed without difficulty. The integration of the results produced an accurate description of the object. The final global description insisted of two spherical primitives and a cylindrical primitive.

Barbell / View 1 / Shaded Image

Figure 6.3.7-1
Barbell / View 1 / Input Data
Figure 6.3.7-2

Barbell / View 1 / Surface 1
Figure 6.3.7-3
Barbell / View 2 / Shaded Image

Figure 6.3.7-6
Figure 6.3.7-8

Figure 6.3.7-9
Barbell / View 2 / Surface 3

Figure 6.3.7-10

Barbell / View 3 / Shaded Image

Figure 6.3.7-11
Figure 6.3.7-12

Figure 6.3.7-13
Figure 6.3.7-14

Figure 6.3.7-15
Barbell / View 4 / Shaded Image

Figure 6.3.7-16

Barbell / View 4 / Input Data
Figure 6.3.7-18

Barbell / View 4 / Surface 1

Barbell / View 4 / Surface 2
Figure 6.3.7-20

Barbell / View 4 / Surface 3
3.8 Mug -

The test object labelled "mug" is composed of a handle and a cup part. Two cylinders, one positive and one negative form each part. For the cup part, the cylinders are arranged to form a closed bottom of the mug. For the handle, the negative cylinder completely removes the central section of the positive cylinder to form a hole. The purpose of the object is to test the ability to describe a complex object with a hole in it. The analyses of the four primary views did not use all the data points in any case. In all views, requests were made for additional information. In view one, the data lacked sufficient number of cells to permit the shape labelling process to identify the quadric surfaces forming the outside of the cup and handle. In view two, there were not enough data to identify the inside of the handle. In view three, the side of the handle appeared as a long, narrow plane surface and was rejected because of the cylinder restriction. In view four, separate requests for supplemental information were made corresponding to the areas of the bottom outside and the inside of the handle. In both cases there were too few data points for identification. The integration of results produced an incomplete description of the object. The inside cylindrical surface of the handle was not identified in any view. The final global description consisted
Three cylindrical primitives and five planar primitives.

Mug / View 1 / Shaded Image

Figure 6.3.8-1
Mug / View 1 / Input Data

Figure 6.3.8-2

Mug / View 1 / Surface 1

Figure 6.3.8-3
Mug / View 1 / Surface 2

Figure 6.3.8-4

Mug / View 1 / Surface 3

Figure 6.3.8-5
Mug / View 1 / Unused Data

Figure 6.3.8-6

Mug / View 2 / Shaded Image
Mug / View 2 / Input Data

Figure 6.3.8-8

Mug / View 2 / Surface 1
Figure 6.3.8-10

SURFACE 2

Mug / View 2 / surface 2

Figure 6.3.8-10

SURFACE 3

Mug / View 2 / surface 3

Figure 6.3.8-11
Mug / View 2 / Unused Data

Figure 6.3.8-12

Mug / View 3 / Shaded Image

Figure 6.3.8-13
Mug / View 3 / Input Data

Figure 6.3.8-14

Mug / View 3 / Surface 1
Mug / View 3 / Surface 2

Figure 6.3.8-16

Mug / View 3 / Surface 3
Mug / View 3 / Unused Data

Figure 6.3.8-18

Mug / View 4 / Shaded Image

Figure 6.3.8-19
INPUT DATA

Mug / View 4 / Input Data
Figure 6.3.8-20

SURFACE 1

Mug / View 4 / Surface 1
Figure 6.3.8-21
Mug / View 4 / Unused Data

Figure 6.3.8-24

<table>
<thead>
<tr>
<th>Column 1</th>
<th>Column 2</th>
<th>Column 3</th>
<th>Column 4</th>
<th>Column 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data 1</td>
<td>Data 2</td>
<td>Data 3</td>
<td>Data 4</td>
<td>Data 5</td>
</tr>
<tr>
<td>Data 6</td>
<td>Data 7</td>
<td>Data 8</td>
<td>Data 9</td>
<td>Data 10</td>
</tr>
<tr>
<td>Data 11</td>
<td>Data 12</td>
<td>Data 13</td>
<td>Data 14</td>
<td>Data 15</td>
</tr>
<tr>
<td>Data 16</td>
<td>Data 17</td>
<td>Data 18</td>
<td>Data 19</td>
<td>Data 20</td>
</tr>
<tr>
<td>Data 21</td>
<td>Data 22</td>
<td>Data 23</td>
<td>Data 24</td>
<td>Data 25</td>
</tr>
<tr>
<td>Data 26</td>
<td>Data 27</td>
<td>Data 28</td>
<td>Data 29</td>
<td>Data 30</td>
</tr>
</tbody>
</table>

UNUSED DATA
5.9 Telephone Handset -

The test object labelled "telephone handset" is composed of a mouth piece and an ear piece connected by a middle approximated with an elliptic cylinder. The mouth and ear pieces are ellipsoids with single planar sections; moved to flatten an area. In views three and four, only one of the planar sections is visible because of the viewing parameters. The ability to describe a telephone was the motivation for this work. The analyses of the four primary views were completed without difficulty. The integration of the results of these local analyses produced an accurate description of the object. The final global description consisted of two planar primitives, two ellipsoidal primitives and a cylindrical primitive.
Figure 6.3.9-1

Figure 6.3.9-2
Figure 6.3.9-3

Figure 6.3.9-4
Telephone Handset / View 1 / Surface 3

Figure 6.3.9-5

Telephone Handset / View 2 / Shaded Image

Figure 6.3.9-6
Telephone Handset / View 2 / Input Data

Figure 6.3.9-7

Telephone Handset / View 2 / Surface 1

Figure 6.3.9-8
SURFACE 2

Telephone Handset / View 2 / Surface 2

Figure 6.3.9-9

SURFACE 3

Telephone Handset / View 2 / Surface 3

Figure 6.3.9-10
SURFACE 4

Telephone Handset / View 2 / Surface 4

Figure 6-3.9-11

SURFACE 5

Telephone Handset / View 2 / Surface 5

Figure 6.3.9-12
Telephone Handset / View 3 / Shaded Image

Figure 6.3.9-13

Telephone Handset / View 3 / Input Data

Figure 6.3.9-14
Figure 6.3.9-15

Telephone Handset / View 3 / Surface 1

Figure 6.3.9-16

Telephone Handset / View 3 / Surface 2
Telephone Handset / View 3 / Surface 3

Figure 6.3.9-17

Telephone Handset / View 3 / Surface 4

Figure 6.3.9-18
Telephone Handset / View 4 / Shaded Image

Figure 6.3.9-19

INPUT DATA

Telephone Handset / View 4 / Input Data

Figure 6.3.9-20
Telephone Handset / View 4 / Surface 1

Figure 6.3.9-21

Telephone Handset / View 4 / Surface 2

Figure 6.3.9-22
This chapter presents conclusions drawn from experience gained in doing the work reported. Specific conclusions and suggestions about the implementation are presented first. Conclusions about general issues in representation are presented next. Finally, suggestions for further study are presented.

7.1 Specific Conclusions

The analysis of the examples presented showed success of the use of the registered arrays which sum input data. These arrays permit data to be organized used effectively in a hierarchical manner. This increases the efficiency of computation. However, the for greater resolution is seen also. The resolution required is a function of the complexity of the viewed. The proposed model builder does a hierarchal analysis. It starts with a coarse resolution and when detail is needed, it requests supplemental views of resolution but limited domain. To prevent repeated for views of the same general area at greater and 194
solution, it appears that the approximate four hundred points and the eighty local cells be minimal. Using two to five times as much information should improve the situation without an undue increase in computational cost.

The use of the average and standard deviation values to summarize observed properties is shown to be a useful method to obtain reliable information in a noisy world. It is an double edged sword, however. The worse the input data quality or the greater the desired accuracy of the result! the larger the amount of input data required. There are infinite practical limits on the accuracy and amount of information that can be collected. Ambitious application always test the limits of the available technology to collect more and better information.

The identification of data points to be grouped together for representation by a primitive based on both spatial and orientation information is supported by the test results. The implementation of the assumptions used in identification is not the best, but it works for the objects examined. Specifically, the shape labelling process could be improved. Only qualitatively different shapes are differentiated in the current implementation. As long as the underlying surface is, at most, quadric, no trouble occurs. However, if the underlying surface is more complex and more than one surface primitive is required to model i
The shape labelling process may not give adequate results whether to break the complex surface at places of high curvature is unexplored in this work because the current labelling scheme does not convey sufficient quantitative information about the local surface shape.

Various criteria for fitting surfaces were considered before settling on the two used. One criterion used is related to spatial information, and the other criterion is related to orientation information. They are considered in isolation of each other. When no error in the data is present, the results are predictable. However, it is difficult to predict the effect of input data error on the fitting of surfaces. In chapter five, the mix of spatial and orientation criteria used was shown explicitly. There is a dissimilarity in the units of measure of the two criteria, and an implicit equivalence was defined in an ad hoc fashion. Experiments using different weights were inconclusive as to the best mixture. The fact that the combined criteria proved superior to just the spatial criterion suggests that the spatial and orientation data are complementary in nature. However, this idea seems to be contradicted by the fact that orientation information can be approximated from spatial information.
The identification of the same surface from different views has been studied. The need to avoid classification schemes that use rigid, absolute criteria to determine identification is shown. An ad hoc similarity measure that uses the quantitative surface coefficients has been implemented. The identification of the same surface from different views works when there is little error in the estimated surfaces. However, the measure's performance in a noisy environment is difficult to predict in terms that are geometrically intuitive. Another method is proposed that makes a distinction between location and shape information. It is hoped that by explicit separation, the role of each type of information can be clarified. However, the proposed method remains untested.

1.2 General Conclusions

The use of multiple views, while not new [UNDERWOOD/COATES75], is unusual in the three-dimension analysis of scenes. In the past, use of multiple views has overlooked the concept of closure. Only when the analyst is committed to obtaining an object description based on complete information does closure become important. Just as fitting the last piece of a jigsaw puzzle in place unit all the pieces into one picture, closure guarantees that the information needed to create a complete and consistent
object description is available. The analysis, knowing fact, can check the final description and insure that are no holes, like a missing edge, and discontinuities two surfaces not terminating cleanly at an Unfortunately, this idea is untested because the analy edge graph information was not implemented.

The importance of using both spatial and orient information in a representation is confirmed, representation that does not consider both type? information will have representational flaws that disqualify it from use in some applications. This work motivated initially to study the Gaussian image intermediate representation for use in buildin description of an object [SMITH79], [BAJCSY80]. Gaussian image ignores spatial information totally; a result, many objects are mapped to the same Gat description. Some ed[ hoc method of augmenting the Gat image representation to compensate for the lack of s] information could be proposed. However, such a pr would be like placing a small bandage over a gaping W This representation was abandoned in favor of the regi arrays that are capable of treating both spatial orien information in an integral f; [DANE/BAJCSY81].
The need for a primitive expressed in a canonical form has been highlighted in the discussion of the measure of similarity. The ability to separate information defining shape from spatial information defining location is required if results that have intuitive geometric interpretation are expected. [HINTON8i] cites evidence reported in the psychology literature that supports the idea that humans use a "canonical, object-based" reference frame in the description of three-dimensional objects. The canonical form also serves to simplify the problem of recognition.

The use of a viewer-independent coordinate system for the object's description is a feature of the model builder. Non-essential, viewer-dependent information is not incorporated in the description. In order to accomplish this fact, the relationships between the various views are needed to establish a common reference frame. Ideally, the final object-centered coordinate system should permit high-level processing, such as finding symmetries, to be done easily and allow results to be expressed concisely.

What can be achieved by using the proposed model builder? A surface description of an object is constructed from which it is possible to estimate global properties such as volume, structural symmetries, and positional stability. The model can serve as an intermediate step in the derivation of other types of representation from the raw data.
data. A global smoothing of raw data occurs as an effect in the process of building the model. This smoothing has beneficial effects on data obtained for reconstruction based on the model. Such a reconstruction algorithm can be used as a source of information for input to a graphics display or CAM system.

7.3 Future Work

During the implementation and testing of programs, the use of artificial data proved to be invaluable for developing the ideas for the programs and exposing weaknesses in the proposed algorithms. However, before the ideas the programs seek to just be considered proven, additional tests must be run using real data. Without this real test, the abilities of the programs to perform to specification cannot be taken for granted. The use of both real and artificial data in the development process of programs is important and type of testing should be neglected.

There are two distinct areas where future work will be done. The first area is categorized as doing or work to build descriptions of objects in the real world which will help the model builder function in a real world environment. The suggestions presented previously should be implemented. The need to test the surface fitting technique and implement a new measure of similarity for identification...
purposes is especially important. It is suggested strongly that future work dealing with three-dimensional data not be attempted on a computer with a 32k address space. While it is not impossible, it does make implementation difficult.

The implementation of edges as the mathematical intersection of primitives and the confirmation of the existence of the edges is a major project. It is a well-understood problem with solutions suggested by others [EVIN76], [SHAPIRO/FREEMAN78]. Once completed, the local analysis can be expanded to generate edge graph information. The availability and use of this information would strengthen the integration step and permit the implementation of the closure analysis to evaluate the completeness of the global description. Beyond this work, there are two rather large, uncharted regions to be explored. The first is in the area of recognition. Given an object-centered model, how effectively can it be used in recognition, graphics display, or CAD/CAM? The second is in the area of strategies. What are the criteria for acquiring more data? What are the criteria for analyzing the existing data more? What is the optimal mix to obtain a solution to a given problem?
This work has answered some questions and represented the existing knowledge. But it has revealed many new questions and perspectives. In this respect, it is a success.
# INDEX

<table>
<thead>
<tr>
<th>Term</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>closure</td>
<td>61, 87</td>
</tr>
<tr>
<td>corners</td>
<td>31–33, 60</td>
</tr>
<tr>
<td>data points</td>
<td>39, 81, 83</td>
</tr>
<tr>
<td>data, hierarchy</td>
<td>27, 45, 92, 194</td>
</tr>
<tr>
<td>data, input</td>
<td>4, 9, 37, 44, 69, 76, 90</td>
</tr>
<tr>
<td>data, input, artificial</td>
<td>38, 70</td>
</tr>
<tr>
<td>data, input, real</td>
<td>38, 72</td>
</tr>
<tr>
<td>edge graph</td>
<td>31</td>
</tr>
<tr>
<td>edges</td>
<td>31–33, 60</td>
</tr>
<tr>
<td>global reference frame</td>
<td>37, 63, 68, 88, 198</td>
</tr>
<tr>
<td>least squares</td>
<td>58</td>
</tr>
<tr>
<td>orientation information</td>
<td>10, 51, 54, 57, 71, 77, 90</td>
</tr>
<tr>
<td>record, corner</td>
<td>33</td>
</tr>
<tr>
<td>record, edge</td>
<td>33</td>
</tr>
<tr>
<td>record, object</td>
<td>32</td>
</tr>
<tr>
<td>record, surface primitive</td>
<td>32</td>
</tr>
</tbody>
</table>
Representation . . . . . . . . 2, 16
Representation, canonical . . 20, 86, 198
Representation, voxel . . . . 68, 88
Resolution . . . . . . . . . . 84, 92, 195

Similarity . . . . . . . . . 65, 88, 156, 196
Spatial information . . . . . 13, 52, 54-55, 78, 90
Surface primitives . . . . . . 28, 32, 90, 195
Surface properties . . . . . 46
Surface types . . . . . . . . 64

View . . . . . . . . . . . . . 39, 42, 74
View, primary . . . . . . . . 40, 75, 91
View, supplemental . . . . . 40-41, 75, 92
Views, multiple . . . . . . . 4, 22, 40, 63, 197
REFERENCES

[AGIN/BINFORD73]

[BADLER/BAJCSY78]

[BADLER/DANE79]

[BAJCSY80]
[BALLARD/BROWN/FELDMAN78]

[BALLARD/SABBABH81]

[BARROW/TENENBAUM79A]

[BARROW/TENENBAUM79B]

[BINFORD71]


[DANE/BAJCSY81]

[DERISI81]

[DRESDEN64]

[DUDA/HART73]

[DUDA/NITZAN/BARRETT79]
[FAUX/PRATT79]


[FISCHLER/BOLLES81]


[FREEMAN/POTMESIL79]

Freeman, H., and Potmesil, M., "Curved Surface Representation Utilizing Data Extraction from Multiple Photographic Images," Workshop on Representation of Three-Dimensional Objects, University of Pennsylvania, Philadelphia, Pennsylvania (May 1979), H.

[FREUDER76]


[FULKS69]

[GENNER77]


[GENNER79]


[GOLDSTE/NA71]


[GRIMSON79]


[GRIMSON80]


[HORN75]

[HORN77]

[HORN/WOODHAM/SILVER78]
Horn, B. K. P., Woodham, R. J., and Silver, "Determining Shape and Reflectance Using Multiple Images" AIM-490, MIT Artificial Intelligence Laboratory, MIT (1978).

[HORN79]

[HUFFMAN71]
[KEUCHI80]
Keuchi, K., "Shape from Regular Patterns ( An Example of Constraint Propagation in Vision )," in the Proceedings of the Fifth International Conference on Pattern Recognition, Miami Beach (December 1980), 1032-1039.

[ANADE78]

[ANADE79]

[ELLY71]

[ENDER77]
ender, J. R., "Shape from Texture: An Aggregation Transform that Maps a Class of Textures into Surface Orientation," in the Proceedings of the Sixth International Joint Conference on Artificial Intelligence (August 1979), 45-480.
[LEVIN76]

[LEVIN79]

[MARR/NISHIHARA77]

[MARR/POGGIO76]

[MARR/POGGIO77]
[MOHR81]

[MORAVEC79]
Moravec, H. P., "Visual Mapping by Robot Rover," in Proceedings of the Sixth International Joint Conference Artificial Intelligence, Tokyo, Japan (August 598-600.

[NITZAN/BRAIN/DUDA77]

[O’ROURKE/BADLER79]
O’Rourke, J., and Badler, N., "Decomposition of Three-Dimensional Objects into Spheres," IEEE Transactions on Pattern Analysis and Machine Intelligence (July 295-305.

[O’ROURKE81]
[OSHIMA/SHIRAI79]

[OSHIMA/SHIRAI81]

[POTMESIL79]

[REQUICHA80]

[ROBERTS65]
[GERS/ADAMS76]


[OSENTHAL78]


[HAPIRA/FREEMAN77]


[HAPIRO/e_t.a_1.81]

[SLOAN77]

[SMITH79]

[SOROKA79]

[STEVENS79]

[STRAUSS80]
[UNDERWOOD/COATES75]

[ALTZ75]

[LL/PENNINGTON72]

[ITKINS80]

[DOLF74]
[WOLFELD81]

[WOO77]

[WOODHAM77]

[WOODHAM78]