COMPUTING PARAMETRIC GEON DESCRIPTIONS OF 3D MULTI-PART OBJECTS

Kenong Wu

Department of Electrical Engineering
McGill University

April 1996

A Thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfilment of the requirements of the degree of
Doctor of Philosophy

Abstract

A new approach for computing qualitative part-based descriptions of 3D objects from single- and multi-view range data is presented. This research is motivated by both a theory of human image understanding (Recognition-by-Components) and the need for qualitative recognition by an autonomous robot in order for it to efficiently interact with its environment.

Object descriptions are obtained in two consecutive steps: (1) object segmentation into parts and (2) part model identification. Segmentation is achieved by first computing the simulated electrical charge density distribution on a tessellated triangular mesh of the object surface. The algorithm then detects the object part boundaries where the the charge density achieves a local minimum. The charge density distribution can simultaneously provide an indication of the gross and fine object structures. Parametric geons are introduced as the part models, which indicate both qualitative shape and quantitative attribute information. Model recovery is achieved by fitting all parametric geons to a part and then selecting the best model based on the minimum fitting residual. A new objective function used for model recovery is optimised by a global optimisation technique (Very Fast Simulated Re-Annealing).

The advantages of this approach are demonstrated through experimentation. By using a physical analogy to the well known transversality principle, part segmentation does not require an assumption of surface smoothness or the choice of a particular scale to compute local surface features. The formulation for parametric geons provides a global shape constraint, which ensures reliable part model recovery even when the part shape is not an exact instance of a parametric geon. By directly comparing a part with all candidate models, this approach explicitly verifies the shape of the resultant part descriptions. The computed part-based descriptions are well suited for the object recognition task carried out by an autonomous robot.

Résumé

Ce travail présente une approche pour le calcul qualitatif de la description en parties d'objets 3D, perçus à partir d'une ou de plusieurs vues de distance. Cette recherche est motivée à la fois par la théorie de compréhension des images de l'être humain (reconnaissance par composantes) et le besoin d'une reconnaissance qualitative permettant à un robot autonome d'interagir efficacement avec son environnement.

Les descriptions d'objets sont obtenues en deux étapes: (1) la segmentation d'objet en parties et (2) l'identification de ces parties. La segmentation est accomplie en calculant tout d'abord la simulation de distribution de densité de charge sur tesselation triangulaire de la surface de l'objet. L'algorithme détecte alors les frontières des composantes de l'objet, dont la densité de charge atteint un minimum local. La distribution de densité de charge peut également fournir une indication sur les structures fines et grossières de l'objet. Les Géons paramétriques seront le modèle des parties d'objet, et indiqueront à la fois une information qualitative de forme et une information quantitative. La sélection du modèle d'une partie est obtenue par la mise en correspondance des mesures et des Géons paramétriques et en sélectionnant le modèle qui obtient l'erreur résiduelle minimale. Une nouvelle fonction objectif utilisé pour la reconnaissance de modèle est optimisée par une technique d'optimisation globale (Recuit simulé très rapide).

Les avantages de cette approche sont démontrés par l'expérimentation. En utilisant l'analogie physique bien connue du principe de transversalité, la segmentation en partie ne nécessite pas d'hypothèse de surface lisse ou le choix d'une échelle particulière pour calculer les caractéristiques locales de la surface. La formulation des Géons paramétriques fournie une contrainte globale de forme qui assure une identification fiable du modèle d'une partie même lorsque la forme de cette dernière n'est pas une instance exacte d'un Géon paramétrique. Par une comparaison directe d'une partie avec tous les modèles candidats, cette approche vérifie explicitement la forme des descriptions en partie résultantes. Les description en parties ainsi calculées s'avèrent bien adaptées pour les tâches de reconnaissance d'objet requises par tout robot autonome.

This dissertation is dedicated to my parents

Feng Zhi-Zhong *and* **Wu De-Qing**

and to my wife

Jing

Acknowledgement

Many people have provided guidance, assistance and encouragement during my study, research and thesis writing. First and foremost, I would like to thank my supervisor Professor Martin D. Levine for his persistent guidance, support and encouragement. His insightful advice and valuable criticism made the completion of this work possible.

I would like to thank Professor Lester Ingber, the originator of the Very Fast Simulated Re-Annealing algorithm. He provided the software, guidance for using his software and many comments on global optimisation techniques. I would also like to thank Professor Jonathan Webb of the Department of Electrical Engineering at McGill University, who lead me into the field of electrostatics and provided valuable advice on charge density computations.

It was a pleasure to be a member of the Centre for Intelligent Machines at McGill University. The stimulating research environment and friendly atmosphere made my latest student life pleasant. I am thankful to Professor Frank F. Ferrie and his graduate students for developing 3D vision software tools, which made my programming work much easier. I wish to thank Duncan Baird, Don Bui, Thierry Baron, Gerard Blais, Marc Bolduc, Michael Kelly, Michael Langer, Kaleem Siddiqi, Gilbert Soucy, Peter Whaite and John Zelek for their technical assistance and many discussions. My thanks are also due to Norma Procyshyn and Ornella Cavaliere for their administration assistance.

I would like to express my deepest gratitude to my parents, who encouraged and supported me in my pursuit of my career goals. Many thanks are due to my wife, Jing, for her love, understanding, encouragement and patience.

I wish to thank Douglas DeCarlo, Professor Demitri Metaxas (GRASP Lab, University of Pennsylvania), and Dr. Marc Soucy (InnovMetric Software Inc.) for kindly providing triangular meshes for the rangefinder data. Certain range images used in this thesis were obtained from the PRIP Lab at the Michigan State University under the direction of Professor Anil K. Jain.

This research was partially supported by a Natural Sciences and Engineering Research Council of Canada Strategic Grant and an FCAR Grant from the Province of Quebec.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

To interact with the environment, an autonomous robot must identify targets and avoid obstacles in its environment. More specifically, the robot must often recognise and localise objects by interpreting sensor data in terms of previously existing knowledge about the objects. This task requires image data, object models in a database, an *object representation* process that computes a description of the object from image data and an *object recognition* process that matches the description against object models. Sensor data usually only provide point-by-point measurements such as the distances from the sensor to objects in the viewed scene. However, these *numerous* and *unstructured* data are not appropriate to represent the environment for a mobile robot executing complicated roles in, say, an office or factory milieu. Such a robot must make use of symbolic models which are *concise* and *organised* descriptions about the structure of the world. Thus the object representation process is important because it transforms sensor data to symbolic descriptions which are consistent with the models in a database and support efficient model matching. This signal-to-symbol mapping is, thus, at the heart of any functioning autonomous robot carrying out complex tasks.

In this thesis, we present a new approach to three-dimensional (3D) shape representation of objects based on parts. The input to our system is a single range image or multiple range images. Our task is twofold. The first is to segment the object into individual parts. The second is to select a particular part model from a few predefined model candidates which describes the best *shape approximation* of each segmented object part. Our approach first computes a novel *physics-based* surface property, *the simulated charge density distribution* over the object surface and decomposes the object into parts at the part boundary where the charge density achieves a local minimum. We then employ a new *top-down* strategy to recover models for each segmented part by *directly* comparing the part shape with

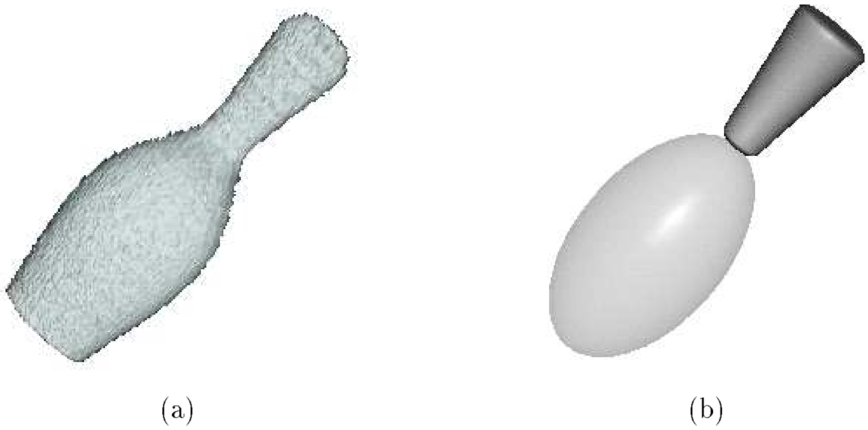


FIGURE 1.1. Part-based description of a 3D object. (a) The range image of an object. The shading is added artificially to enhance the 3D nature of the object. (b) The data in (a) are interpreted by our system to consist of two parts. Each part is approximated by an ideally-shaped volumetric model. In this case, the top part is a tapered cylinder and the bottom part is an ellipsoid.

the shapes of known part models. The segmentation method efficiently deals with certain problems in traditional approaches, such as unrealistic assumptions about surface smoothness and instability in the computation of the surface feature. The model recovery method obtains part models more robustly and accurately than previous work.

Consider a simple example of a 3D object that could be imaged by a laser rangefinder mounted on a mobile robot. Figure 1.1 (a) shows the *range image* of a bowling pin. Here the value at each pixel indicates the distance from a position on a regularly spaced grid in the sensor image plane to a point on the object surface in the scene. Our approach produces an object model, which simply consists of two generic shapes, an ellipsoid and a tapered cylinder, as shown in Figure 1.1 (b). This model is a coarse description of the object. It reveals that the object consists of two intuitive *parts* described by *ideal shape symbols or models.* The part models are also associated with certain parameters which describe their size, position and orientation, as well as the tapering rates for the top part. This kind of description reflects the object structure at a scale that is similar to our intuitive notion of a part [**93**]. It encodes symbolic shape information for object parts at the qualitative level. All part model shapes are ideal and distinctive; the number of shape types of part models is finite. The descriptions also include size, deformation and pose information at the quantitative level. Restrictions on an object's shape are applied solely to parts. Although their number is indeed limited, the composition of these part models using various simple spatial relations can represent many objects [**15**]. Such an object description can support symbolic reasoning for an autonomous robot which needs to recognise objects in its environment.

1. Background

In order to recognise an object from a range image, a computer vision system must convert the image data into a symbolic description or model of the object shape which is somehow consistent with the actual object shape. Two major questions need to be addressed in this regard. First, at what level of description should object shapes be represented? This issue is related to the domain of perceptual psychology [**60, 136, 15, 93**]. The description must reflect structural information in nature and permit efficient matching to models. Secondly, how can one derive such a description robustly? This question is of primary interest to the field of computational vision [**82, 6, 143**]. The answers to these questions serve as the constraints for and motivate the development of computer vision algorithms.

The significance of object descriptions at the part level is well understood [**82, 72, 60, 136, 80**]. Many objects consist of parts or components which can be distinguished perceptually, geometrically or functionally from each other. Object parts have perceptual salience and reflect the natural structure in the world [**6**]. Building object part-based descriptions for various tasks has been a major strategy for many years [**87, 82, 94, 6, 95, 46, 133, 41, 65, 13**]. The primary reason for this is that part-based descriptions help to bridge the gap between image features and symbolic descriptions of objects [**96**]. Therefore they can be more robustly and efficiently indexed into a database than some other features, such as edges or surfaces. Parts are also advantageous for representing non-rigid objects [**60**]. Moreover, part descriptions also support function-based object recognition [**129, 130**]. In this case, objects are specifically identified by a certain functionality, which is most likely associated with the object's parts rather than its edges or surface patches.

It is a common observation of nature that a variety of objects can be constructed from a few simple primitive prototypes. In the art of sculpture, a few volumetric shapes are regarded as fundamental units for constructing sculptures [**154**]. Three basic colours suffice to make different pointillistic paintings [**75**]. Twenty-six letters make up the list of all English characters. We need only 44 phonemes to code all of the words in the English language [**15**]. In an analogous fashion to the phonemes in speech, Biederman has proposed a theory of human image understanding for the early identification of objects, the so-called Recognition-By-Components (RBC) [**15**]. He has proposed 36 distinct volumetric shapes,

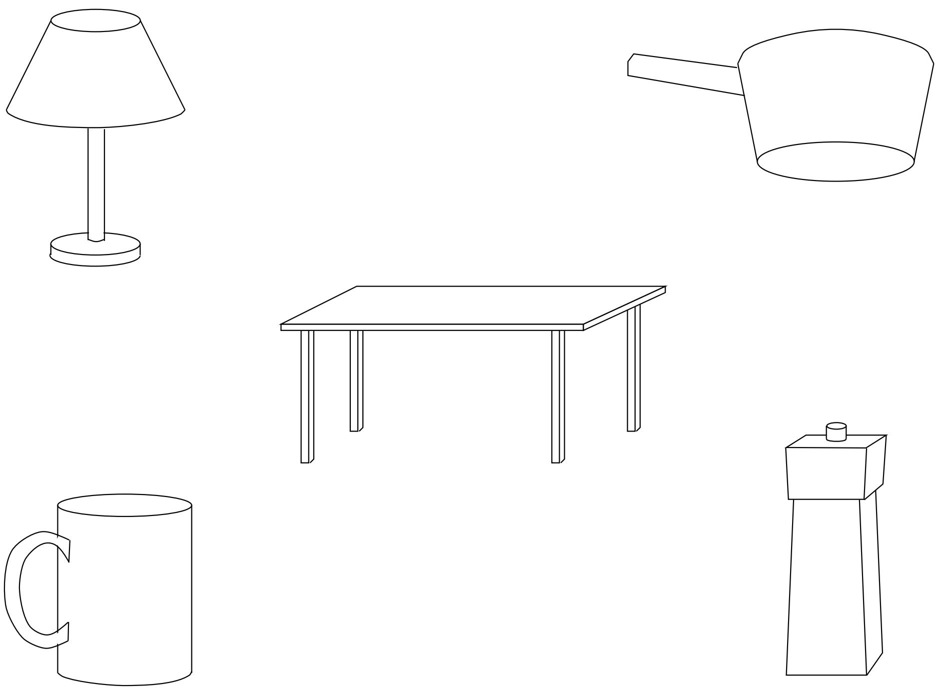


FIGURE 1.2. Objects described by a few geons.

called *geons* (geometrical ions) as descriptions for object parts. The shapes of geons, such as a block or a cylinder, are *generic* in the sense that they are defined qualitatively and can be combined to describe many different objects. Line drawings of several examples are given in Figure 1.2. Biederman postulated that if an arrangement of a few geons could be recovered from line drawings of an object such as one of those shown in Figure 1.2, the object could be recognised quickly, even when occluded, novel, rotated or degraded. He and his colleagues have conducted psychophysical studies [**15, 18, 19, 20, 33, 21**] and created artificial neural network simulations [**65**] to support geon-based recognition. The RBC proposal that an object's part-based representation can be constructed using a finite set of generic shape prototypes is the essential motivation behind our research. In this thesis, we focus on several important issues regarding the computational aspects related to this theory.

2. Statement of the Problem

The specific problem we consider in this thesis is to segment an isolated 3D object into its parts and describe the shape of each part in terms of a few generic part models. The objects are composed of multiple parts. It is important to note that the object surfaces need not be smooth and the shapes of object parts need not fully conform to the shapes of the part models.

RBC hypothesises that line drawings of objects extracted from intensity images by an edge detection operation [**30**] are used to derive geon-based descriptions [**15**]. However, edges and line drawings are significantly different. Edges reveal image information which reflects the rapid changes in image intensity. But line drawings generally convey information about object surface discontinuities in orientation and depth. In practice, "clean" line drawings can rarely be obtained from edge maps due to the colour and texture of object surfaces or complex illumination configurations. All studies based on line drawing analysis have assumed that complete line drawings can be extracted and that the objects of interest are composed of instances of complete geons. Unfortunately, it is not clear how to derive geons from imperfect line drawings of objects which consist of non-ideally shaped parts. As an alternative we use range rather than intensity images. We believe that the depth information provided by range images can facilitate the extraction of geons from objects which consist of imperfect geon-like parts. In many situations, an autonomous mobile robot is equipped with a laser rangefinder. It is therefore important to study how to derive symbolic descriptions of objects using range information. These could then be used as the basis for object recognition.

A *range image* is defined as a set of M discrete samples of a scalar function $f\colon \mathcal{I}^2 \rightarrow \mathcal{R}$

$$z_i = f(\mathbf{u}_i)$$

where $\mathbf{u}_i \in \mathcal{I}^2$ is the index of the 2D image grid, $z_i \in \mathcal{R}$ and $i = 1, ..., M$. A range image gives the distances between the image plane and the points on the surface of objects in the scene. By consulting a lookup table that indicates the relationship between the image coordinate system and the rangefinder coordinate system, a range image can be further converted to *range data*. These are defined as a set of M discrete samples of a vector function $\mathbf{f}\colon \mathcal{I}^2 \rightarrow \mathcal{R}^3$

$$\mathbf{d}_i = \mathbf{f}(\mathbf{u}_i)$$

where $\mathbf{d}_i \in \mathcal{R}^3$ and $i = 1, ..., M$. Range data provide explicit 3D coordinates in the rangefinder coordinate system. Our system starts with range data and applies the RBC theory to the range data rather than to line drawings as in the original proposal [**15**]. In RBC, geons are defined by qualitatively specifying the 3D properties of generalised cylinders [**22**] which are parameterised volumetric models. This concept is clearly not restricted just to line drawings of objects and can be applied to range data. However, the technique for actually deriving part-based descriptions from range data is very different from Biederman's computational hypothesis. More importantly, some difficult problems which exist in the approaches based on line drawing analysis can be efficiently dealt with by using range data. This will be elaborated in the next few paragraphs and emphasised in Chapter 3.

Some terminology used in this thesis is worth explaining at this point. A *physical surface* is the boundary between the space occupied by an object and free space. The positions on the object physical surface are encoded as *surface data points* in range images or range data. By a *physical part*, we mean a portion of the object surface which can be distinguished meaningfully [1] from the rest of the object. In the context of part segmentation, a part is a set of surface data points on a physical part.

RBC proposes that a part-based description includes part models, geons, and spatial relations between parts [**15**]. Both geons and the spatial relations are described in pure qualitative terms. Our part-based description of an object is composed of the number of parts and the descriptions of part models. Each part model includes its shape type, its pose, its size, the tapering rates if it is a tapered shape and the axis curvature if it is a curved shape. As an example, Table 1.1 shows the part-based description of the bowling pin shown in Figure 1.1. The shape type for each part model is given in qualitative terms and other features are described quantitatively. In our case, the spatial relationship between parts is encoded in the pose information, which can be converted into a qualitative relation. To obtain such a description, one needs to know: (1) Which are the parts? and (2) What is the model for each of the parts? The former is the issue of *part localisation (object segmentation)*, while the latter deals with *part identification (model recovery)*.

The problem of object segmentation can be stated as follows: *given a set of surface points of an object, classify these data points into* meaningful *subsets, each of which is on a single physical part of the object.* Figure 1.1 (a) is an example of a range image which needs to be segmented into parts. Since range data are only a set of discrete data, they do not provide any explicit part information. To segment the object into parts, one must first specify a definition of *meaningful* parts. Then based on this definition one may conceive of a segmentation algorithm. Such algorithms for part segmentation have been traditionally based upon the *geometrical properties* of objects. For example, objects can be defined in terms of part boundaries where object surfaces are sharply concave [**60**]. In general, algorithms employ the surface principal curvature to locate the deep concavity [**49, 47**]. As

[1] This is not defined exactly and must draw upon psychophysics. Definitions of parts are discussed in [**72, 60, 80, 118**].

number of parts	2	
part index	1	2
shape type	ellipsoid	tapered cylinder
translation	$t_x = -0.974$	$t_x = 2.643$
	$t_y = 15.917$	$t_y = -43.104$
	$t_z = 0.090$	$t_z = -0.670$
rotation	$r_x = -89.3127$	$r_x = 90.6738$
	$r_y = -3.57199$	$r_y = 3.49784$
	$r_z = -179.45$	$r_z = -172.721$
size	$a_x = 21.503$	$a_x = 8.719576$
	$a_y = 21.987$	$a_y = 8.675747$
	$a_z = 38.048$	$a_z = 20.69575$
tapering rate	$k_x = 0.0000$	$k_x = 0.3379$
	$k_y = 0.0000$	$k_y = 0.3515$
curvature of axis	0.0000	0.0000

TABLE 1.1. The part-based description of the bowling pin shown in Figure 1.1. Here the shape types of the part models are given in qualitative terms and the rest of the features are described quantitatively. The details of the latter will be explained later. Qualitative object recognition can be performed by matching the number of parts and the shape types of parts to the object models in a database. The other quantitative information is used only if they are needed.

an alternative approach, we use a surface physical property, *the simulated electrical charge density distribution*, to perform 3D object segmentation into parts [**149, 148**]. There exists an analogy between a discontinuity of surface tangents and the singularity of the electrical charge density over the object surface. It is well known that for a charged conductor, the charge is only distributed over the outer surface of the object, tends to accumulate at a sharp convexity, and vanishes at a sharp concavity. *In this thesis we propose that object part boundaries, which are usually denoted by a sharp surface concavity* [**60**], *can be detected by locating surface points which exhibit local charge density minima.* This physical definition allows us to develop an efficient algorithm which does not suffer from some of the traditional problems, such as unrealistic assumptions about surface smoothness, instability of local surface feature computations, and the need to select crucial user-defined parameters.

The problem of part identification can be stated as follows: *Given a set of data points on a particular part and all candidate part models, find a model which is the best description of that part.* This is illustrated in Figure 1.3. A single-part object is shown at the left and

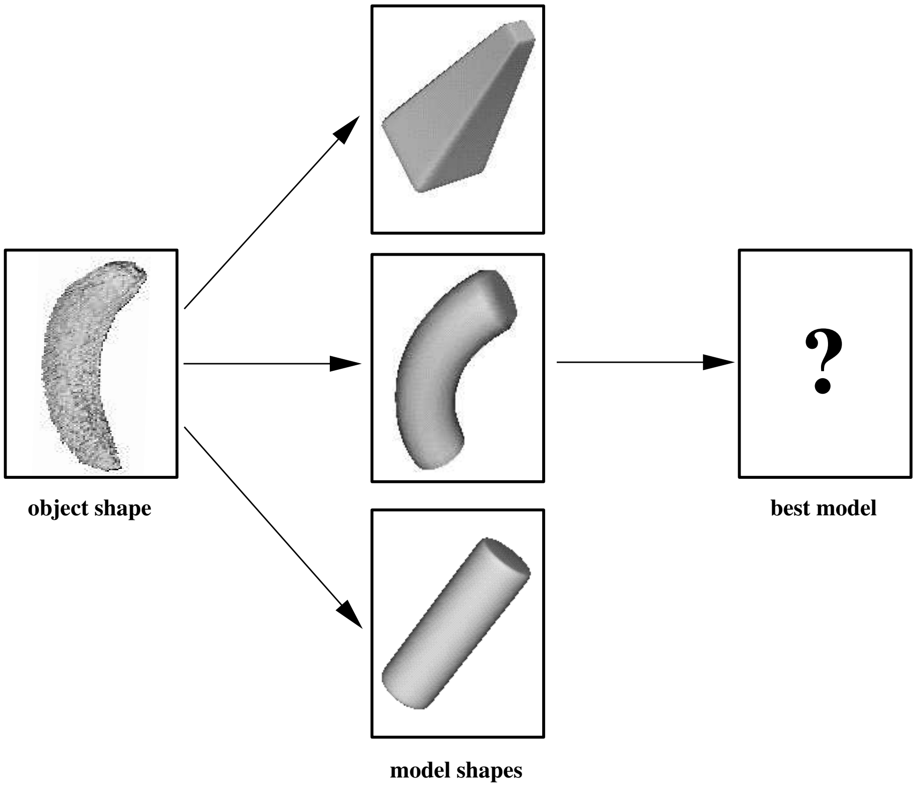


FIGURE 1.3. An example of part identification. A single-part object is shown at the left and three possible part models are presented in the middle. The question to be answered is: Which model is the best description of the object?

three possible part models are presented. The question to be answered is: Which model is the best description of the object? It is important to stress at this juncture that the shapes of object parts are usually not exact instances of the part models. Therefore the model recovery algorithm must be able to *approximate* a part shape by a model shape. However, in all previous work with geons, qualitative part models were recovered without any *explicit shape verification*. They either did not verify the overall shape of the resultant model, or verified it in the model parameter domain but not by shape. In this thesis, we propose a *top-down* strategy to compute the qualitative shape models of parts *robustly and accurately* from data representing parts whose shapes are not fully consistent with their models. We introduce *parametric geons* as object part models in the form of implicit equations of restricted deformed superellipsoids [**8, 123**]. Parametric geons are *seven* qualitative shapes associated with pose and attribute parameters which control the model size, tapering rate, and axis curvature [**145**]. Parametric geons provide explicit global constraints on the *qualitative* shape of part models. This constraint allows the algorithm to directly compare the model shapes with a part shape. In the literature, bottom-up approaches are used [**41, 65, 88, 13, 101**] which do not employ such a global constraint. In these citations, computed geons are matched to local features only. Similarly, in the only previous work which used parametric models [**100**], shape verification is performed by comparing the Euclidean distances in parameter space of superellipsoid parameters [2]. However, the Euclidean distances of model parameters do not exactly measure the shape similarities of superellipsoids. For example, one superellipsoid shape may be described by two different sets of parameters. In contrast to previous work, *we perform qualitative model recovery by comparing all parametric geon models to a part and selecting a model of that part based on the similarity between the part shape and each candidate model* [**146, 147**]. The parametric geons and model recovery strategy provide a mechanism for explicit shape verification, thereby achieving more reliable shape approximation.

The part-based descriptions produced by our approach have wide potential application. In many situations objects can be easily distinguished by their parts. More exact information about objects is only secondary. For an autonomous robot navigating in an office environment, its task is to avoid obstacles and identify possible targets. Obstacle shapes are usually very different from target shapes. Descriptions of objects based on parametric geons are often sufficient for an autonomous agent to perform those tasks. Another example pertains to automated manufacturing where an assembly machine may need to quickly classify and select industrial parts on a conveyor belt. This could be done on the basis of coarse shape descriptions. Parametric geon-based descriptions are very useful for this purpose.

3. Thesis Overview

In Chapter 2, we begin with a review of the previously related research on object segmentation into parts, volumetric primitives and part model recovery. Segmentation can be approached in two different ways: by searching for differences or similarities [**155**]. The technique for searching for differences is *boundary-based*; it locates part boundaries and decomposes the object into parts at these boundaries. The search for similarities is *shape (or region or primitive)-based* and directly find parts based on their shapes. We categorise volumetric primitives as *quantitative* and *qualitative* according to whether the definition of shape relies on a continuous metric or qualitative terms. We also discuss two major model recovery schemes, *bottom-up* and *top-down*, used in the past. The bottom-up approaches

[2] Although a major aspect of this work is to recover the superellipsoid models, the ultimate goal is to derive a qualitative shape model from the superellipsoid parameters. Thus, there is a difference between this work and others dealing with superellipsoid model recovery [**94, 27, 123**].

STAGE	PROCESSING	INFORMATION
data preparation	image acquisition	range data
	multiview integration (optional)	multiview 3D data
object segmentation	surface triangulation	triangular mesh
	charge density computation	charge density distribution
	part decomposition	segmented parts
part identification	model fitting	potential models of parts
	model selection	best models of parts

FIGURE 1.4. A simplified flow diagram for information processing.

infer part models by collecting and grouping local features. In top-down approaches, global shape constraints of part models are used to guide model search.

In Chapter 3, we present the motivation for and methodology of this thesis. We first summarise the RBC theory and its computational hypotheses. We then discuss some difficult issues in previous work on geon recovery, such as, inability of perfect line drawing extraction, nonuniqueness of geon definitions, instability of part segmentation, lack of explicit shape verification and insufficiency of using single-view data. We also describe our alternatives for dealing with these problems. General assumptions about object shapes, part shapes, part models and preprocessing for our approach are proposed and used as constraints for developing algorithms. A general framework in terms of a constrained optimisation is presented to solve the problem dealt with in this thesis. We also make a comparison between our work and previous research in part segmentation, qualitative shape approximation, volumetric models and part model recovery.

Chapters 4 and 5 describe the details of our approach. A flow diagram of the pertinent information processing is given in Figure 1.4. Single-view or multiview range data are obtained by a laser rangefinder. Multiview range data are further registered and integrated into a world coordinate system. Redundant data which are contained in more than one view are removed.

In Chapter 4, we propose a new approach to 3D object segmentation into parts based on a simulated electrical charge density distribution. In order to compute the distribution numerically, we tessellate the object surface using a closed *triangular mesh* and then apply a *finite element method*. A *direct connection graph* based on the spatial relations between triangles in the triangular mesh is then constructed as a convenient coordinate system on the object surface. Employing this graph, we detect triangles on part boundaries where the charge densities achieve local minima. These triangles are deleted from the graph which are then divided into a few subgraphs. In this way, the object is broken into parts. Triangles belonging to the same physical part are easily obtained by a connected component labelling process.

In Chapter 5, we introduce a new approach to part model identification. We define *parametric geons* as object part models and compare them with Biederman's geons. We perform the model recovery using an optimisation procedure and introduce a new objective function for this purpose. We also discuss the characteristics of the objective function and the optimisation technique.

In Chapter 6, we examine the systematic experimentation we performed. Both synthetic and real data obtained from single and multiple viewpoints are used. We describe the data acquisition system and specify all user-defined parameters in our algorithms. For part segmentation, we investigate the characteristics of charge density distributions over 2D contours and surfaces of 3D solid objects. Then we segment objects into parts. For part identification, we obtain parametric geon-based descriptions of multi-part objects. We also examine the efficiency of the objective function for model fitting, the discriminative properties of parametric geons, the effect of object shape imperfection and the salience of multiview data for shape approximation.

In the last chapter, we summarise the thesis, discuss the contributions and limitations, and point out some directions for future work.

4. Claims of Originality

This research explores computational strategies for qualitative shape representation of a 3D object sensed by a laser rangefinder. Our approach computes a coarse object shape description by (1) segmenting an object into parts based on physics, and (2) representing each part by a qualitative shape type and its quantitative size, pose and deformation information. This kind of description supports efficient object recognition which would be useful for an autonomous agent carrying out complex tasks.

The contribution of this thesis is a new paradigm for qualitative volumetric shape-based representation of 3D objects in range images. The major contribution this research makes to knowledge in the field of computer vision is a physics-based approach to object segmentation into parts. It is based on an analysis of the *simulated electrical charge density distribution on the object*. To our knowledge, this is the first time such an analogy has been used to characterise an object's shape and segment it into parts. This approach provides a superior alternative to traditional geometry-based approaches and creates a new direction for object shape representation.

The main contributions of our research are as follows:

- **Shape Characterisation** We propose a novel physical property, *the simulated electrical charge density distribution*, to characterise three-dimensional object shapes.

- **Part Segmentation** We segment an object into parts at the part boundary points, which are characterised by the local minima of the simulated electrical charge densities.

- **Model Recovery** We define the part models, parametric geons, by explicitly specifying the qualitative shapes of part models and recover the models by a *direct shape comparison.*

- **Implementation and Evaluation** We successfully obtain parametric geon-based descriptions of objects. We also examine properties of charge density distributions over 2D contours and 3D surfaces, and investigate the quality of model recovery affected by different objective functions, shape imperfections and amount of input data.

To summarise, this thesis explores computational strategies for obtaining a qualitative shape representation of a 3D object sensed by a laser rangefinder. We propose a new approach to object segmentation into parts and part model recovery. The charge density-based segmentation is a novel computational paradigm for part segmentation and overcomes some of the difficult issues which are characteristic of traditional approaches. Our approach to model recovery performs explicit shape verification by taking advantage of the global shape constraints provided by parametric geons, thereby achieving reliable shape approximation. These will be described in detail throughout this thesis.

CHAPTER 2

Related Work

In this chapter, we will first survey previous research on part-based object representation. We particularly focus on three topics: (1) part segmentation, (2) volumetric part models, and (3) part model recovery. We do not intend to describe in detail the individual contributions on these topics, but prefer to classify different approaches into categories and evaluate the methodology in each one. The related work on charge density computation and surface triangulation is briefly reviewed in Chapter 4. A *comparison* between our work and the previous research discussed in this chapter is made in the next chapter.

Before reviewing the previous work, we specify certain terminology related to the concept of dimension as used in this thesis. Generally speaking, the dimensionality of an entity is referred to as the domain of that entity. Accordingly, 3D data are a collection of discrete samples, $\mathbf{d}_i \in R^3, i = 1, ..., M$, which define certain positions on the object surface. *2D data* are indexed by a pair of variables and lie in a plane. *Range data* are a set of 3D data which are obtained from range images and which can be also indexed by a 2D integer grid of the range image. An object surface is represented by a set of 3D data. A *3D object* is a connected, bounded space enclosed by its surface. A *2D object* is strictly defined in a 2D domain, such as a planar closed contour.

1. Part Segmentation

The problem of 3D object segmentation into parts is to decompose the complete object surface into different *meaningful* regions. A more formal definition of image region segmentation is given as follows [**64, 79**]: Let $\mathcal{X}$ be the sampling lattice, the domain of the image data, $I(i, j)$. A logical predicate $P(\cdot)$ is defined on the subsets S_k of $\mathcal{X}$.

Definition 2.1. *A segmentation of $\mathcal{X}$ is a partition of $\mathcal{X}$ into subsets or regions $S_k, k = 1, .., N$ for some N such that*

1. $\mathcal{X} = \bigcup_{k=1}^{N} S_k$	*every pixel* (i, j) *must be in a region*
2. $S_k \bigcap S_l = 0$	*regions must not overlap for all* $k \neq l$
3. $P(S_k) =$*TRUE*	P *is the evaluation of the region property for each* k
4. $P(S_k \bigcup S_l) =$*FALSE*	*adjacent regions must have different properties for all* $k \neq l$

This definition was originally proposed for the problem of 2D image segmentation. In this case, the predicate $P(\cdot)$ was defined for image intensities. We believe that it can also be applied to other segmentation problems if the domain of the image data and the logic predicate are generalised. For example, if we use an appropriate domain for 3D data and apply $P(\cdot)$ to the object surface, the definition will apply to 3D object segmentation.

Various part segmentation strategies differ fundamentally by how they utilise different predicates $P(\cdot)$. Thus in our discussion of previous research, we will consider the underlying data domains and the evaluation of part properties. Note that we only discuss segmentation of 3D objects into parts. Reviews for 2D objects can be found in [**114, 117**].

1.1. Shape-based Approaches Shape-based approaches [1] decompose objects into parts according to the similarity between the shapes of part models and object parts. Let $\mathcal{S}$ be a set of data points representing the surface of an object. Let $\mathcal{M} = \{M_i, i = 1, ..., N_m\}$ be the set of part models, which are known before segmentation; here N_m is the number of part models. M' is defined as a collection of individual M_i's which, when combined, constitute the complete object. Let Y be a measure of the similarity between M_i and $S_j \subset \mathcal{S}$, where $j = 1, ..., N_s$. Let β be a tolerance threshold associated with the shape similarity measure Y. A logical predicate $P(\cdot)$ is defined on the $\mathcal{S}$ and M':

Definition 2.2. *A* **shape-based segmentation** *of* $\mathcal{S}$ *is a replacement of* $\mathcal{S}$ *with* M'*, such that*

1. for some i, j, $Y(S_j, M_i) < \beta$	*the shape of the partial surface is similar to a part model*
2. for all $k \neq i, Y(S_j, M_k) > \beta$	*this surface is not similar to other part models for all* k
3. for M', $P(M', \mathcal{S}) =$ *TRUE*	*global shape verification*

We assume that M' is the result of segmentation and the spatial relationships between parts are implicitly contained in M'. An important aspect of this kind of approach is that part segmentation and part identification are performed simultaneously. An example is presented in Figure 2.1. Before segmentation, it is assumed that object parts are defined by model shapes in the set $\mathcal{M}$, such as a cylinder, a cuboid and a cone. Since the shape of

[1] These are also referred to in literature as primitive-based or region-based approaches.

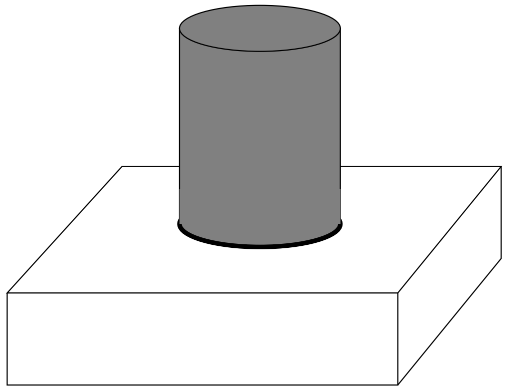


FIGURE 2.1. Shape-based segmentation. The object is segmented into two parts, the cylinder shown shaded and the cuboid shown as a line drawings. The shapes of these parts are consistent with predefined part models.

a subset of the object surface (the shaded portion) is the same as a cylinder, the unshaded portion is the same as a cuboid, and the combination of the two with the appropriate spatial relation is similar to the shape of the entire object, we can say that this object has been segmented into two parts, a cylinder on the top of a cuboid.

Several shape-based approaches have been proposed which use hypothesis-and-verification strategies [**90, 84, 94, 92, 34, 42, 57, 62, 124, 37, 77**]. They first generate a hypothesis of a configuration of an object composed of part models, assuming that the shape of each part is the same as that of a model. Secondly, they evaluate a measure of the similarity between the hypothesis and the real object shape. If the measure is worse than a preselected threshold, another hypothesis is generated and evaluated until the similarity measure is less than the preselected threshold. The last hypothesis is then adopted as the desired object segmentation.

This kind of approach performs object segmentation into parts directly using part shapes as constraints. It is particularly advantageous when part boundaries are locally ambiguous, for example, the "elbow problem" given in [**60**] (see Figure 2.2). In this case, part shape plays a crucial role for decomposition because the results of any segmentation are ultimately verified by the part models [**7**]. Some researchers argue that when image data are incomplete and imperfect, object shape itself is necessary to achieve good segmentation [**109, 102**]. That is, scene segmentation and shape representation are interdependent. However, from a practical point of view, this approach is more complex than boundary-based approaches (see Section 1.2). This is because one must determine not only the locations of parts but also their shapes. In contrast, boundary-based methods first determine the locations of the parts and leave part identification to the following stage. Another problem

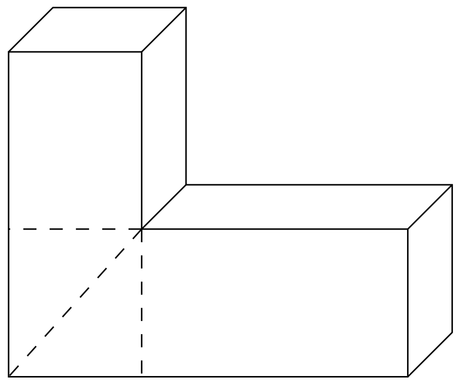


FIGURE 2.2. The elbow problem. On the front flat surface of the object, there is no local indication showing how the surface can be segmented into parts. The dashed lines indicate three possible options.

with these techniques is the possible non-uniqueness of the decomposition. This is because the model shape information only provides the necessary condition for segmentation. For example, in Figure 2.1, it is true that one part is a cylinder. However, two cylinders with the same diameter (one on the top of the other) can also make up the same part. If the model shape was not very similar to the actual part shape, the problem of nonuniqueness would be worse. Additional constraints and further corrections are then required [**62, 124**].

Another type of shape-based segmentation uses an aspect hierarchy of part shapes [**41, 39, 101**]. These methods also commonly employ a finite set of distinctive part models. Each model exhibits a restricted number of configurations of surface patches in all possible views. Thus, for all models, the number of surface configurations in all possible views is also limited. The procedure is to first identify surface patches using region growing [**61**] or edge detection [**30**] and then group surface patches into a potential part according to possible surface configurations. For "clean" image data and perfect object shapes, this approach would be efficient. However, if image data contain noise and object part shapes are not exactly the same as that of the models, errors in surface patch segmentation and actual surface combinations will cause incorrect part segmentation. This kind of approach cannot deal with the elbow problem shown in Figure 2.2 because the front surface which belongs to different parts will be treated by this approach as the surface on one object part.

1.2. Boundary-based Approaches This approach uses a predicate $P(\cdot)$ to find object part boundaries instead of part shapes. The predicate defines the part boundary, permitting us to extract the boundary points and divide the surface data into different sets, each of which corresponds to one part. Let $\mathcal{S}$ be the set of data representing the surface of

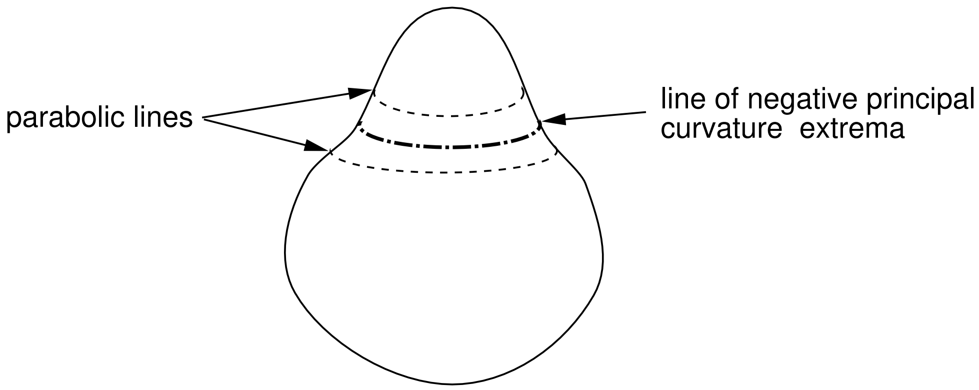


FIGURE 2.3. Parabolic lines and negative curvature extrema. Parabolic lines divide the object into two parts and a connecting region. The negative curvature extrema separate the object into two parts which are consistent with human intuition.

an object and $S_i \subset \mathcal{S}, i = 1, ..., N$ be a part, a subset of the object surface. Let $S_b \subset \mathcal{S}$ be the set of data points on the part boundaries. A predicate $P(\cdot)$ is defined for part boundary points:

DEFINITION 2.3. *A* **boundary-based segmentation** *of $\mathcal{S}$ is a partition of $\mathcal{S}$ into $S_i, i = 1, ..., N$, such that*

1. $S_i \bigcap S_j = 0$	*different subsets of object surface belong to different parts*
2. $S_i \bigcap S_b = 0$	*data points on part boundaries and on parts do not overlap*
3. $\left(\bigcup_{i=1}^{N} \mathcal{S}_i\right) \bigcup S_b = \mathcal{S}$	*every data point must be either on parts or on part boundaries*
4. $P(S_b)$=*TRUE*	*evaluation of boundary points*
5. $P(\mathcal{S} - \bigcup_{i=1,i\neq j}^{N} S_i)$=*FALSE,*	*S_j is any subset of object surface belonging to a part, surface points on parts do not belong to boundaries*

In the literature, there are two theories to define part boundaries. Koenderink and Van Doorn [**72**] have proposed parabolic lines as part boundaries. At the parabolic line, one of the principal curvatures [**89**] of the surface changes from convex to concave. That is, the Gaussian curvature [**89**] at the parabolic line is zero as shown in Figure 2.3. Such parabolic lines possess several attractive properties. For example, parabolic lines do not intersect and are always closed curves [**72**], which outline elliptic surfaces. However, parabolic lines may not always indicate boundaries of actual parts [**105**]. An example is given in Figure 2.4. The parabolic line (the dashed black line) is contained in a single-part object and does not lie between two perceptually intuitive parts. Hoffman and Richards have also shown

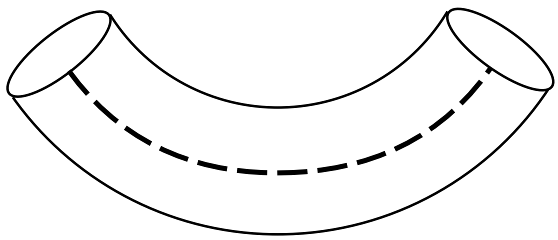


FIGURE 2.4. The parabolic line shown by the black dashed line is not a boundary of two intuitive parts.

that Gaussian curvature cannot indicate part boundaries on cylindrical surfaces [**60**]. Also, since this method is based on the classification of positive and negative Gaussian curvature regions, it is not clear how to apply it to objects containing planar surfaces. Moreover, it sometimes produces results which are not consistent with human visual perception of volumetric parts. For example, in Figure 2.3 the object will be segmented by this approach into two parts, corresponding to two elliptical regions, and a connecting surface corresponding to the hyperbolic region between the two parts. These three surfaces are separated by two dashed lines. The theory does not indicate how to deal with the separating surface. Readers are referred to Rom and Medioni [**105**] who have performed part decomposition based on this theory using range data as input.

Hoffman and Richards [**60**] have proposed another criterion for defining part boundaries. They claimed that the mental concept of a part is based upon a particular regularity in nature – transversality [**56**]. This theory states that when two arbitrarily shaped surfaces are made to interpenetrate, they always meet at a contour of concave discontinuity of their tangent planes, as shown in Figure 2.5. Here two ellipsoids interpenetrate and their surfaces meet at a contour of concave discontinuity of the tangent planes.

In fact, many objects actually contain parts which are joined smoothly as shown in Figure 2.3. This can be viewed as a smoothed transversal intersection where the surface is observed to have greatest negative curvature [2]. Accordingly, the rule of part segmentation has been proposed as follows [**60**]:

DEFINITION 2.4. **Minimum rule.** *Divide a surface into parts at loci of negative minima of each principal curvature along its associated family of lines of curvature.*

Figure 2.3 shows an example of loci of local minima of negative curvature by a dashed-dotted line. According to the minimum rule, this object can be segmented into two parts, separated by this line of negative curvature minima.

[2] In 1989, Bennett and Hoffman proved that near the intersection contour, there are points of arbitrarily large negative curvature on the slightly-smoothed surface [**11**].

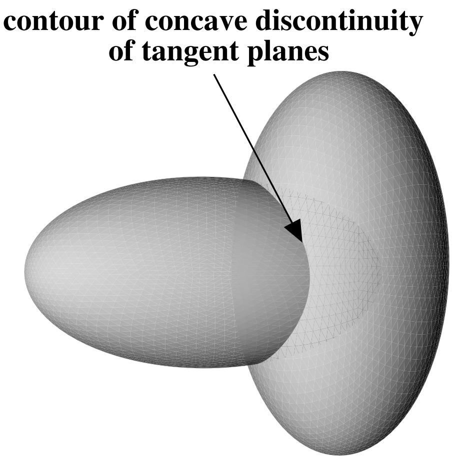


FIGURE 2.5. Transversality. Two ellipsoids joined together create a contour of concave discontinuity at their intersection.

This part definition has been used for segmenting range images [**49, 47, 134**] and 2D [**12**], 3D [**88**] line drawings. The advantage of this approach when compared to those based on shape is that it is independent of the number and shape of the parts. The transversality assumption implies that a part boundary is explicitly demarcated by surface concavities. In other words, a part boundary must be a closed contour of the principal curvature minima. In this way, each individual part is completely separated by the part boundary. Note that if a concavity does not exist where two parts meet, as in the "elbow" problem shown in Figure 2.2, the segmentation algorithm will not be able to find the boundary between these two parts. Therefore it could not separate them. In the case where the part boundary is slightly broken, Lejeune and Ferrie [**76**] have used as putative parts, regions which have positive principal curvature to interpolate the part boundary.

All boundary-based approaches are based on surface curvature which measures the changing rate of the surface tangent plane in a neighbourhood. Its computation involves the first and second partial derivatives of the surface. Thus, an assumption on smoothness of the object surface is mandatory [**14**]. Since curvature computation uses local data and the surface is represented by discrete data, the results are often very sensitive to noise and data sampling errors. A smoothing operation on the range data is usually required [**135**]. Therefore, it would be desirable to develop a new computational approach which does not require smoothing and is not very sensitive to noise and sampling errors. Moreover, all part segmentation algorithms work on partially-viewed objects that can be easily indexed by a

2D image grid. However, if complete shape information has been obtained [**128**], a one-to-one mapping from a 2D grid to surface points is rarely obtained. Thus, a segmentation algorithm which does not rely on a 2D grid is required.

2. Part Models

By part models, we mean volumetric primitives which describe shapes of object parts. The models embody information about the spatial distribution of a shape [**82**] and represent the most intuitive decomposition of an object into parts. The volumetric primitives developed in previous research can be categorised as *qualitative* or *quantitative (parametric)* models.

2.1. Qualitative Models Qualitative models do not rely on a fine metric and provide distinctive shape characteristics which are useful for symbolic object recognition. These qualitative models differ mainly by the types of shapes and the specification of their attributes. Ferrie and Levine [**49**] used ellipsoids and cylinders as coarse descriptions of object parts. The axis lengths, position and orientation of parts were also provided by the part models. Since only two shape types were used for the part models, this approach produced only very coarse object descriptions. Shapiro et al. [**115**] have proposed sticks, plates and blobs as 3D part models. Sticks are long, thin parts that have only one significant dimension. Plates are flattish, wide parts with two nearly flat surfaces connected by a thin edge between them. They have two significant dimensions. Blobs are parts that have all three significant dimensions. A few simple parameters, such as centre of mass, length, area and volume, are also associated with each shape type. These shape models are distinctive perceptually and capture certain salient features of volumetric primitives. This model emphasises volumetric information and contains no intuitive shape information.

Biederman [**15**] has proposed geons as qualitative part models. Geons are thirty-six volumetric component shapes[3], which are described in terms of four qualitative attributes of generalised cylinders [**22**] as shown in Figure 2.6. It is claimed that these properties can be readily detected by an analysis of relatively perfect 2D line drawings. Furthermore, the geons can be differentiated in a 2D image on the basis of perceptual attributes that are largely independent of viewing position and degradation. Psychological experimentation [**18, 19**] and computational frameworks [**13, 41, 85**] have provided support for the descriptive power of such geon-based descriptions.

[3]Biederman later changed the number to twenty-four by merging two asymmetrical attributes [**17**].

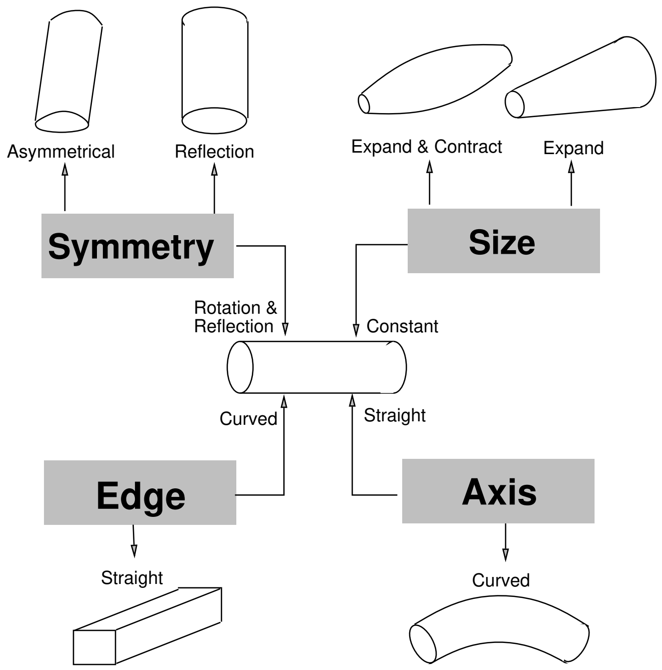


FIGURE 2.6. Biederman's Geons. The geons are defined in terms of three attributes of their cross-sections and one attribute of their axes.

The selection of the actual number of primitives is an open question. In general, the larger the number of primitives a system uses, the greater the descriptive power it has. However, complexity also increases with a larger number. Clearly, there is also a tradeoff between the descriptive and discriminative power of volumetric primitives. There are other systems which use subsets of Biederman's geons. Dickinson et al. [**41**] have defined ten qualitative primitives as part models. They explicitly specified the properties of the volumetric primitives, for example, rectangular or elliptic instead of straight or curved cross-sectional shape. The property of the asymmetrical cross section was not included in their models. Raja and Jain [**101**] have employed 12 primitives by eliminating the asymmetrical property in Biederman's geons.

2.2. Quantitative Models In contrast to qualitative models, quantitative models provide metrics or parameters to control model shapes and attributes on a continuous scale. Binford [**22**] first proposed *generalised cylinders* as object part models. A generalised cylinder is the volume swept out according to a rule by an arbitrary planar shape (the *cross-section*) moving along a 3D curve (the *axis*). The axis, the cross section and the sweeping rule are parameterised individually. This formalism has been accepted as a useful volumetric primitive for a wide variety of shapes [**3, 87, 125, 121**]. However, generalised cylinders are not unique. There exists a large number of descriptions corresponding to one volumetric shape, depending on how the axis and cross sections are selected. Thus the usage of generalised cylinders has been restricted to certain subsets of the model, for example, straight and homogeneous generalised cylinders (SHGC) [**113**]. Accordingly, the shapes of object parts are often assumed to be consistent with the shapes of the more restricted SHGC.

Pentland [**93**] has proposed the use of *superellipsoids*[4], which are a parameterised family of closed surfaces [**51, 8**]. Superellipsoids and their normals are defined parametrically as follows [**8**]:

$$\mathbf{x}(\eta,\omega) = \begin{bmatrix} x(\eta,\omega) \\ y(\eta,\omega) \\ z(\eta,\omega) \end{bmatrix} = \begin{bmatrix} a_1 \cos^{\epsilon_1}\eta \cos^{\epsilon_2}\omega \\ a_2 \cos^{\epsilon_1}\eta \sin^{\epsilon_2}\omega \\ a_3 \sin^{\epsilon_1}\eta \end{bmatrix} \tag{2.1}$$

$$\mathbf{n}(\eta,\omega) = \begin{bmatrix} n_x(\eta,\omega) \\ n_y(\eta,\omega) \\ n_z(\eta,\omega) \end{bmatrix} = \begin{bmatrix} \frac{1}{a_1}\cos^{2-\epsilon_1}\eta \cos^{2-\epsilon_2}\omega \\ \frac{1}{a_2}\cos^{2-\epsilon_1}\eta \sin^{2-\epsilon_2}\omega \\ \frac{1}{a_3}\sin^{2-\epsilon_1}\eta \end{bmatrix} \tag{2.2}$$

$$-\frac{\pi}{2} \le \eta \le \frac{\pi}{2} \quad -\pi \le \omega \le \pi.$$

Here η is a north-south parameter, like latitude, and ω is an east-west parameter, like longitude. ϵ_1 is the "squareness" parameter in the north-south direction; ϵ_2 is the "squareness" parameter in the east-west direction. a_1, a_2, a_3 are scale parameters along the x, y, z axes, respectively. Superellipsoids can be also expressed in the form of an implicit equation as follows [**8**]:

$$\left(\left| \frac{x}{a_1} \right|^{2/\epsilon_2} + \left| \frac{y}{a_2} \right|^{2/\epsilon_2} \right)^{\epsilon_2/\epsilon_1} + \left| \frac{z}{a_3} \right|^{2/\epsilon_1} = 1 \tag{2.3}$$

The advantage of adopting the superellipsoid model is that by using only two more parameters than ellipsoids, it can describe a large variety of volumetric shapes.

Hyperquadrics [**74**] and *fourth order polynomials* [**70**] employ parametric equations and can also be used to describe a large number of volumetric shapes. However, the parameters obtained are not intuitively related to the object shapes. The number of degrees of freedom associated with these two models weakens their uniqueness in describing individual object classes.

[4] Superellipsoids are a subset of the class of superquadrics [8] and are also referred to as superquadrics in the literature.

Terzopoulos et al. [**132**] have proposed the *symmetry-seeking deformable model* which is constructed from generalised splines. This physics-based model is *active* in the sense that the model continuously reacts to external forces produced by the image data. Model recovery is performed by applying forces to models in space so that the shape of its projection onto the image plane is consistent with an object of interest. This model is a powerful means of describing the fine details of irregular objects. However, the solution depends on the initial estimation and does not provide unique information about the volumetric shape. Following this work, Metaxas and Terzopoulos [**133**] have further developed a deformable superquadric model which provides both global and local deformation information. Also, the resulting models are not unique with respect to object shapes since different combinations of local and global deformations can describe the same shape.

Pentland [**96**] has proposed an alternative physically-based model inspired by *modal analysis*. The modal representation yields mode parameters which do possess intuitive interpretations of object shapes. However, without the higher modes and special care taken with respect to the correspondence between data points and nodes, it is difficult to represent objects with surfaces of high curvature.

No previous work has used part models that provide both qualitative symbolic shape types and a quantitative formulation for each shape type. This kind of model can take advantage of the merits of both qualitative and quantitative models. The parametric geons proposed in this thesis have this property.

3. Model Recovery

As discussed in Section 1.1, part segmentation and model recovery may be accomplished simultaneously if the model shape is consistent with the part shape. This type of approach has already been reviewed in that section. Here we only consider approaches to model recovery in which part segmentation has been previously done. More precisely, the problem can be stated as:

PROBLEM 2.1. *Given a set of image data of an object part,* $\mathcal{D} = \{\mathbf{d}_i, i = 1, ..., N\} \subset \mathcal{R}^3$*, a set of part models,* $\mathcal{M} = \{M_j, j = 1, ..., N_m\}$ *and a difference measure* Y *between* $\mathcal{D}$ *and* M_j*,* **part model recovery** *is the problem of finding a particular model* $M_j \in \mathcal{M}$ *that minimises* $Y(\mathcal{D}, M_j)$*.*

Almost all techniques for part model recovery can be categorised as either bottom-up or top-down approaches. Bottom-up approaches take advantage of the small set of local shape features of *ideal* models. They infer a model for a part by collecting and grouping

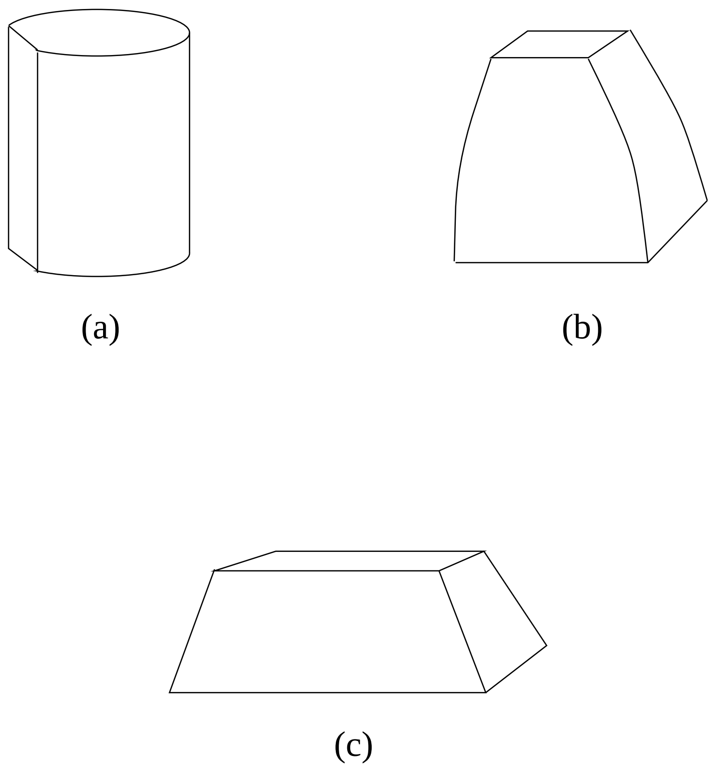


FIGURE 2.7. Some imperfect geon-like parts.

local evidence. These features can be shapes of line segments [**13, 88**] and regions [**41**], curvature of surface patches [**101**], and the relationships between line segments and surface patches. However, when there are variations in the part shapes, these approaches will fall short. Figure 2.7 shows some examples of imperfect geon-like parts. Geon models cannot be uniquely determined from these part shapes. In (a) the cross section boundary contains both straight and curved portions; in (b) the axis is actually straight but a curved axis would be inferred based on information obtained from the object silhouette; in (c) the part can have either a constant or non-constant cross section depending on how the axis is selected. Since none of these shapes satisfies an exact geon definition, it is very difficult for a bottom-up approach to determine an appropriate geon label.

In top-down schemes, part models are derived by fitting models to all data points [**123, 153, 70, 74, 96, 133, 100**]. In this case, parametric models of parts are explicitly defined before model recovery. These models provide global shape constraints for model recovery. An optimisation problem is formulated to minimise the difference between a part model and part data. The advantage of such an approach is that it does not critically rely on local support. Also, the imposed global shape constraint helps to reduce the influence of missing data, image noise and minor variations in object shape. In this way, approximate shape descriptions of objects can be obtained efficiently, thereby bypassing some of the common error-prone processing steps such as building point-by-point descriptions of lines and surfaces.

An appropriate objective function is important for optimisation procedures. Solina and Bajcsy defined an objective function based on the relative position of a data point respect to the inside or outside of the model surface [**123**]. Gross and Boult proposed an objective function based on the mean distance between each data point and the corresponding point on the surface of the model along a line connecting the data point with the centre of the model [**55**]. After carefully studying four kinds of objective functions, they pointed out that their objective function has significant advantages over the others in terms of convergence and accuracy. Yokoya et al. have introduced a two-term objective function [**153**]. The first term is the sum of the squared distance between the data point and the model surface. The second term is a measure of the squared difference between the normals of the object surface and the model surface. The solution of the resulting nonlinear optimisation problem is stabilised by introducing the second term. This is because the surface normal is invariant to scale.

Due to the complex shapes of object parts, the objective function for model fitting usually contains many local minima. A straightforward gradient decent method for function optimisation will often fail to converge to the global minimum. To counteract this, some researchers have used a nonlinear least squares minimisation (Levenberg-Marquardt) method, adding random walks to escape local minima [**55, 123, 131**]. In some cases, where the properties of the objective function are known or an initial parameter estimation close to the global minimum in a nearly convex region can be obtained, this approach will usually take much less time than general global optimisation methods. However, with an inappropriate initial guess, the algorithm may get stuck at a local minimum. This is because global convergence cannot be assured. Yokoya et al. [**153**] employed a global optimisation technique, simulated annealing, whose global convergence has been proven theoretically [**52**]. However it requires a long computational time to find the optimal solution.

Geons can be also derived from quantitative models. Raja and Jain [**100**] have explored the recovery of 12 geons from single-view range images by classifying the actual parameters of globally-deformed superellipsoids. They obtained good results using synthetic data and real objects with smooth surfaces. However, they also found that the estimated parameters were extremely sensitive to viewpoint, noise and objects with coarse surfaces. One of the reasons for their poor results in these latter circumstances is that superellipsoids are nonunique and cause uncertainties in the estimated model parameters, especially when representing noisy and partially-viewed data [**140**]. Certain parameters in globally-deformed superellipsoids tend to interact with each other in ways that make the model difficult to control.

An interesting addendum to superellipsoid parameter classification is provided by Arbel et al. They have proposed a method for *recognising* typical superellipsoid shapes from multiview range images [**4**]. A conditional probability density function is derived by combining model information with *a priori* context-dependent information, the parameter estimated for the unknown object, and the uncertainties of the parameters. They showed that recognition performance is nearly perfect when complete object surface information is available to the algorithm, and that it falls off when only partial information is known. Since they used objects that were regular superellipsoids, they did not deal with tapered and curved shapes and also did not investigate the issue of qualitative shape approximation of imperfect objects.

4. Chapter Summary

This chapter has reviewed previous work on part-based 3D object representation. We have concentrated on three subjects: (1) object segmentation, (2) part models and (3) model recovery. Object segmentation has been accomplished using both shape- and boundary-based approaches. We have seen that all boundary-based segmentation approaches have used a particular geometrical feature, the surface curvature. However, surface curvature computation is known to be extremely unstable. Part models are categorised as being qualitative and quantitative. Qualitative models are defined in terms of a finite number of distinctive shape types. Quantitative models are defined in terms of numerical metrics. A strategy for part recovery usually depends on how the part model is defined. Most approaches for recovering qualitative models follow a bottom-up strategy, which have shown disadvantages in handling imperfect shapes. The techniques for quantitative shape recovery have some useful properties. A computation of qualitative shape primitives from quantitative models has indeed been implemented by classifying superellipsoid parameters. However, the approach did not work well for noisy and coarse-surfaced objects. In the next chapter, we describe the motivation and methodology for our research, and in the following two chapters, present our techniques in detail.

CHAPTER 3

Motivation and Methodology

We begin this chapter with a summary of the Recognition-By-Components (RBC) theory. Then we discuss certain limitations of this theory, as well as previous computational implementations, and describe the methodology proposed in this thesis. These issues include the difficulty of line drawing extraction, ambiguous definition of geon shapes, sensitivity of model recovery to the amount of input data and robustness of part segmentation. We also elaborate the general assumptions made in our research and present the general framework of this thesis. Finally we make comparisons of our research with previous work on part-based, especially geon-based, representations of 3D objects.

1. Overview of RBC Theory

Starting with range data, our objective is to compute part-based descriptions of 3D objects in terms of a finite number of volumetric primitives. This is motivated by a theory of human image understanding, Recognition-by-Components [**15**]. Inspired by speech perception, a process mediated by the identification of individual elements, phonemes, from a relatively small set of primitives, this theory is meant to account for what can be called *primal access*: the first contact of perceptual input from an isolated, unanticipated object to a representation in memory. With support from psychophysical experimentation, RBC posits that:

(i) Objects can be efficiently represented by a set of natural components, the parts;

(ii) Detection of these components from 2D object line drawings is relatively invariant over viewpoints, object size and line drawing degradations;

(iii) If the components in their specified arrangement can be readily identified, object recognition will be fast and accurate.

In addition, this theory hypothesises that:

(i) Parts are segmented at points of deep surface concavities;

(ii) Thirty-six geons are derived qualitatively using four attributes of generalised cylinders. The features can be detected from 2D line drawings based on five non-accidental and detectable properties of edges in images [**81**];

(iii) The geon description includes shape labels, relative size and aspect ratio of each geon, as well as the spatial relationships between geons.

RBC's central contribution is its proposal for a particular vocabulary of components derived from perceptual mechanisms and how it accounts for rapid object recognition using an arrangement of these components. Since RBC was introduced to the computer vision community, it has motivated considerable research [**13, 68, 41, 38, 44, 65, 88, 101, 116, 139, 145**]. However, it is realized that there exist certain limitations on both the theory [**75**] and its computational implementation [**36**]. In the next few sections, we discuss some of these issues and describe the methodology adopted in this thesis.

2. From Line Drawings to Range Data

RBC assumes that the line drawings of an object can be obtained by applying an edge detection technique to the intensity image of the object. However, line drawings and edges are not exactly the same in computational vision. Edge detection extracts curves in the image where rapid changes occur in intensity [**63**]. The detected edges may be due to the changes in surface orientation and reflectance properties, colour, texture, object occlusions, shadows and noise. On the other hand, line drawings convey information about object surface discontinuities in orientation and depth. In practice, "clean" line drawings are rarely obtained from edge maps because of insufficient constraints for extracting line drawings. Only with carefully selected objects and well controlled lighting conditions can perfect line drawings be produced. This is equivalent to imposing a constraint on the image acquisition process for the extraction of the desired properties of the objects. This can be readily accomplished in machine vision by using an alternative sensing device, such as a laser rangefinder [**69**]. With such a range sensor, the explicit 3D information about an object surface can be directly obtained while other effects due to colour, texture and shadows are inhibited. An edge-junction graph containing the surface discontinuity information can be extracted relatively easily from range images [**54**] and geon-based descriptions can be built from the graph [**88**]. From a practical point of view, raw range data provide 3D surface information and can be directly used for object segmentation and model recovery. Since geons are volumetric shape models derived from generalised cylinders [**22**], it should not matter whether the input data are from intensity or range images. Hence, our approach

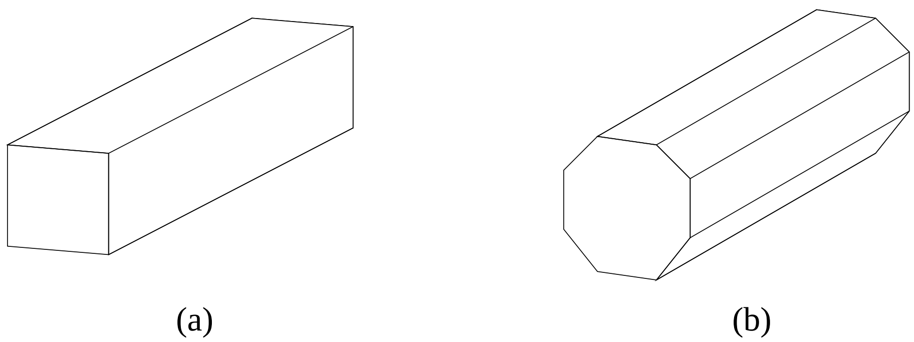


FIGURE 3.1. Ambiguous geon shapes. Two visually different shapes in (a) and (b) have the same geon definition.

employs a laser rangefinder and uses surface information to derive part-based descriptions of objects. Other researchers have also used range data for geon recovery [**101, 88, 100, 39**]. However, their techniques differ from ours (see Section 9).

3. Unambiguous Definition

There are shapes that are quite different in appearance but which have the same shape type in terms of the RBC's geon definitions. Figure 3.1 shows two tubular shapes having square and hexagonal cross sections, respectively. Although they are perceptually different in shape, they have the same geon definition -namely straight axes, straight cross section edges, symmetrical cross sections and constant cross section sizes. This is caused by the ambiguity of the geon definitions in the RBC theory. Because the geon definitions are important constraints for designing the geon recovery systems in some cases [**13, 88, 101**], this ambiguity makes the design of such systems extremely difficult. In practice, all computational implementations have regarded the shape in Figure 3.1 (a) as the default shape of this type of geon. In addition, geons are a subset of generalised cylinders [**22**], whose description is usually nonunique (see discussion in Chapter 2). Thus, strictly speaking, the geon definition of Biederman is insufficient for developing geon recovery algorithms and more constraints are required. Our approach is to make explicit the specifications for the qualitative shapes of models, thereby ensuring unambiguity in model recovery. We note that the qualitative part models used by Dickinson et al. [**41**] were defined in terms of explicit specifications of the *attributes* of volumetric shapes.

4. Shape Approximation

Nearly all of the work inspired by RBC has focused on the recovery of geon models from objects consisting of *perfect* geon-like parts. In these cases, the input was the edge map

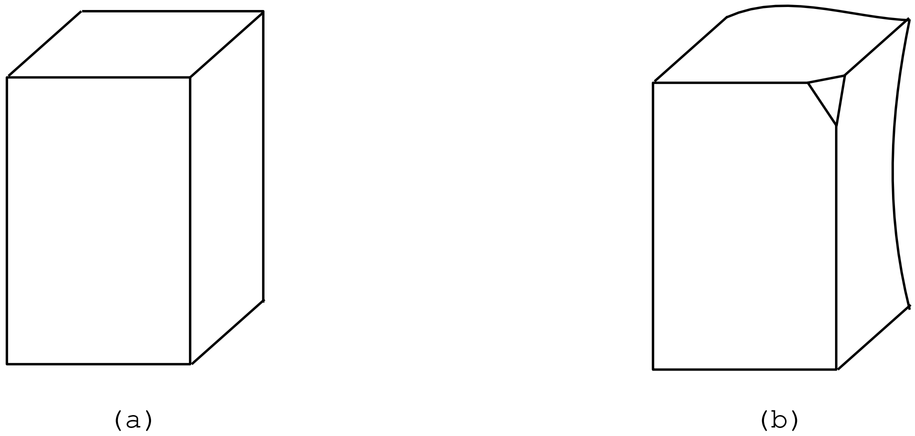


FIGURE 3.2. Perfect and imperfect geon shapes. (a) A block whose shape is a perfect geon. (b) A block with minor shape virations depicting an imperfect geon-like shape. Note that all geon properties of the block in (a) do not exist for this shape.

or line drawings. The geon shapes were defined in terms of attributes. The computation of these attributes was constrained by local evidence. As such, part descriptions were determined in a bottom-up fashion, inferring global properties by aggregating local features. This type of approach cannot succeed when object features do not fully satisfy the exact definitions of the geons. This is illustrated in Figure 3.2. The shape of a perfect block conforms well to the definition of the geon, which is specified by a symmetrical cross-section, straight cross-sectional edges, a constant cross-sectional size and a straight axis. If minor shape variations are introduced to the block as shown in Figure 3.2 (b), all of the geon properties the block possessed do not exist for this shape. It is clear that object shapes in the real world vary in many ways and that image data are often contaminated by noise. However, the number of geons is finite and can never completely depict all possible shape variations of parts. Thus, the process of finding geons for imperfect data of a non-ideally shaped object must focus on *shape approximation*, that is, deriving a simplified and compressed description in terms of perfect models. According to RBC [**15**], "the memory of a slight(sic) irregular form would be coded as the closest regularised neighbour of that form". In order to accomplish this, we define our models in terms of implicit functions, fit these models to data of objects having imperfect geon-like parts, and then select the model for the data based on the minimum fitting residual. In this way, we actually impose global shape constraints during the model recovery procedure so that the result must be one of the predefined shapes. In this way, the purpose of the shape approximation of imperfect geon-like parts by perfect models is achieved.

5. Amount of Information

RBC hypothesises that any geon is rapidly identifiable and invariant over viewpoints. This assumption is reasonable in many situations where a fair amount of shape information about the object and certain constraints on object shape are available. However, for general single-view data, complete shape information of a part will never be available due to self-occlusion. Palmer et al. [**91**] studied the perceptibility of various objects presented in many views. They showed that their subjects clearly preferred a three-quarters frontal view over all others for recognition. This study revealed that the amount of information present for recognition varies in different views. RBC also concurs with this view that objects can be more readily identified from certain orientations than others [**15**].

A few research results have pointed out the effect of the amount of data on sensitivity when fitting quantitative models. Boult and Gross [**27**] concluded that single-view range data may not be sufficient for reconstruction of superellipsoids without additional assumptions and multiview data yield much better results. In the study of the discriminative ability of superellipsoid parameters, Raja and Jain found that estimates of model parameters are highly dependent on the viewpoint [**101**]. More extensive studies carried out by Whaite and Ferrie have demonstrated the nonuniqueness of fitting superellipsoids to single-view range data [**140**]. They have shown, by an example, that the lack of a unique fit can be attributed to the extra degree of freedom allowed by a superellipsoid shape parameter. Thus many models can be fit to data resulting in almost the same fitting residuals. Since there is generally no *a priori* knowledge of which view should be taken, the authors suggested two basic alternatives for dealing with the ambiguity of interpretation: (1) impose further constraints on the superellipsoid models or (2) seek additional data by minimising some measurement of uncertainty in the model. In accordance with their suggestions, we have postulated the parametric geon model by restricting the superellipsoid shape parameters. We also investigate the uniqueness of parametric geon recovery with both single-view and multiview data when the shape of object parts is not fully consistent with any parametric geon.

6. Part Segmentation

RBC proposes that objects be segmented into parts at deep surface concavities. This boundary-based segmentation scheme was originally proposed by Hoffman and Richards [**60**]. The reason for this approach is that it conforms well with human intuition about parts, exploits a property of nature - transversality, and does not require *a priori* knowledge of part shapes - even in the case of a nonsense object [**15**]. A common tactic in this kind of segmentation is to compute surface features which contrast boundary and non-boundary points and decompose the object into parts at boundary points. The key issue here is how to *reliably* locate the part boundaries. To our knowledge, the only feature used in *all* previous boundary-based segmentation approaches [**49, 47, 76, 105**] is surface curvature, which is defined in terms of differential geometry [**89**]. However, surface curvature computation has proven to be unreliable [**134**]. Contrasted with this strategy, we present a new paradigm for part segmentation which employs a simulated electrical charge density as a surface feature. We regard an object as a physics-based model rather than a geometrical one and compute a physical feature which readily indicates the part boundary points. The advantage of this approach is that the computation of the charge density does not require an assumption on surface smoothness and is robust to noise. Therefore, object part boundaries can be reliably located.

7. General Assumptions

Several assumptions are made in this thesis. These provide the constraints necessary for developing the algorithms and the basis for comparison with other related work.

Objects: We use isolated objects composed of either one or multiple parts. All objects are simply connected. Object surfaces need not be smooth. For multi-part objects, parts must be completely separated by relatively sharp surface concavities which delineate part boundaries. Each part boundary must be a closed 3D curve on the object surface.

Part Models: The set of part models consists of seven volumetric primitives whose shapes are distinctive and are explicitly represented by restricted deformed superellipsoids.

Part Shapes: The shapes of object parts and single-part objects may vary from perfect geon shapes. These variations should be moderate and a human should definitely be capable of categorising their qualitative shape.

Input Data: The input data are obtained by a laser rangefinder which scans objects supported by a turntable. Multiview (four views) data and single-view data are collected. No smoothing operation is applied to the range data. Segmentation of the object from its background should be simple. A triangular mesh which tessellates the object surface must be readily computed from both multiview and single-view data for charge density computations.

Evaluation: The performance of the algorithms is evaluated through systematic experimentation on both synthetic data and real data. Comparisons with both theoretical results and subjective criteria are made.

8. General Framework

In this section, we provide a formal description of the task performed in this thesis. Let $\mathbf{x} \subset R^3$ be a set of input data of a multi-part object. The index $i = 1, ..., N_t$ represents the shape types associated with a part model, where $N_t = 7$ is the total number of shape types. Let $\alpha_i \in \{0, 1, ...\}$ be the number of recovered part models with shape type i, where $j(i)$ is the index of α_i. Thus, the total number N_p of parts or recovered models for the object is

$$N_p = \sum_{i=1}^{N_t} \alpha_i.$$

Let g_i be the measurement of the difference between $\mathbf{x}_{j(i)} \subset \mathbf{x}$, a subset of input data, and $\beta_{j(i)}$, the $j(i)$th part model. $\mathbf{x}_b \subset \mathbf{x}$ is a set of points on the associated part boundaries. A predicate $P(\cdot)$ is defined on part boundary points. The task of parametric geon-based representation can be stated as:

Problem 3.1. *Given* $\mathbf{x}$, g_i, N_t *and* $P(\cdot)$, *find a particular set of* $\mathbf{x}_b, \mathbf{x}_{j(i)}$, α_i *and* $\beta_{j(i)}$ *which minimises a function*

$$F(\mathbf{x}_{j(i)}, \alpha_i, \beta_{j(i)}) = \sum_{i=1}^{N_t} \sum_{j(i)=1}^{\alpha_i} g_i(\mathbf{x}_{j(i)}, \beta_{j(i)}), \tag{3.1}$$

subject to:

$$\begin{array}{rlll}
(i) & \left(\bigcup_{i=1}^{N_t} \bigcup_{j(i)=1}^{\alpha_i} \mathbf{x}_{j(i)}\right) \bigcup \mathbf{x}_b = \mathbf{x}, & & \\
(ii) & \mathbf{x}_{j(i)} \cap \mathbf{x}_{n(m)} = 0, & \forall: & i \neq m \quad or \quad j(i) \neq n(m), \\
(iii) & \mathbf{x}_{j(i)} \cap \mathbf{x}_b = 0, & \forall: & j(i), \\
(iv) & P(\mathbf{y}_b) = TRUE, & \forall: & \mathbf{y}_b \in \mathbf{x}_b, \\
(v) & P(\mathbf{y}_p) = FALSE, & \forall: & \mathbf{y}_p \in \mathbf{x}_b.
\end{array}$$

The constraint (i) ensures that input data must be either belong to a part region or a part boundary. The constraint (ii) says that part regions must not overlap. The constraint (iii) means that there is no overlap between part boundary regions and part regions. Constraints (iv) and (v) are evaluations for data points on boundaries and parts, respectively. The function given in Equation (3.1) measures the difference between the object model, a combination of all part models, and the object data, in which a particular data set $\mathbf{x}_{j(i)}$ for a part model $\beta_{j(i)}$ satisfies the constraints from (i) to (v).

By solving the above optimisation problem, we can obtain the total number of parts, N_p, the shape types i for each part, $j(i)$, and their model parameters, $\beta_{j(i)}$. In the actual implementation, we split the procedure into two consecutive stages: (1) object segmentation (solving for $\mathbf{x}_b, \mathbf{x}_{j(i)}$ and N_P) and (2) part identification (solving for $\beta_{j(i)}$).

9. Comparison with Previous Work

This thesis differs from previous research in the following ways.

9.1. Physical Model Our part segmentation strategy is consistent with boundary-based approaches [**60**]. However, unlike previous work, we employ a *physical* property, the simulated charge density distribution over an object surface, to find part boundaries. Other work on part segmentation [**49, 43, 76, 105**] uses *geometrical* properties.

Our approach has some distinguishing characteristics and advantages. Briefly speaking, object segmentation into parts is a partitioning of the object surface based on surface features. Surface feature detection has been traditionally dominated by curvature-based approaches [**14**] because curvature directly reveals geometrical properties of surfaces. Since the curvature computation is a differential operation, surface smoothness must be assumed. This limits the power of the curvature-based approaches. However, a charge density computation, which is based on integration, is applicable in general cases where smoothness of the object surface is not required.

Another advantageous aspect of the charge density computation is its preference for characterising surface properties of complete 3D objects. Curvature estimation for an object surface embedded in a 3D Euclidean space requires a voxel-based coordinate system. Since the latter involves a large amount of memory, a coarse resolution and an integer grid are usually preferred. Because of the inherent noise in images and the quantisation of the coordinate grid, curvature computations based on differentiation have proven unreliable [**135**]. They often need to be corrected by sophisticated analysis [**110**]. Alternatively, a larger area or scale, may be employed to reduce noise effects. However, selecting a suitable scale is in general a difficult problem. Furthermore, a larger scale will increase the computational time. In contrast, our approach uses an integral equation rather than performing surface curvature computations. Since the charge density computation uses all data and these are weighted by a distance factor, the influence of noise is reduced. Scale is not an issue in this regard. The only scale that must be selected is the one associated with the size of the triangles when performing triangular mesh construction. The influence of this scale on the result is much less than that of scale in the curvature computation. This is because the charge density computation is based on an integration over the global surface while the curvature computation is based on differentiating local discrete data. Moreover, if larger scales are chosen for surface triangulation, less computational time is required for the charge density computation. This is consistent with human perception. That is, a human can perceive the coarse structure of an object much more rapidly than its fine structure.

It is important to note that the charge density distribution is not equivalent to the surface curvature. Local minima of the charge density distribution provide only an approximation to the local minima of principal curvature for the object shapes. The charge density distribution can only provide a measure of the contrast between surface concavity and convexity while curvature actually determines absolute geometrical information.

An alternate strategy for obtaining surface features is isotropic diffusion [**151**]. Working in 3D Euclidean space, this approach simulates the propagation of a specific number of particles among object voxels. At a certain intermediate stage of the diffusion process, particle accumulations at sharp surface concavities and convexities become significant. Thus diffusion must be stopped before reaching the equilibrium state. This stopping time is very crucial and difficult to determine. If the diffusion process reaches equilibrium, the particle density is uniform everywhere within the object and therefore cannot indicate any specific property of the object surface. Another difficulty is that the stopping condition varies for different object shapes. We note that, if diffusion is stopped at an inappropriate time, the distinction between particle densities at concave and convex surface points will not be strong. In contrast, our approach evaluates the electrical equilibrium, where charge densities at concave and convex surface points are very distinct. In addition, the diffusion-based approach, which essentially solves a partial differential equation in 3D Euclidean space, must work in a voxel-based coordinate frame. However, we consider only the surface of the object and do not need to perform computations within its interior, as shown in Figure 3.3. This produces a significant reduction in dimensionality and requires the manipulation of many fewer unknowns.

Two papers [**1, 2**] on 2D shape analysis which intended to use the electrical potential have come to our attention. Unfortunately, they arbitrarily specified an equipotential line as the initial condition of their algorithm, either on the image border which is far from the conductor [**1**], or on the contour of a uniformly charged 2D object, which cannot be in equilibrium [**2**]. However, in general the potential difference is dependent on the charge density distribution. Only in the particular case of electrostatic equilibrium can one state

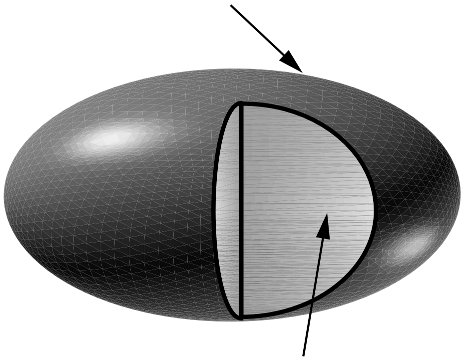


FIGURE 3.3. Comparison between the particle diffusion space (the outer surface and the interior) and the charge distribution space (the outer surface) for a solid ellipsoid(section removed for illustration purposes only).

that the 2D object contour is an equipotential line without knowing the charge density distribution. By failing to determine the equipotential line based on the actual charge density distribution, the resulting algorithms employed a nonphysical initial condition. Thus, their method was physically incorrect. Nevertheless the results were quite reasonable but very slow to compute. This is because the mechanism underlying their algorithm is similar to the 2D diffusion-based approach [**120**].

9.2. Qualitative Shape Approximation Another interesting aspect of our system is the ability to achieve *qualitative* shape approximation. This is a prerequisite for efficient symbolic object recognition. This ability is a direct result of using global shape constraints for models, an idea borrowed from the quantitative approaches. Such constraints are defined in terms of implicit functions and restrict the models to a particular shape family no matter how the input data vary. This significantly assists the process of shape approximation. Most previous work using qualitative models cannot tolerate shape variations outside their model classes [**13, 41, 88, 65**]. This is because they have created their part descriptions in a bottom-up fashion, inferring global properties by aggregating local features. This type of methodology is not robust when object features do not fully satisfy the original definition of the geon features. It is a fact that geons are simple and regular volumes, but objects in the world actually appear in a variety of shapes. Clearly, any computer vision system which successfully recovers qualitative descriptions must address the problem of shape approximation.

We note that the *only* previously reported attempt to perform qualitative shape approximation is due to Raja and Jain [**100**]. They explored the recovery of 12 geons from single-view range images by classifying the actual parameters of globally-deformed superellipsoids, based on the distance in Euclidean space. Although they obtained 89% accuracy for smooth-surfaced objects, they found that estimates may be very poor for parts with noise or "rough" surfaces. They also noticed some "strange effects" in that their major classification errors were due to misclassification of straight and curved geon cross-sections [**101**]. We believe that this drawback is mainly caused by the Euclidean distance measure they used for classifying part shapes. This can be easily illustrated in the 2D case where the geon cross-section is a planar curve. Figure 3.4 shows a series of superellipses. The shape parameter changes uniformly from 0.1 to 1. Accordingly, the shape changes gradually (row by row) from a square to a circle. The number under each figure indicates the value of the associated shape parameter. If these shapes were to be classified into two groups based on the Euclidean distance of the shape parameter, the top three rows would be classified into one group and the rest into the other. However, we clearly observe that the shapes in at least the first four rows are more similar to the square than the circle. Thus, if the Euclidean classification is applied to squared shapes like those in the fourth row, misclassification will occur. This example indicates that there is a significant difference in shape discrimination between an Euclidean distance-based method and human perception.

Another reason is the ambiguity between superellipsoid shapes and their associated parameters, as noted by Solina and Bajcy [**123**]. For example, different parameter sets can correspond to the identical superellipsoid shapes. Therefore, extreme care must be taken when using superellipsoid parameters for qualitative shape identification. In contrast, with our method, we directly compare the difference between the shapes of the models and the object part, and select the part model whose shape is most similar to the object. Therefore, our approach obtains shape approximations which are closer to the human intuition.

9.3. Parametric Geon Models We use a new set of volumetric primitives which have qualitative shapes and are quantitatively deformable in size, tapering rate and bending curvature. This set of parametric geons consists of a finite number of different shape types, but their deformation property makes their shapes appear in many forms. Previous work has been restricted to either purely quantitative models - such as superellipsoids [**123**], fourth-order polynomials [**70**] or hyperquadrics [**74**], or purely qualitative models, such as

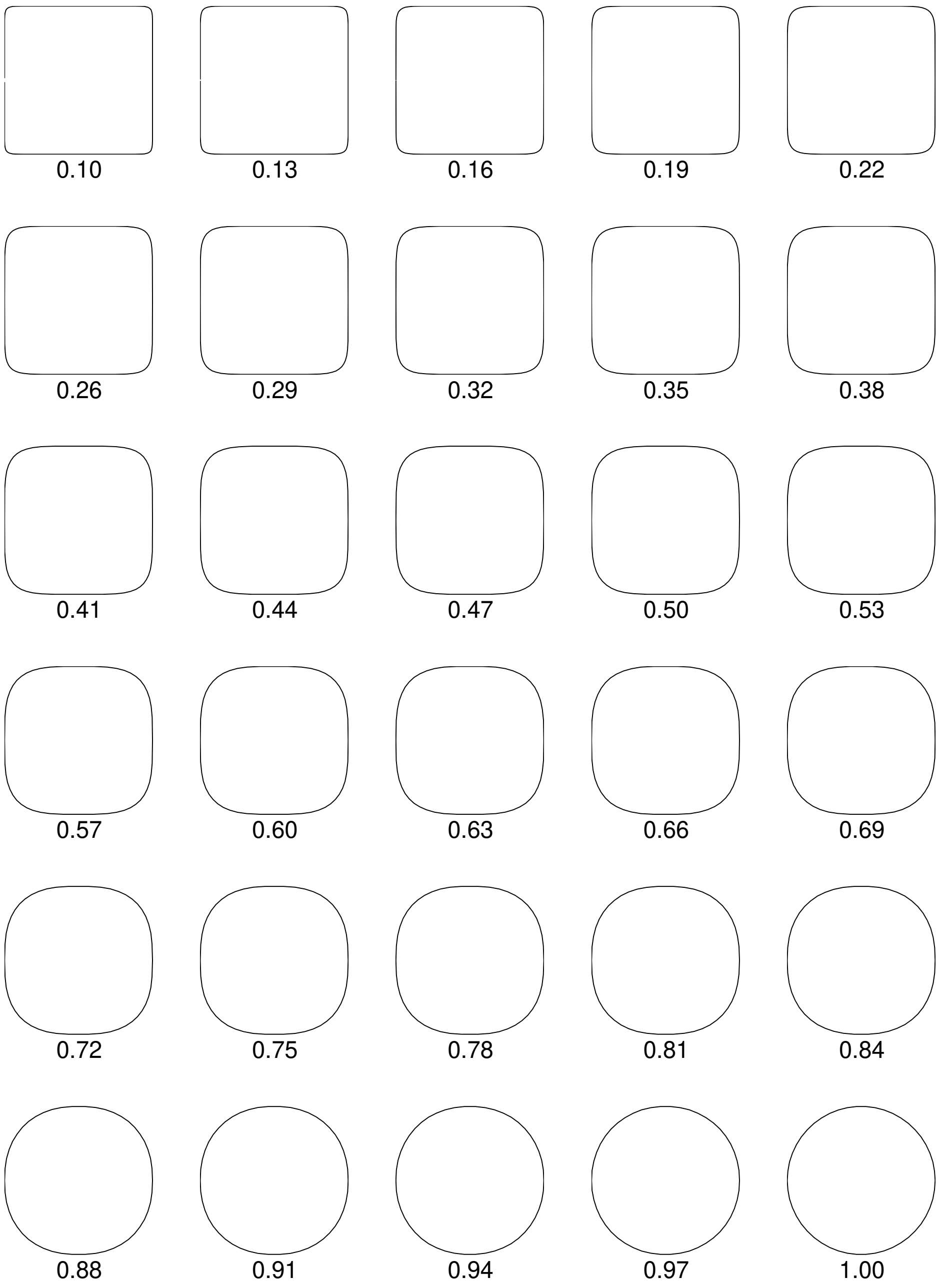


FIGURE 3.4. Classification of cross sections of objects. A series of shapes of a superellipse is given row by row. The shape parameter of the superellipse changes from 0.1 to 1 and consequently its shape changes from a square to a circle. The number under each figure indicates the value of the shape parameter. The task is to classify these shapes into two group, square-like shapes and circle-like shapes. If the classification were based upon the shape parameter, the shapes in the first three rows would be classified into square-like shapes. However, human perception seems to classify more shapes into square-like shapes.

geons [1]. Parametric geons are defined in terms of implicit equations of globally deformed superellipsoids. This formulation yields a global constraint which facilitates model recovery from imperfect geon-like object parts. Shape constraints have also been provided by other quantitative models such as, superellipsoids, hyperquadrics and fourth-order polynomials. However, these constraints do not directly result in qualitative shapes. In order to convert them into symbolic descriptions, further classification is needed.

9.4. Part Model Recovery Strategy We use a new strategy to recover volumetric primitives. A fitting scheme is used to minimise an objective function which measures a property difference between an object and a model. This strategy for recovering parametric geons is similar to that for other parametric primitives [**94, 123, 74**]. However, there is an additional requirement for parametric geon recovery. The process must also produce discriminative information such that the resulting metric data can be used to postulate a qualitative or symbolic description. Therefore, we combine model fitting and selection in the model recovery procedure.

A new objective function has been defined which measures: (i) the spatial distance between data points and the model surface, and (ii) the squared difference between the normal vectors of the model and object. This is different from another two-term objective function [**153**], in which the first term is a squared distance. Our modification changes the behaviour of the objective function, thereby enabling an efficient model fitting procedure.

Model fitting is performed by minimising the objective function using a stochastic global optimisation approach, Very Fast Simulated Re-annealing (VFSR), which statistically guarantees finding the global minimum. Yokoya et al. [**153**] employed the classical simulated annealing technique to perform superellipsoid fitting. However, the algorithm we use is much faster than theirs. This is because VFSR permits the 'temperature', a control parameter, to decrease exponentially while the classical simulated annealing can only decrease temperature logarithmically in order to stochastically guarantee a global convergence.

The selection of the model which best fits the data is based on the fitting residuals, rather than on the model parameters. Thus, using parametric geons and the proposed model recovery scheme, we can robustly obtain qualitative shape descriptions from object data even though object shapes do not exactly conform to the shape of parametric geons.

Dickinson et al. [**39**] have proposed a method for recovering volumetric primitives by integrating qualitative and quantitative techniques. Using a range image containing *perfect*

[1] Although geon models contain information pertaining to the aspect ratio and the relative size, they are described in qualitative terms, such as "much smaller", "approximately equal to" and "much longer" [**15**].

geon-like objects as input, they *first* recovered a qualitative geon-based model and *then* fit a deformable superquadric to range data. Their geon models and system output are very similar to ours. However, the major difference between their work and ours is that they compute qualitative and quantitative information in *two consecutive steps*[2], while we derive both kinds of information simultaneously. They use model shape information derived from the first step in the following fitting procedure. This shape information imposes constraints on some model shape parameters such that model fitting can be performed efficiently. However, such part shape information is not available in our case. Thus, global optimisation for finding model parameters is necessary. In addition, their qualitative shape recovery, which is a bottom-up approach [**41**], is problematic when an object containing imperfect geon-like parts is analysed. In our approach, shape approximation can be achieved and parametric geon-based descriptions can be computed from imperfect geon objects. Dickinson et al. have also applied a similar strategy to the more difficult case of intensity images [**38**]. The difficulty with this approach is that in the case of perspective projection, many objects in 3D space can yield similar projections in a 2D image. Thus, they cannot obtain the actual quantitative information from an object's 2D projection. In order to deal with this problem, they used two images as input inspired from stereo vision.

10. Chapter Summary

Our research is motivated by a theory of human image understanding. The Recognition-By-Components (RBC) Theory [**15**] postulates that if an arrangement of a few geons can be recovered from line drawings, then objects can be quickly identified, even when they are occluded, rotated in depth and degraded. We have addressed certain issues from the computational point-of-view and have proposed an alternative approach to qualitative volumetric primitive-based representation. The features of our system include the use of range data as input, a physics-based part segmentation, parametric geons, a shape approximation strategy and a qualitative model recovery procedure. We have listed the general assumptions made in our system and presented the general framework of this research. Finally, comparisons between our research and previous work were highlighted.

[2] They also performed part segmentation in the first step. Here we focus on a comparison of schemes for part model identification.

CHAPTER 4

Object Segmentation into Parts

In this chapter, we propose a new approach to the segmentation of 3D objects into parts. The input is either range data or a list of 3D data obtained by multiview range data integration. Our method segments an object at deep surface concavities, resulting in several sets of 3D data. Each set contains object data that belong to the same physical object part. Motivated by physics, we employ the simulated electrical charge density distribution over the object surface as the surface feature, which differentiates surface concave and convex points. In order to compute the charge density distribution numerically, a finite element model in the form of a closed triangular mesh is created over the object surface. A direct connection graph is then constructed based on the spatial relations between triangles in the triangular mesh as a coordinate system over the object surface. Triangles on part boundaries where the charge densities approach local minima are detected and removed. Thus, the triangulated object surface is decomposed into several parts. The triangles belonging to the same physical part are obtained by a connected component labelling process.

Section 1 describes an analogy between the concave and convex discontinuity and the singularity in the electrical charge density distribution. Section 2 makes assumptions about the object shapes used in this research. Section 3 and 4 present the mathematical formulation and the numerical solution to the computation of the charge density distribution over the object surface, respectively. Section 6 describes the issue of surface triangulation. Section 5 discusses the characteristics of the charge density distribution. Section 7 introduces the direct connection graph and describes the object decomposition algorithm which is based on this graph. A summary is given in the last section.

1. Physics

Following the strategy of boundary-based approaches, we segment an object into parts by extracting object part boundaries. Since the object data are unstructured and do not

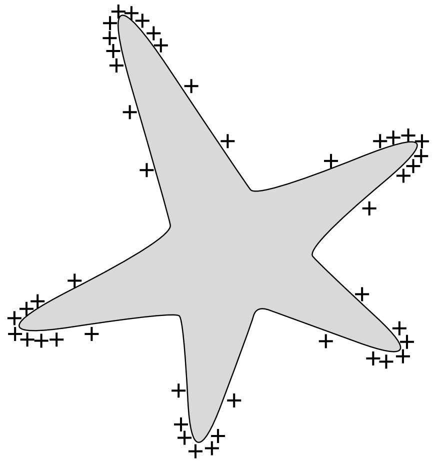


FIGURE 4.1. The charge distribution on a cross-section (shaded area) of a charged conductor. The electrical charge indicated by "+" tends to accumulate at the convex edges and corners and vanish at the concave ones. Thus, the object part boundary, defined by deep surface concavities, can be indicated by significant local charge minima.

provide explicit information about part boundaries, we must seek a principle for locating part boundary points. According to Hoffman and Richards [**60**], the mental category 'part' of shapes is based upon a regularity of nature – transversality – as defined as follows:

DEFINITION 4.1. **Transversality regularity.** *When two arbitrarily shaped surfaces are made to interpenetrate, they always meet in a contour of concave discontinuity of their tangent planes.*

It has been proved that smoothing of such concave discontinuities gives rise to contours of negative minima of a principal curvature [**11**]. Thus, it has been proposed that an object surface can be partitioned into parts along contours of surface concave discontinuity or negative extrema of a principal curvature. Inspired by this principle, we develop a new computational approach to segment an object into parts.

Our algorithm for locating the part boundaries is derived from an analogy between the curvature discontinuity and the electrical charge density distribution over the object surface. When a charged conductor is in electrical equilibrium[1], all charge on a conductor must reside only on its outer surface [**29**]. Electromagnetic theories [**67, 24**] and physical experiments have shown a singular behaviour of charge density distributions (see Figure 4.1). That is, the charge density is very high at the sharp convex edges and corners on the object surface and close to 0 at sharp concavities.

[1]When there is no net motion of charge within the conductor, it is in electrostatic equilibrium.

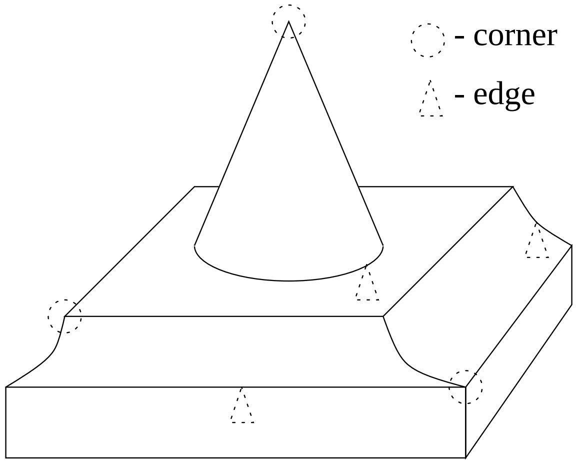
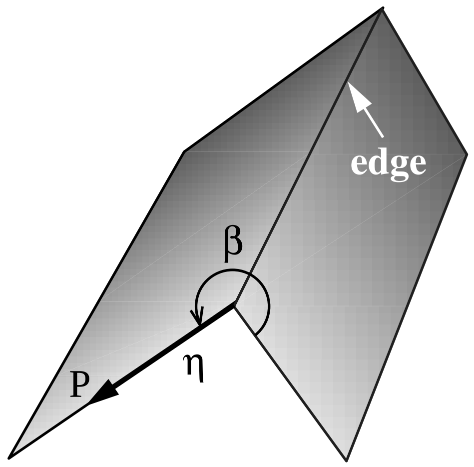


FIGURE 4.2. Examples of edges and corners.

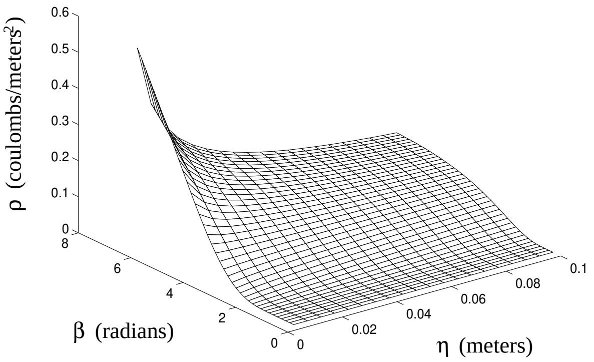
It is important for shape analysis to know the mathematical nature of charge density distributions. Electrical charge densities at sharp edges and corners have been carefully studied by Jackson [**67**] and Van Bladel [**24**]. Here we give a brief description of their findings. Details can be found in two books [**67, 24**]. By edges and corners, they mean orientation (C^1) discontinuities of the object surface. These are physical entities, which are different from those defined in image processing. An edge is an intersection of two smooth surfaces, forming a line or curve segment in 3D space. A corner is an intersection of surfaces, tapering into a point in 3D space. Figure 4.2 depicts examples of edges and corners. By ignoring secondary global effects, the authors have derived the following approximate relationship governing the charge density ρ at an edge formed by two conducting planes, as shown in Figure 4.3 (a):

$$\rho(\eta,\beta) \approx -\frac{a_1}{4\beta}\eta^{(\pi/\beta)-1} \tag{4.1}$$

Here β is the angle between two planes defining the edge. η is the distance from the edge to a point P, where the charge density is measured. a_1 is a constant determined by the approximation used for deriving (4.1). Figure 4.3 (b) shows ρ as a function of β and η. This relation indicates that the larger β and the smaller η, the greater the charge density. We observe that all sections are monotonic for constant η. Note that the power of η is a nonlinear function of β. The theoretical singular behaviour of the charge density at edges



(a)



(b)

FIGURE 4.3. Charge densities at edges. (a) an edge formed by two planes with an angle β. (b) The charge density at $P(\eta, \beta)$.

(for $\eta = 0$) can be stated as follows:

$$\rho = \begin{cases} \infty & \text{if} \quad \beta > \pi \\ \text{constant} & \text{if} \quad \beta = \pi \\ 0 & \text{if} \quad \beta < \pi \end{cases} \tag{4.2}$$

This means that the charge density is infinite, constant and zero when the angle defined by two planes is convex, flat and concave, respectively. The singular behaviour of charge density at corners similar to these edges has also been studied [**67, 24**].

We have observed that at slightly smoothed edges and corners, the positions of local extrema of charge densities are not changed. Consequently, by assuming that a multi-part object is a charged conductor, we can locate the part boundaries at surface points where charge densities reach significant local minima. It is noted that the charge density distribution is not equivalent to the surface curvature. We will make a clear distinction between these two surface properties in Section 5.

2. Assumptions about Object Shapes

In order to perform boundary-based part segmentation using the simulated electrical charge density distribution, we must make assumptions about the object shapes. Obviously, certain assumptions are required by any boundary-based segmentation approach. We note that unlike curvature-based approaches, our method does not assume that the object surface has to be smooth, that is, the second partial derivatives of the surface are continuous [**14**]. The assumptions we make are as follows:

(i) *A part boundary must be explicitly indicated by deep surface concavities* and closed for a complete object. As a counter example, an "elbow" (see Figure 2.2) does not satisfy this assumption because the surface concavity points constitute an open curve. Additional constraints are required to segment such an object.

(ii) *Objects to be segmented must be simply-connected.* That is, the object has no holes[2].

(iii) *We assume a multi-part object.* That is, at least one part boundary satisfying Assumption (i) must exist. Our method locates a part boundary at the local charge density minima. Since all values of charge density are positive, it can only reveal the relative information about concavities or convexities but cannot indicate absolute information. For example, a minimum of the charge density distribution on a convex object will not indicate a surface concavity. This assumption on multi-part objects ensures that at least one deep surface concavity. Curvature-based approaches do not require this assumption.

In the following two sections, we will describe the mathematical formulation of and the numerical solution to the charge density distribution.

3. Computation

Our physical model is the charge density distribution on a charged conductor in 3D free space, where there is no other charge or conductor. To begin with, we list three physical facts which can be derived from physical laws and which we will use to develop the algorithm for the charge density computation.

[2] Although a hole may be viewed as an indented or negative part, we only consider protrusive parts in this thesis.

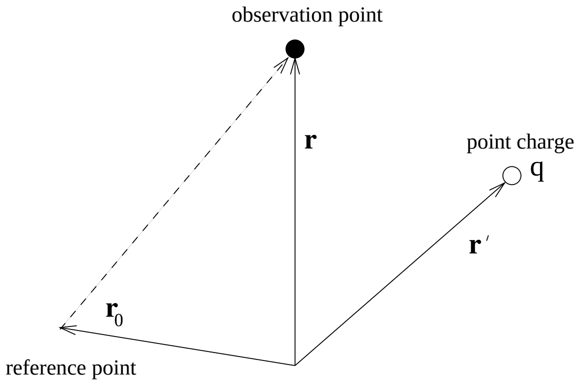


FIGURE 4.4. Configuration of a point charge. $\mathbf{r}$ is the vector position where the electrical potential is observed. $\mathbf{r}'$ is the vector position of the source point charge. $\mathbf{r}_0$ is the vector position of the potential reference point.

FACT 4.1. *In electrical equilibrium, any charge on an isolated conductor must reside entirely on its outer surface* [**112**].

This means that there is no charge inside the conductor. The structure within the object does not affect the charge density distribution. This fact shows that in this case, the charge density distribution is a surface property.

FACT 4.2. *The surface of any charged conductor in electrical equilibrium is an equipotential surface* [**112**].

FACT 4.3. *Conservation of Charge: Charge cannot be created or destroyed, for the algebraic sum of the positive and negative charges in a closed or isolated system does not change under any circumstances* [**78**].

These facts provide us with the conditions needed to establish mathematical equations with charge densities as their variables.

Consider the electrical potential at the vector position $\mathbf{r} \in R^3$, produced by a point charge q, located at the vector position $\mathbf{r}' \in R^3$, as shown in Figure 4.4. The corresponding electrical field at $\mathbf{r}$ can be calculated by an application of Gauss's law. Thus,

$$\mathbf{e}(\mathbf{r}) = \frac{q}{4\pi\epsilon_0}\frac{\mathbf{r}-\mathbf{r}'}{|\mathbf{r}-\mathbf{r}'|^3} \tag{4.3}$$

Here ϵ_0 is a constant, known as the permitivity of free space.

The electrical potential $\phi(\mathbf{r})$ at $\mathbf{r}$ can be derived by an integration of (4.3) along the dashed line from $\mathbf{r}_0 \in R^3$, the vector position of the reference point, to $\mathbf{r}$ (see Figure 4.4):

$$\phi(\mathbf{r}) - \phi(\mathbf{r}_0) = \frac{q}{4\pi\epsilon_0}\left(\frac{1}{|\mathbf{r}-\mathbf{r}'|} - \frac{1}{|\mathbf{r}_0-\mathbf{r}'|}\right) \tag{4.4}$$

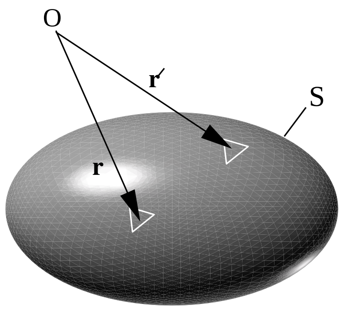


FIGURE 4.5. Configuration of charge distribution over the surface. O is the origin of the coordinate system.

In physics, it is customary to choose the reference potential to be zero at $|\mathbf{r}_0| = \infty$. Accordingly, $\phi(\mathbf{r}_0) = 0$ and Equation (4.4) becomes:

$$\phi(\mathbf{r}) = \frac{q}{4\pi\epsilon_0} \frac{1}{|\mathbf{r} - \mathbf{r}'|} \tag{4.5}$$

Secondly, consider that the charge is continuously distributed over the object surface S (see Figure 4.5). Thus the electrical potential at $\mathbf{r}$ is contributed by *all the charge on S* and satisfies the principle of superposition. It can be expressed as follows:

$$\phi(\mathbf{r}) = \frac{1}{4\pi\epsilon_0} \int_S \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} dS' \tag{4.6}$$

Here $q = \rho(\mathbf{r}')dS'$, $\rho(\mathbf{r}')$ is the charge density at $\mathbf{r}'$, and S' is the area over S.

Thirdly, according to Fact 4.2 that all points on a charged conductor in electrical equilibrium are at the same electrical potential, if we restrict $\mathbf{r}$ in Equation (4.6) to the conductor surface, $\phi(\mathbf{r})$ is constant. Thus, (4.6) may be rewritten as follows:

$$V = \int_S \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} dS' \tag{4.7}$$

Here $V = 4\pi\epsilon_0\phi(\mathbf{r})$ is a constant. In the next section, we will introduce a numerical algorithm for computing the charge density based on Equation (4.7).

4. Finite Element Solution

Our objective is to compute the charge density distribution over the outer surface of an object with an irregular shape. Since S in (4.7) is an arbitrary surface, it is impossible to express the charge density analytically. However, we can obtain an approximate solution to the charge density by using a finite element method [**119**]. The idea is to approximate the 3D object by a polyhedron, each face of which is a planar triangle which possesses a constant charge density. Then the problem of integration over the complete surface (see Equation (4.7)) can be converted into a summation of integrations over each triangle. Since

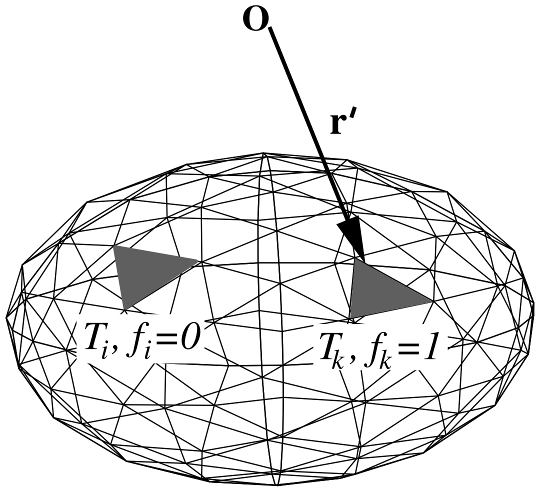


FIGURE 4.6. Polyhedral approximation of an ellipsoid. When $\mathbf{r}'$ is on T_k, $f_k = 1; f_i = 0 (i \neq k)$.

the latter can be solved analytically, the charge density on each triangle can be easily computed.

The finite element solution is obtained as follows. We tessellate the object surface using a triangular mesh which has N planar triangles, $T_k, k = 1, ..., N$. Each triangle is assumed to have a constant charge density, ρ_k, as shown in Figure 4.6. A set of basis functions $f_k, k = 1, ..., N$ is defined on this triangular mesh as follows:

$$f_k(\mathbf{r}') = \begin{cases} 1 & if\ \mathbf{r}' \in T_k \\ 0 & otherwise \end{cases} \tag{4.8}$$

Thus the basis function, f_k, is nonzero only when $\mathbf{r}'$ is on the triangle T_k, as shown in Figure 4.6. Therefore, the charge density $\rho(\mathbf{r}')$ can be approximated by a piecewise constant charge density function as follows:

$$\rho(\mathbf{r}') \approx \sum_{k=1}^{N} \rho_k f_k(\mathbf{r}') \tag{4.9}$$

Substituting (4.9) into Equation (4.7), we have

$$V = \sum_{k=1}^{N} \rho_k \int_{T_k} \frac{1}{|\mathbf{r} - \mathbf{r}'|} dS' \tag{4.10}$$

Since the charge density is assumed to be constant on each T_k, we may take $\mathbf{r}_i$ as the observation point on each T_i and rewrite Equation (4.10) as:

$$V = \sum_{k=1}^{N} \rho_k \int_{T_k} \frac{1}{|\mathbf{r}_i - \mathbf{r}'|} dS';\ i = 1, ..., N. \tag{4.11}$$

According to Fact 4.3, the sum of the charges on each triangle equals the total charge on the surface of the contour. Let Q be the total charge on the contour and S_k be the area of T_k. Then we have

$$Q = \int_S \rho(\mathbf{r}) dS' \approx \sum_{k=1}^{N} \rho_k S_k \tag{4.12}$$

Assuming Q is known, and given (4.11) and (4.12), we obtain a set of linear equations with $N+1$ unknowns, $\rho_1, ..., \rho_N$ and V, as follows:

$$A\rho = \phi \tag{4.13}$$

Here

$$\rho = \begin{pmatrix} \rho_1 \\ \rho_2 \\ . \\ . \\ . \\ \rho_N \\ V \end{pmatrix}; \qquad \phi = \begin{pmatrix} 0 \\ 0 \\ . \\ . \\ . \\ 0 \\ Q \end{pmatrix}; \tag{4.14}$$

and

$$A = \begin{pmatrix} A_{11} & A_{12} & . & A_{1N} & -1 \\ A_{21} & A_{22} & . & A_{2N} & -1 \\ . & & & & . \\ . & & & & . \\ . & & & & . \\ A_{N1} & A_{N2} & . & A_{NN} & -1 \\ S_1 & S_2 & . & S_N & 0 \end{pmatrix} \tag{4.15}$$

where

$$A_{ik} = \int_{T_k} \frac{1}{|\mathbf{r}_i - \mathbf{r}'|} dS', \qquad i, j = 1, 2, ..., N. \tag{4.16}$$

Since the integral in (4.16) can be evaluated analytically [**142**], as shown in Appendix A, the charge density distribution ρ_k and the constant V can be obtained by solving the set of linear equations given in (4.13). Since the potential on a particular triangle is actually contributed by the charge on all of the other triangles, the matrix A is dense. In the actual computation, the observation point $\mathbf{r}_i$ on each triangular patch is selected at its centroid. The set of linear equations is solved by a conjugate gradient squared method [**10**].

5. Characteristics of Charge Density Distribution

Computing the simulated charge density distribution is an intermediate step in the procedure for segmenting an object into parts. In the literature, several researchers have proposed criteria for deriving shape descriptions[**82, 23, 28, 143**]. These include accessibility, uniqueness, stability, scope and sensitivity. In the following, we explain the criteria and examine how well the charge density computation satisfies them.

Uniqueness: It has been proven that the charge density distribution on a charged conductor in electrostatic equilibrium is uniquely determined [**122**]. Thus given a specific shape of an object, our method produces a unique description of the surface property, and in turn, obtains a unique segmentation result based on this surface property.

Invariance: An object description which is invariant to object rotation and translation is important. For example, when an autonomous robot looks at an object from different viewpoints, it must recognise the object as the same entity. Since the charge density distribution depends completely upon the total charge, as well as the shape and size of an object, it is independent of the coordinate system chosen for the computation (see Equation (4.11) and (4.12)). Also the relative position of the extrema in the charge density distribution will not change with object size. Therefore, our part segmentation method is invariant to object scale, translation and rotation.

Versatility: The charge density computation, which is based on integral equations, does not require an assumption on smoothness of object surfaces. However, the surface curvature computation, which is based on differentiation, needs a smooth surface, namely, the continuous second partial derivatives of the object surface [**86**]. Note that the method based on particle diffusion [**151**] also does not need this assumption. However, it suffers from another severe problem, as will be described in the next paragraph.

Computability: The algorithm for the charge density distribution is based upon three physical facts and Gauss' law. There are no crucial user-defined parameters required. After surface triangulation, the total computation only involves analytical function evaluations and a set of linear equations. This approach deals with the electrical equilibrium, which produces distinctive charge densities at concave and convex surface points. However, in the particle diffusion-based approach [**151**], one has to set a threshold, the maximum number of iterations of the diffusion process. This parameter is dependent on object size and shape. It is very difficult to determine because shape information is generally not available beforehand. If diffusion is stopped at an inappropriate time, the distinction between particle densities at concave and convex surface points will not be strong. Moreover, to compute the surface property of a complete 3D object, both curvature-based [**110**] and diffusion-based [**151**] approaches require a voxel-based coordinate system, which is an explicit 3D coordinate system. By contrast, the charge density-based approach restricts its variables only to the outer surface of the object. Therefore, the amount of computation is reduced.

Scope and sensitivity: It is often required that an object description represent shape at different scales. Descriptions at coarse scales relate to the gross shape features. Details at finer scales include features that are more local. The charge density distribution carries information about scales in a different way from the more common curvature-based approach [**5**]. This is because a strict relationship between the charge density distribution and the curvature does not exist [**98**]. The heuristic that the charge density on object surfaces changes monotonically with curvature seems to be quite acceptable. However, we note that Jackson's theoretical result [**67**] (see Section 1) holds only *locally* at a surface with C^1 discontinuity. But the surface at part boundaries may be C^1 continuous or smooth. Curvature is completely determined by local data. But the charge density distribution is affected by all of the points on the object surface. Note that these points do not contribute equally to the potential at a particular observation point. Instead, their influence is weighted by the reciprocal of the distance between the observation and source points (see Equation (4.7)). Thus, a charge density computation possesses both "quasi-global" and "quasi-local" properties.

The "quasi-global" property helps reduce noise effects. In curvature-based approaches, the second partial derivatives of the object surface must be computed in a local neighbourhood. Because of the inherent noise in the input data and the fact that differentiation is required, this kind of approach has proven unreliable [**135**]. However, since the charge density computation uses weighted global shape information, it reduces the sensitivity to noise and produces a more robust result. The "quasi-global" property produces information that can also reveal the significance of the protrusion and indentation of parts.

The "quasi-local" property helps isolate fine local features. Because of this property, the charge density segmentation approach can handle situations of object self-occlusion. When only single-view range data are provided, shape information on the "invisible surface" is not available. The surface triangulation of the "invisible surface" could be rather arbitrary. Nevertheless, it turns out that the local distribution of charge is almost not affected by selecting a different "invisible surface". Therefore, the position of the local minima of charge density remains essentially the same. The property will be justified by experiments in Appendix C.

These properties will be demonstrated by experiments in Chapter 6. Using curvature, these gross and fine features may be detected at different scales by smoothing with different window sizes. It is noted that the selection of an appropriate scale is very difficult. However, our approach produces both features simultaneously and without intervention.

Relative and absolute surface information: For the diffusion-based method and the charge density-based method, since the number of the particles and the charge density are positive values, they can only determine the *relative* contrast between convexities and concavities. However, curvature-based approaches do compute positive and negative surface curvature, namely the absolute concave and convex information. If an object is a single-part object, which does not have negative curvature, a curvature-based method will detect this and not conduct any further segmentation; both the diffusion-based and charge density-based approaches will not be able to do this.

These characterisations will be demonstrated through experiments which will be discussed in Chapter 6.

6. Surface Triangulation

In Section 4, we described the computation of the charge density distribution over an object surface, which we represented in terms of a triangular mesh. Since the image data consist of a set of discrete points in 3D space, it was found necessary to use triangular mesh tessellation. This technique, which is called *surface triangulation* [**53**], has been widely used for reconstructing object shapes [**26, 45, 32, 128**]. A triangular mesh also specifies a data indexing system for the object surfaces, which are represented by a set of discrete 3D points. Thus, triangulation establishes a specific spatial relationship between these points and facilitates the extraction of part boundaries. In this section, we will describe the surface

(a) (b)

FIGURE 4.7. Single-view range data of an object. (a) Frontal view. (b) Side view. Due to self-occlusion, surface information on the other side cannot be seen by the laser rangefinder.

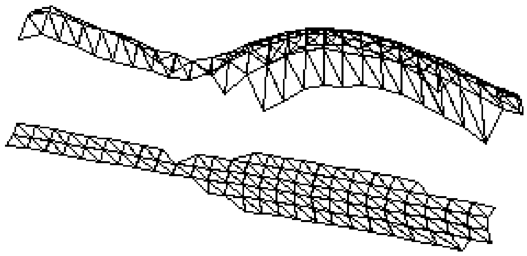
triangulation using either single-view or multiview range data. The details of the algorithm are given in Appendix C. The problem of part decomposition will be addressed in Section 7.

6.1. Constructing a Triangular Mesh for Single-View Data The single-view range data shown in Figure 4.7 indicate object shape information on visible surfaces. The data points $\mathbf{d}_i = \{x_i, y_i, z_i\} \in \mathcal{R}^3$ are specified by a vector function $\mathbf{f}$ of the range image grid, $\mathbf{u}_i$, as follows:

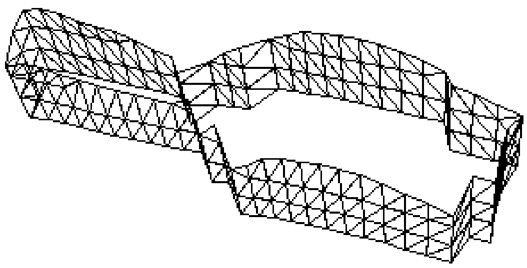
$$\mathbf{d}_i = \mathbf{f}(\mathbf{u}_i), \ i = 1, ..., N,$$

Here $\{\mathbf{u}_i, i = 1, ..., N\} \subset U$ and $U \subset \mathcal{I}^2$ is the range image domain. Thus, not all pixels contain data points. Surface triangulation of the single-view data can be performed in the 2D image domain using the explicit neighbourhood relation specified by the image grid [**126**]. Triangles are constructed within individual 2×2 pixel regions. The triangle vertices are range data points. If the region contains three or four data points, one or two triangles respectively are formed. This local triangulation permits us to establish a triangular mesh for the visible object surfaces. Since this approach is based on using the actual data points, it requires dense data to obtain a connected mesh.

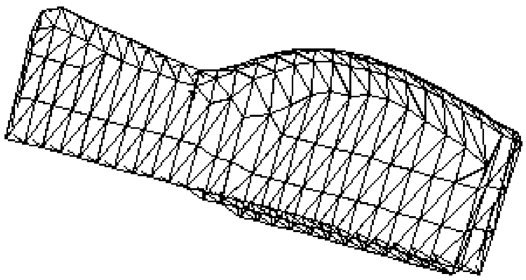
To compute the charge density, a *closed* triangular mesh is required. Since the data points on the invisible surface are not available, as shown in Figure 4.7, it is impossible to directly tessellate the invisible surface using the above method. In practice, we *artificially* construct a mesh on the invisible side in order to make up a closed triangular mesh. As discussed in Section 5, the actual shape of the invisible surface only affects the absolute value of the charge density on the visible surfaces. The position of the extrema of the charge density distribution remains almost the same. Thus, it makes sense to construct an artificial mesh on the invisible surface. For the sake of simplicity, we construct the closed mesh using three patches, as shown in Figure 4.8. The first, called the *top patch*, is obtained by triangulating the range data on the visible surface. The second, called the *bottom patch*, is planar, and is actually the (spatial) projection of the top patch onto an arbitrary plane



(a)



(b)

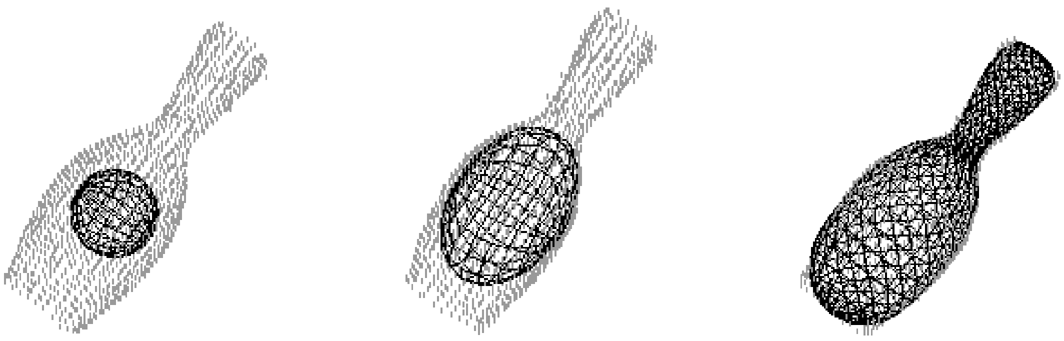


(c)

FIGURE 4.8. Triangulation of the range data in Figure 4.7. (a) shows the top (real data) and bottom patches of the closed triangular mesh. (b) shows the side patch. (c) is the closed triangular mesh obtained by merging (a) and (b).

perpendicular to the Z axis. These two patches are illustrated in Figure 4.8 (a). The third one , called the *side patch*, fills the gap between the top and bottom patches, as shown in (b). The complete closed triangular mesh in Figure 4.8 (c) is obtained by merging the patches in (a) and (b). A similar strategy has been proposed for generating a closed surface in 3D space for diffusion-based shape analysis [**152**].

6.2. Constructing a Triangular Mesh for Multiview Data Multiview data are obtained by transforming multiple single-view range data into a common coordinate system



(a) (b) (c)

FIGURE 4.9. Surface triangulation using multiview range data. Gray dots are data points and the triangular mesh is in black. (a) The initial model. (b) The intermediate model. (c) The final triangular mesh.

(see Appendix B). They contain more complete information about the object shape and are expressed as a sequence of discrete 3D points. But multiview data cannot be uniquely indexed by a particular range image grid, and their neighbourhood relations on the object surface cannot be explicitly specified. They are often referred to as unorganised or unordered data. The method used for tessellating single-view data is not applicable for unorganised data. Other approaches have been proposed [**137, 83, 32, 73, 35**].

In this thesis, an approach based on mesh blending [**35**] is used for surface triangulation [3]. Figure 4.9 illustrates the procedure of model construction. For a given set of data points on the object surface, a triangular mesh representing a spherical shape is initialised. During the model reconstruction, the sphere is deformed towards the shape of the object and residuals between the model and data points are computed. At a certain stage, if residuals between a subset of data and a corresponding local region on the model surface are not reduced no matter how the model is deformed [4], both data points and the mesh model are divided into two sets, having large and small residuals, respectively. A new triangular mesh model is created to represent the subsurface of the old model where large residuals have been measured. This new model is continuously deformed to fit the subset of data which have caused the large residuals. The old model remains and represents the rest of data points. These two models are blended at their intersection by carefully pairing vertices of triangles between models. New models can be further generated to represent finer subsurfaces until the maximum residual is smaller than a predefined threshold. The resultant model is a

[3]The surface triangulation was actually performed by Douglas DeCarlo and Demitri Metaxas at the University of Pennsylvania. We sent them rangefinder data and they produced the triangular meshes.

[4]Since the stiffness of the model surface is specified beforehand, the model surface cannot be deformed by an arbitrary degree.

closed triangular mesh tessellated on the complete object surface. This approach performs surface triangulation *without* the requirement of dense and uniformly-distributed data. This makes it possible to extrapolate the object surface based on the model shape, even when there are no data available due to object self-occlusion. However, for a complicated object, the algorithm may not be able to perform the blending operation properly and, therefore, cannot produce a valid triangular mesh [**35**].

7. Part Decomposition

This section describes the technique for decomposing a whole object into parts, given the charge density distribution over the object surface. After obtaining the simulated charge densities on the object surface, we segment an object into parts by detecting and then deleting points on the part boundaries where the charge densities achieve local minima. For a triangular mesh of multiview data, we decompose the complete mesh. For a single-view range image, only the top patch which represents the visible surface of the object is decomposed. Each resultant part forms a connected triangular mesh which is a subset of the closed triangular mesh. This method is based on a so-called *Direct Connection Graph* (DCG), which serves as a specific coordinate system defined on the triangular mesh. We will first introduce the concept of DCG and then describe the algorithm for object decomposition.

7.1. Direct Connection Graph We first give the definition of *direct neighbours* as follows:

DEFINITION 4.2. *Two triangles are* **direct neighbours** *in a triangular mesh if and only if they share a common side or two vertices.*

Then we define a Direct Connection Graph (DCG):

DEFINITION 4.3. **A Direct Connection Graph** *is a graph defined on a triangular mesh. Its nodes represent the triangles and its branches represent the connections between a node and its direct neighbours.*

Figure 4.10 shows a triangle mesh in (a) and its DCG in (b).

The algorithm for DCG construction is described in Appendix D. Since the DCG provides an explicit neighbourhood relationship between individual triangles on the surface of the object, it is a convenient coordinate system over the object surface. It permits the tracing of the part boundaries on the triangular mesh without employing a voxel-based

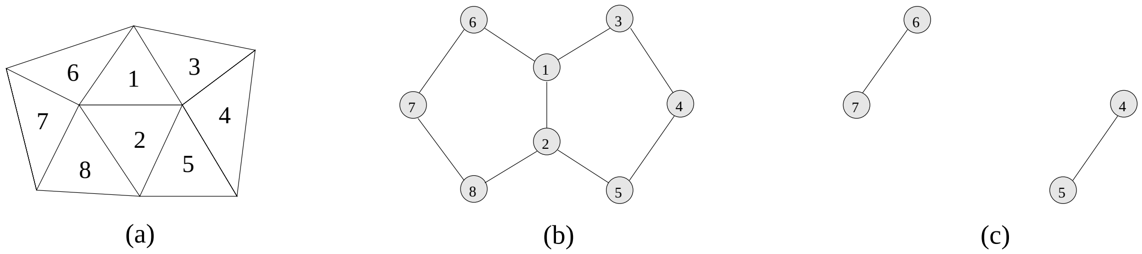


FIGURE 4.10. Direct Connection Graph (DCG). (a) A triangular mesh. For example, triangles 1 and 2 are direct neighbours while 2 and 3 are not. (b) DCG of the triangular mesh in (a). (c) Subgraphs of (b) after boundary node deletion. Here triangular patches 1, 2, 3 and 8 are assumed to be located on the part boundary.

coordinate system. This significantly reduces the required memory space for describing the object and increases the computational speed.

7.2. Finding Parts As described in Chapter 2, the transversality principle states that when two objects interpenetrate, they intersect transversally with probability one[**11**]. This means that the tangent planes to the two intersecting surfaces are of different orientations at *all* points where the surfaces meet (see Figure 2.5). Following this, we have assumed that a part boundary is explicitly defined by deep surface concavities. For a complete object, the part boundary is a *closed* contour. This ensures that the decomposition algorithm will be able to segment a part from the rest of the object. The assumption also provides a stopping criterion for the boundary tracing procedure. Since the part boundary is located at local charge density minima, it can be traced along the "valley" of the charge density distribution. We note that for single-view data, the top patch of triangular mesh is not closed and therefore, the part boundary may not be a closed contour. In this case, when the tracing process reaches the mesh boundary, which has only two direct neighbours, it stops.

The algorithm examines the charge density on all triangles to find an initial triangle for tracing each boundary. An initial triangle must satisfy the following conditions:

(i) It must be a *concave* extremum; that is, its charge density must be a local minimum.

(ii) It must be located at a *deep* concavity. Thus the charge density on the triangle must be lower than a preselected threshold[5].

(iii) It and its neighbours must not have been visited before. This ensures that the same boundary will not be traced again.

[5]This threshold determines when an object should not be decomposed any further. If the charge density at an initial triangle is greater than this threshold, we assume that all boundary points have been found. The selection of the threshold depends on a priori knowledge of the surface concavity and there is no universal rule for selecting it. Obviously, the higher the threshold, the more segmented parts will be found . Currently we choose 120% of the lowest charge density on the object surface as the threshold.

Beginning at the initial triangle, the algorithm proceeds to the neighbour with the lowest charge density. During the tracing procedure, all triangles detected on the boundary are marked. The marked ones will not be checked again and eventually will be deleted from the DCG. For the mesh constructed from *multiview data*, the process continues until it returns to the initial triangle. As a result of the assumption stated at the beginning of this section, this means that all triangles on this part boundary have been visited. For the mesh constructed from *single-view data* as illustrated in Figure 4.10, the process continues until it reaches a triangle(face 3 in Figure 4.10) on the boundary of the mesh. If the initial triangle(face 2 in Figure 4.10) possesses three direct neighbours, the procedure will move in the other direction until reaching a triangle(face 8 in Figure 4.10) on the boundary of the mesh. Thus all triangles on the part boundary have been visited. Next the algorithm finds a new initial triangle and traces another boundary. It repeats the same tracing procedure, and finally stops when the charge density at an initial triangle is higher than the preselected threshold. After all triangles on part boundaries have been found, the nodes of the DCG representing these triangles are deleted. Thus the original DCG is now divided into a set of disconnected subgraphs, as shown in Figure 4.10 (c). Physically the object has been broken into parts. Each object part can be obtained by applying a component labelling algorithm to a subgraph of the DCG. The result of this algorithm is several lists of triangles. Each list contains the triangles which belong to the same object part. These triangle lists are then ready for part model identification. The algorithms for tracing part boundaries and finding individual parts are elaborated in Appendix D.

8. Chapter Summary

In this chapter, we have introduced a novel approach to object segmentation into parts. Following the boundary-based segmentation strategy, we obtain object parts without using part shape information. Rather than geometrical properties, we compute part boundaries using a physical property of object surfaces – the simulated electrical charge density distribution. Assuming that the object considered is a perfect conductor, we computed the charge density distribution over its surface, which has been tessellated by a triangular mesh. The charge density distribution indicates the contrast produced by surface concavities or convexities. We then detect part boundaries at deep concave surface points where the charge density is a local minimum. Finally we decompose the object into parts at the boundary points.

Our approach is unique for a particular shape and invariant to object scale, rotation and translation. It works well on both single-view range data and 3D data integrated from multiple views. There is no crucial parameter that needs to be selected and no assumption is made about the smoothness of object surfaces. Our approach restricts its unknowns to the object's surface instead of the entire 3D space. It does not compute the interior of objects. Unlike previous diffusion-based approaches, this method computes local surface information without the frustration of having to choose a crucial stopping condition. Triangle tessellation of the object surface provides an effective coordinate system over the object surface for part boundary tracing and part labelling. The charge density is determined by global data weighted by the distance to the point where the charge density is being considered. This mechanism permits a more stable solution than pure local feature-based approaches.

We will illustrate the experimental results pertaining to object segmentation in Chapter 6. The next step is to derive part models from segmented parts. This will be presented in the following chapter.

CHAPTER 5

Part Identification

We describe an approach to part model identification, in this chapter. Object segmentation produces a few sets of 3D data. In each set, all data belong to the same object part. The next task is to generate a symbolic object description for each part. To do this, we need to answer the following questions: (1) What kind of part models will be used to represent the object parts? (2) What strategy and technique will be employed to recover the part models? Following the shape approximation scheme discussed in Chapter 3, we propose to use parametric geons as object part models. We formulate model recovery as an optimisation problem. All parametric geon models are fitted to an object part by minimising a function of the difference between the shape and size of a part and the models; the best model for that part is selected based on the minimum fitting residuals. Global optimisation – Very Fast Simulated Annealing – is used for minimisation. The part model, the objective function for optimisation and the optimisation algorithm are described in the following sections.

1. Parametric Geons

1.1. Shape Types Similar to Biederman's geons, the class of parametric geons consists of a finite set of distinct shapes. We believe that these shapes should reflect the essential geometry of objects in the real world. The shapes of the part models are primarily motivated by the art of sculpture, perhaps the most traditional framework for 3D object representation. One of the most obvious features of sculptured objects is that they consist of a configuration of solids of different shapes and sizes which are joined together but which can be perceived as distinct units. The individual volume is the fundamental unit in our perception of sculptural form, as indeed it is in our perception of fully 3D solid form in general [**104**]. Figure 5.1 shows some shape primitives described by sculptors, and objects composed of these primitives. The first column indicates a few 3D shape primitives. The

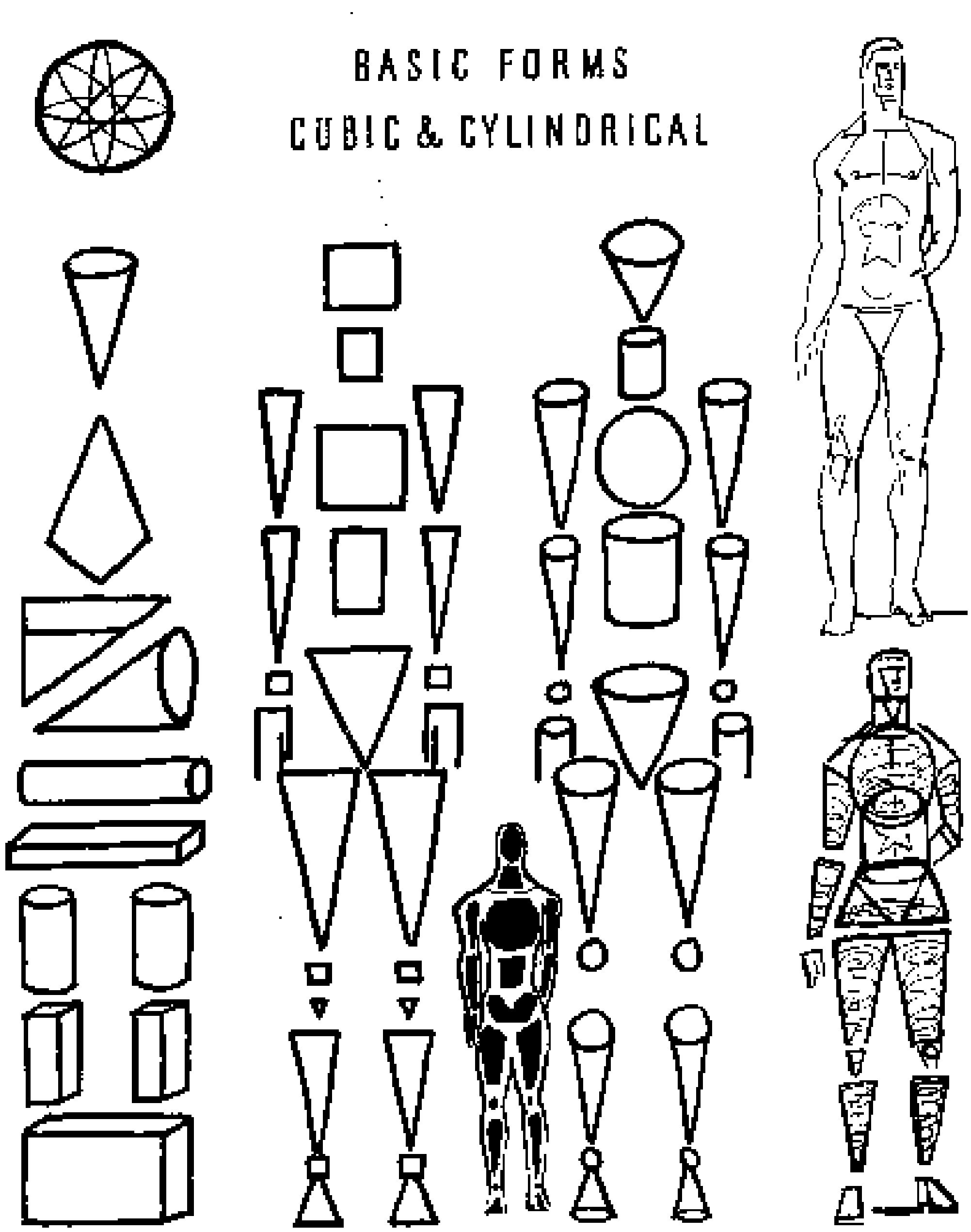


FIGURE 5.1. Shape primitives described by sculptors and objects composed by these primitives. (From W. Zorach, "Zorach Explains Sculpture: What It Means and How It Is Made", Tudor Publishing Company, 1960)

second shows a description of a human body composed of rectangular and triangular primitives. The third column is a representation of the human body in terms of 3D primitives.

The last column indicates the human body and its articulate parts. From a sculptors point of view, all sculptures are composed of variations of five basic forms: the cube, the sphere, the cone, the pyramid and the cylinder [**99, 154**]. Another important belief in the world of sculpture is that each form originates either as a straight line or a curve [**154**]. Straightness and curvature are significant for characterising the main axis of elongated objects and were employed in defining the original geon properties [**15**]. By generalising the five primitive shapes used in sculpture and adding two curved primitives, we arrive at the following seven shapes for parametric geons: the ellipsoid, the cylinder[1], the cuboid, the tapered cylinder, the tapered cuboid, the curved cylinder and the curved cuboid.

1.2. Formulation We choose parametric forms to describe these seven shapes. Their formulations are derived from the superellipsoid equations (2.3) by (i) specifying the shape parameters, ϵ_1 and ϵ_2 and (ii) applying tapering and bending deformations.

1.2.1. *Implicit Equations for the Three Basic Shapes* Since ϵ_1 and ϵ_2 in (2.3) control the degree of "roundness" or "squareness" of superellipsoids in two orthogonal directions, respectively, three of the parametric geons can be derived as follows:

- Given $\epsilon_1 = \epsilon_2 = 1$, the equation of an ellipsoid is

$$\left(\frac{x}{a_1}\right)^2 + \left(\frac{y}{a_2}\right)^2 + \left(\frac{z}{a_3}\right)^2 = 1. \tag{5.1}$$

- Given $\epsilon_1 = 0.1$[2] and $\epsilon_2 = 1$, the equation of a cylinder is given by

$$\left(\left(\frac{x}{a_1}\right)^2 + \left(\frac{y}{a_2}\right)^2\right)^{10} + \left(\frac{z}{a_3}\right)^{20} = 1. \tag{5.2}$$

- Given $\epsilon_1 = \epsilon_2 = 0.1$, the equation of a cuboid is

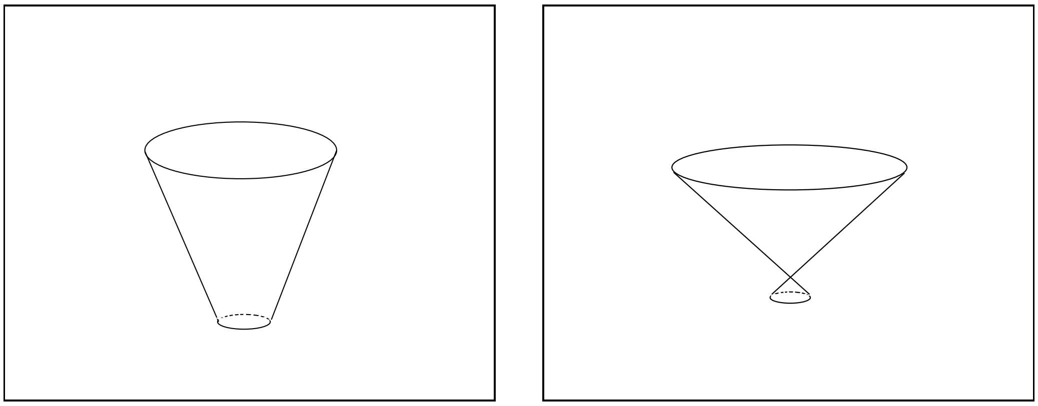
$$\left(\frac{x}{a_1}\right)^{20} + \left(\frac{y}{a_2}\right)^{20} + \left(\frac{z}{a_3}\right)^{20} = 1. \tag{5.3}$$

In the following, we will call these three shapes regular primitives and other shapes deformed primitives.

1.2.2. *Implicit Equations for Tapered Shapes* Two assumptions are made regarding the tapering formulation: (i) tapering deformation is performed along the z axis; (ii) the tapering rate is linear with respect to z. Although this linearity assumption is sometimes

[1] Actually this can be a cylindrical shape with an elliptical cross-section.

[2] Superellpsoid shape changes smoothly with ϵ_1 and ϵ_2. We choose $\epsilon_1 = 0.1$ for a cylinder, based on computational robustness and the perceptual acceptance of its shape. The same reasoning applies to the cuboid.



(a) (b)

FIGURE 5.2. Tapering deformation (a) Downward tapering; (b) Invalid tapering deformation.

violated for real objects, our model is only designed to approximate the shape of tapered object parts. Thus, tapering deformation is given by

$$(5.4)\qquad \begin{cases} X &= (\frac{K_x}{a_3}z+1)x \\ Y &= (\frac{K_y}{a_3}z+1)y \end{cases}$$

where X and Y are the transformed coordinates of the primitives after tapering has been applied to the coordinates x and y. K_x, K_y are tapering parameters in the x and y coordinates. The equation of inverse tapering is given by:

$$(5.5)\qquad \begin{cases} x &= \frac{X}{(\frac{K_x}{a_3}z+1)} \\ y &= \frac{Y}{(\frac{K_y}{a_3}z+1)} \end{cases}$$

To permit downward tapering only in the formulation and avoid invalid tapering (see Figure 5.2), we impose the constraints $0 \leq K_x \leq 1$ and $0 \leq K_y \leq 1$.

By substituting (5.4) into (5.2) and (5.3), respectively, we obtain implicit equations for a tapered cylinder and cuboid, respectively, as follows:

$$(5.6)\qquad \left(\left(\frac{X}{a_1(\frac{K_x}{a_3}Z+1)}\right)^2 + \left(\frac{Y}{a_2(\frac{K_y}{a_3}Z+1)}\right)^2\right)^{10} + \left(\frac{Z}{a_3}\right)^{20} = 1$$

$$(5.7)\qquad \left(\frac{X}{a_1(\frac{K_x}{a_3}Z+1)}\right)^{20} + \left(\frac{Y}{a_2(\frac{K_y}{a_3}Z+1)}\right)^{20} + \left(\frac{Z}{a_3}\right)^{20} = 1$$

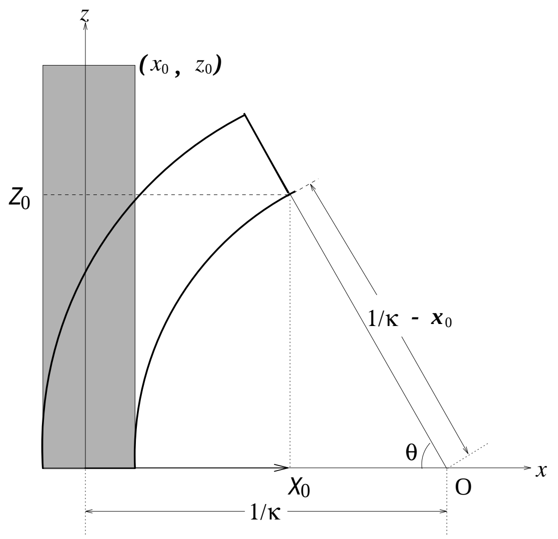


FIGURE 5.3. Bending deformation in the xz plane. Axis y is perpendicular to this plane, projecting into the paper. The shaded area delimits the original primitive. The thick lines depict the curved primitive. O is the centre of bending curvature and θ is the bending angle. Point (x_0, z_0) is transformed into the coordinate (X_0, Z_0) by the bending operation.

1.2.3. *Implicit Equations for Curved Shapes* We use a simple bending operation which corresponds to a circular section, as shown in Figure 5.3. This bending feature is described by only one parameter, i.e. the curvature κ of the circular section. Although many curved object parts do not have constant curvature, we can still amply approximate curved object parts using this qualitative shape model. The bending operation is applied along the z axis in the positive x direction. The operation transforms vectors (x, y, z) into vectors (X, Y, Z). The equations describing the bending deformation are given by (see Figure 5.3):

$$\left\{ \begin{array}{rcl} X & = & \kappa^{-1} - \cos\theta(\kappa^{-1} - x) \\ Y & = & y \\ Z & = & (\kappa^{-1} - x)\sin\theta \end{array} \right. \tag{5.8}$$

Here $\theta = \kappa z$ is the bending angle. The inverse transformation is given by

$$\left\{ \begin{array}{rcl} x & = & \kappa^{-1} - \sqrt{Z^2 + (\kappa^{-1} - X)^2} \\ y & = & Y \\ z & = & \kappa^{-1}\theta = \kappa^{-1}\arctan\frac{Z}{\kappa^{-1} - X} \end{array} \right. \tag{5.9}$$

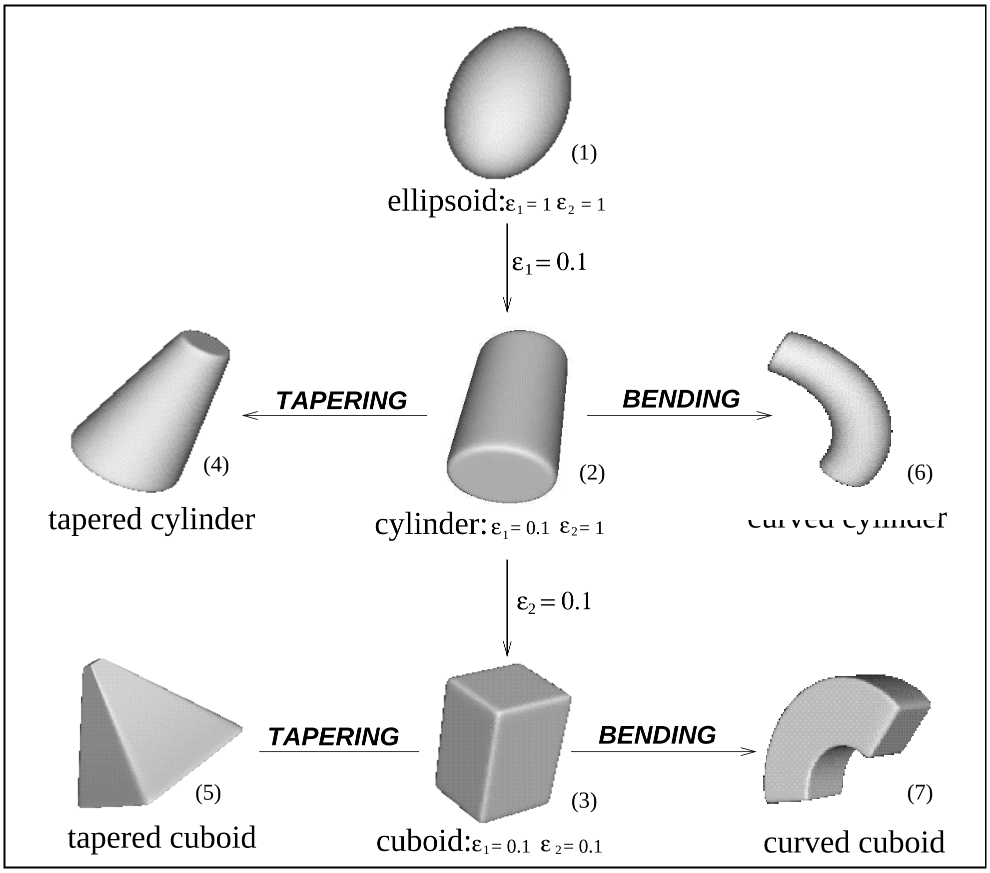


FIGURE 5.4. The seven parametric geons. The numbers in the brackets at the bottom right of each shape represent the index of parametric geon shape types, 1-ellipsoid, 2-cylinder, 3-cuboid, 4-tapered cylinder, 5-tapered cuboid, 6-curved cylinder, 7-curved cuboid.

The equations for curved cylinders and cuboids, as given in (5.10) and (5.11), can be obtained by substituting (5.9) into (5.2) and (5.3):

$$\left(\left(\frac{\kappa^{-1}-\sqrt{Z^2+(\kappa^{-1}-X)^2}}{a_1}\right)^2+\left(\frac{Y}{a_2}\right)^2\right)^{10}+\left(\frac{\kappa^{-1}\arctan\frac{Z}{\kappa^{-1}-X}}{a_3}\right)^{20}=1 \tag{5.10}$$

$$\left(\frac{\kappa^{-1}-\sqrt{Z^2+(\kappa^{-1}-X)^2}}{a_1}\right)^{20}+\left(\frac{Y}{a_2}\right)^{20}+\left(\frac{\kappa^{-1}\arctan\frac{Z}{\kappa^{-1}-X}}{a_3}\right)^{20}=1 \tag{5.11}$$

The seven typical shapes of the parametric geons are illustrated in Figure 5.4. Although these seven shape types are defined qualitatively, their variations can represent a variety of different shapes. Other examples of parametric geon shapes are shown in Figure 5.5.

1.2.4. *Normal Equations* A normal vector at a point on the surface of the parametric geons can be computed from their implicit equations given in (5.1), (5.2), (5.3), (5.6), (5.7),

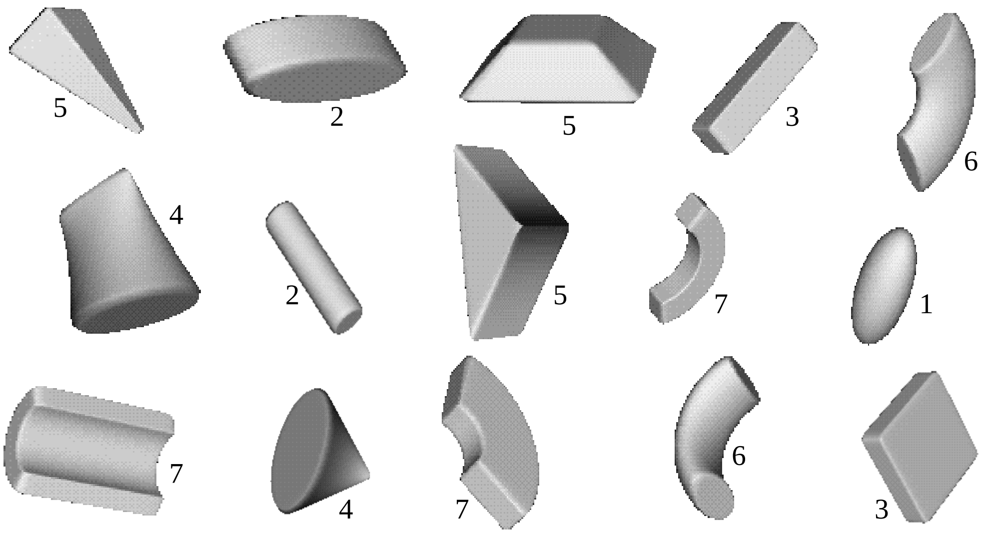


FIGURE 5.5. Some variations of parametric geon shapes. The number beside each shape indicates its geon type defined in Figure 5.4.

(5.10) and (5.11). Let an implicit equation of a parametric geon be defined as $g(\mathbf{x}, \mathbf{a}) = 0$, where $\mathbf{x} = \{x, y, z\}^T$ is the point on the model surface and $\mathbf{a}$ is a parameter vector. A normal vector on the surface of parametric geons is given as follows:

DEFINITION 5.1. *The* **normal vector** *to a parametric geon at a point* $\mathbf{x}$ *is given by the gradient vector*

$$\mathbf{n}_m = \left\{ \frac{\partial g(\mathbf{x}, \mathbf{a})}{\partial x}, \ \frac{\partial g(\mathbf{x}, \mathbf{a})}{\partial y}, \ \frac{\partial g(\mathbf{x}, \mathbf{a})}{\partial z} \right\}. \tag{5.12}$$

An alternative and simpler approach to computing normals for deformed primitives is to apply a transformation to the normal vectors of the three regular shapes. Let tapering or bending be expressed by the equation

$$\mathbf{X} = \mathbf{F}(\mathbf{x}) \tag{5.13}$$

where $\mathbf{X}$ is the transformed point of $\mathbf{x}$. The normal vectors on the surface of deformed parametric geons are given as follows:

DEFINITION 5.2. *A normal vector for deformed parametric geons is given by*

$$\mathbf{n}_m^{\mathbf{X}} = B\mathbf{n}_m^{\mathbf{x}} \tag{5.14}$$

where $B = (detJ)J^{-T}$ is the inverse transpose of the Jacobian matrix of the deformation function and J denotes the Jacobian matrix [**9**] whose ith column is obtained by the partial derivative of $\mathbf{F}(\mathbf{x})$ with respect to ith component in $\mathbf{x}$ as follows:

$$J(\mathbf{x}) = \left\{ \frac{\partial \mathbf{F}(\mathbf{x})}{\partial x}, \frac{\partial \mathbf{F}(\mathbf{x})}{\partial y}, \frac{\partial \mathbf{F}(\mathbf{x})}{\partial z} \right\} \tag{5.15}$$

The determinant of J can be ignored because only the direction of the normals is important.

The normal transformation matrix for tapered primitives can be obtained by applying (5.15) to (5.4) as follows:

$$J^{-T} = \begin{pmatrix} \frac{k_y}{a_3}z + 1 & 0 & 0 \\ 0 & \frac{k_x}{a_3}z + 1 & 0 \\ -(\frac{k_y}{a_3}z + 1)\frac{k_x}{a_3}x & -(\frac{k_x}{a_3}z + 1)\frac{k_y}{a_3}y & (\frac{k_x}{a_3}z + 1)(\frac{k_y}{a_3}z + 1) \end{pmatrix} \tag{5.16}$$

The normal transformation matrix for curved primitives can be obtained by applying (5.15) to (5.8) as follows:

$$J^{-T} = \begin{pmatrix} k(k^{-1} - x)\cos\theta & 0 & \sin\theta \\ 0 & k(k^{-1} - x) & 0 \\ -k(k^{-1} - x)\sin\theta & 0 & \cos\theta \end{pmatrix} \tag{5.17}$$

Given the normal vectors for the regular primitives obtained from (5.12), one can multiply them by either (5.16) or (5.17) to obtain the normal vectors for tapered and curved primitives, respectively. A more detailed discussion of the global deformation of solid shapes can be found in [**9, 123**].

2. Comparison with Original Geons

The major distinction between parametric geons and the conventional geons of Biederman is that the latter are defined in terms of certain specific attributes of volumetric shapes, which do not provide global shape constraints. In contrast, parametric geons are defined in terms of analytical equations, which do provide such constraints. In addition, geons are described in strictly qualitative terms. However, parametric geon descriptions simultaneously supply both qualitative and quantitative characterisations of object parts.

The geometrical differences between these two sets of primitives are given in Table 5.1. Certain qualitative properties of the parametric geons are simplified in comparison with the original geons of Biederman. For example, an asymmetrical cross section is not used in defining any of the parametric geons because of the symmetrical nature of superellipsoid shapes. Biederman has also stated [**16**]:

> *Given that a convex volume is parsed from matching adjacent concavities, it may not be necessary to assume geons with asymmetrical cross-sections ... To my knowledge, there are no cases where basic level classification requires*

ATTRIBUTES	PARAMETRIC GEONS	GEONS
cross sectional shape	symmetrical	symmetrical, asymmetrical
cross sectional size	constant, expanding	constant, expanding, expanding & contracting
combination of properties	either tapering or bending	both tapering and bending

TABLE 5.1. Difference in qualitative properties between parametric geons and Biederman's original geons.

the presence of an asymmetrical cross-section. This does not mean that a component of an exemplar could not have an asymmetrical cross-section, but that primal access need not depend on the preservation of this asymmetry in the image.

The assumption that all parametric geons are symmetrical with respect to their major axes is also consistent with the well-known human perceptual tendency toward phenomenological simplicity and regularity [**59**]. Symmetrical primitives have also been employed in alternatives to the original geons discussed by other researchers [**100, 40**].

3. The Objective Function

The strategy for recovering parametric geons bears some resemblance to that for other parametric primitives. That is, a fitting scheme is used to minimise an objective function which measures some property difference between an object and a model [**55, 74, 94, 123, 153**]. The procedure for fitting parametric geons is formulated as a functional optimisation (minimisation) problem as follows:

PROBLEM 5.1. *Given an objective function*

$$\mathcal{E}(\mathbf{a}) : \mathcal{R}^n \to \mathcal{R}$$

having $\mathbf{a} \in \mathcal{R}^n$ *as a model parameter set, find a particular set of model parameters* $\mathbf{a}^* \in \mathcal{R}^n$ *for which*

$$\mathcal{E}(\mathbf{a}^*) \leq \mathcal{E}(\mathbf{a}), \qquad \textit{for all } \mathbf{a} \neq \mathbf{a}^*.$$

Besides model fitting, there is an additional requirement for parametric geon recovery. The process must also produce *discriminative* information such that the resultant metric data can be converted to a qualitative description. The objective functions studied previously by several researchers were neither intended nor used for this purpose. However, the magnitude of fitting residuals has been used to guide volumetric segmentation [**57**]. To identify individual qualitative shapes based on fitting residuals, we require that the values of the objective function correctly reflect the difference in size and shape between the object data and the parametric models. When a model and an object are close to being the same shape, the objective function should produce a small residual value. When a model is fitted to another class of objects, this same objective function should give a large residual value.

Our objective function consists of two terms expressed as follows:

$$\mathcal{E} = t_1 + \lambda\gamma t_2 \tag{5.18}$$

The first term, t_1, measures the distance between object data points and the model surface; the second term, t_2, measures the squared difference between the object and model normals. λ and γ are parameters which controls the contribution of t_2 made to the objective function. When the model and object pose are the same, the intuitive interpretation of these two terms corresponds to size and shape similarity, respectively. This is a modified version of the objective function proposed in [**153**]. We change the first term from an L_2 norm to an L_1 norm in order to conduct an efficient search (see Section 3.3). In addition, we employ a different weighting coefficient for the second term to be able to discriminate the different objective function values.

3.1. The Distance Measure The first term of the objective function is given by

$$t_1 = \frac{1}{N}\sum_{i=1}^{N} |e(\mathbf{d}_i, \mathbf{a})| \tag{5.19}$$

Here N is the number of data points, $\{\mathbf{d}_i \in R^3, i = 1, ..., N\}$ is the set of data points described in terms of the model coordinate system, and $\mathbf{a}$ is the vector of model parameters.

For the three regular primitives (ellipsoid, cylinder and cuboid), $e(\mathbf{d}_i, \mathbf{a})$ is defined as the Euclidean distance from a data point to the model surface along a line passing through the origin O of the model and the data point [**55, 140**] (see Figure 5.6). Let $\mathbf{x}_s = l\mathbf{d}_i$ where l is a scalar and $\mathbf{x}_s$ is the model surface point on the line joining $\mathbf{d}_i$ and O. A is the distance from $\mathbf{d}_i$ to O. Substituting $\mathbf{x}_s$ into Equations (5.1), (5.2) and (5.3), we obtain

$$e(\mathbf{d}_i, \mathbf{a}) = A\left(1 - \frac{1}{[g(\mathbf{d}_i, \mathbf{a})]^{1/p}}\right) \tag{5.20}$$

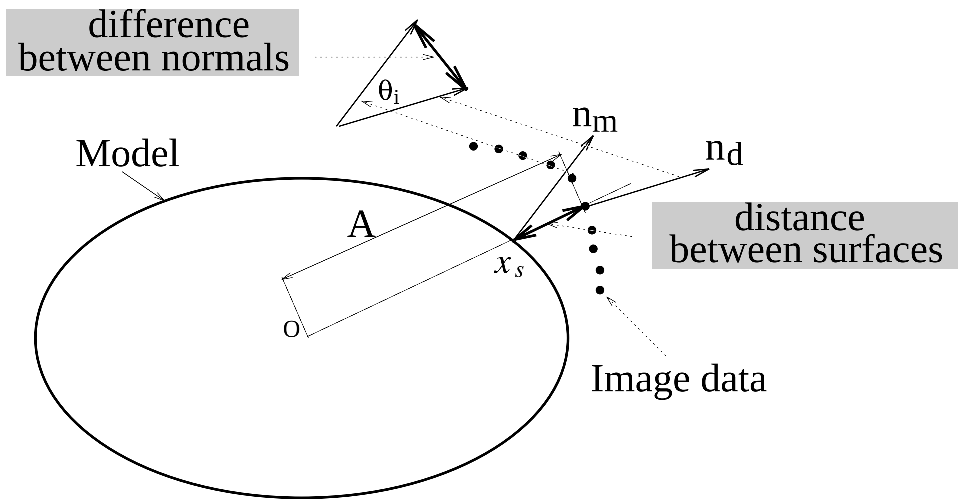


FIGURE 5.6. Defining the objective function. n_m and n_d are the model and data surface normals, respectively . O is the origin of the model. A is the distance between a particular data point and the centre of the model. x_s is a point on the model surface. θ_i is the angle between a model and object surface normals.

where

$$p = \begin{cases} 2 & \text{for the ellipsoid,} \\ 20 & \text{for the cylinder and cuboid,} \end{cases}$$

For tapered and curved primitives, the computation of $e(\mathbf{d}_i, \mathbf{a})$ can be formulated as follows. Let $e(\mathbf{d}_i, \mathbf{a}) = \delta$ and let

$$g(\mathbf{x}_s, \mathbf{a}) = 0 \tag{5.21}$$

be the implicit equation of the model. We can also write (see Figure 5.6)

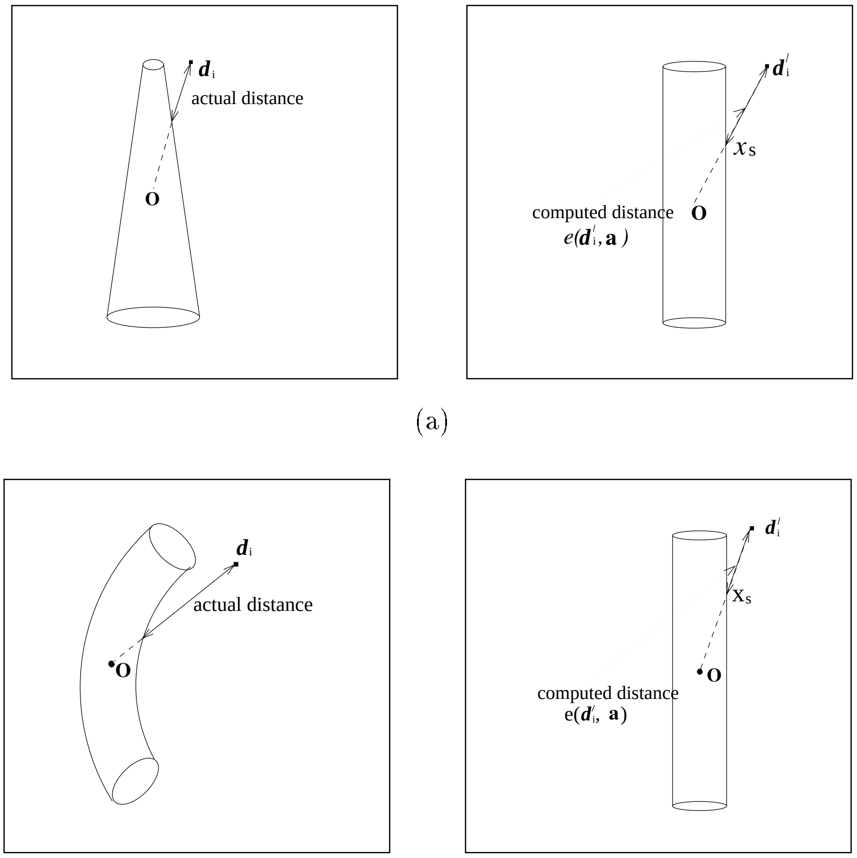
$$\mathbf{x}_s = \frac{A - \delta}{A}\mathbf{d}_i. \tag{5.22}$$

Substituting Equation (5.22) into Equation (5.21) we get

$$g(\frac{A - \delta}{A}\mathbf{d}_i, \mathbf{a}) = 0 \tag{5.23}$$

The problem is as follows: find the minimum value [1] of $\delta \geq 0$, such that Equation (5.23) is satisfied. Since tapering or bending significantly complicates the implicit equations of the deformed primitives, we cannot obtain a closed-form solution for δ or $e(\mathbf{d}_i, \mathbf{a})$, as was done in Equation (5.20). Thus an iterative method would be indicated. However, objective function evaluation is the largest computational component of the model recovery procedure. Hence, for the sake of simplicity, we compute an approximate distance measure for the tapered and

[1] There are at least two intersections of the model surface and the line joining O and $\mathbf{d}_i$. δ is the distance from $\mathbf{d}_i$ to the closest intersection.



(a)

(b)

FIGURE 5.7. The cylinders on the right in (a) and (b) are obtained by applying inverse tapering and bending transformations to the left tapered and curved cylinder, respectively. $e(\mathbf{d}'_i, \mathbf{a})$ is the Euclidean distance along a line $O\mathbf{d}'_i$ in the inverse transformed case.

curved models. No iteration is required. First, we apply an inverse tapering (see (5.5)) or bending transformation (see (5.9)) to both the data and the model in order to obtain the transformed data $\mathbf{d}'_i$ as shown in Figures 5.7; this gives either a regular cuboid or regular cylinder. Second, we use (5.20) to compute the distance from the transformed data point $\mathbf{d}'_i$ to the transformed model surface along a line passing through $\mathbf{d}'_i$ and the model origin O. We interpret $e(\mathbf{d}'_i, \mathbf{a})$ as the approximation of the distance along a line from $\mathbf{d}_i$ to the model surface. Although this approximation creates a small error in the distance measure, it tremendously speeds up computation. Another advantage of this approximation is that one can still use the same Equation (5.20) in the case of regular primitives and replace the

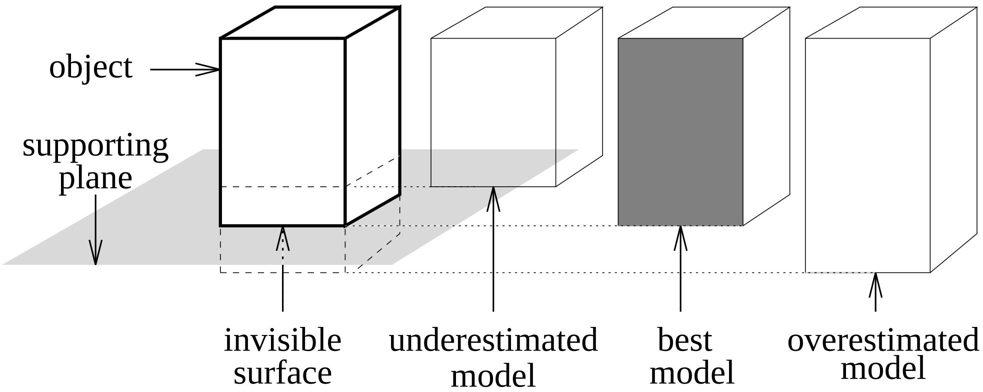


FIGURE 5.8. Since data on the bottom surface is not available due to occlusion, two models - the best model and the overestimated model - could be fit to the data equally well without employing γ. If γ is used, the underestimated model causes both terms in the objective function to be large, while the overestimated model causes the second term to be large. The best model results in the smallest residual value of the objective function.

function $g[\mathbf{d}_i, \mathbf{a}]$ with that of deformed primitives. We note that the objective function used in [**123**] does not have this problem; however, it is not a true distance measure [**55**].

3.2. The Normal Measure We define the second term (t_2) of the objective function by measuring a squared difference between the surface normal vectors $\mathbf{n}_d$ of objects and the surface normal vectors $\mathbf{n}_m$ of models at each corresponding position, defined in the same way as in the first term (see Figure 5.6):

$$t_2 = \frac{1}{N}\sum_{i=1}^{N} e_n(i) \tag{5.24}$$

Here N is the number of data points and

$$e_n(i) = \|\mathbf{n}_d(i) - \mathbf{n}_m(i)\|^2 . \tag{5.25}$$

The $\mathbf{n}_d$ are computed from range image data and $\mathbf{n}_m$ are computed based on methods described in Section 1.2.

In (5.18), $\gamma = (a_x + a_y + a_z)/3$, which makes the second term adapt to the size of the parametric geons, and a_x, a_y and a_z are model size parameters. The units for the first term of the objective function is millimetres but t_2 is the average of differences of unit normals. When multiplied by γ, γt_2 has the same units as the first term. This factor also forces the selection of a model with a smaller size if object data are fit equally well by a model with different parameter sets. This can occur when the data on the bottom surface of an object cannot be obtained. Figure 5.8 demonstrates that the overestimated model and the best model can produce the same residual value of t_2. By multiplying γ, the model having the smaller γ gives a smaller residual than the overestimated model. However, the size of the model is prevented from being arbitrarily small since the value of the objective function increases if the the model size is smaller than the object size. This feature is similar to that of the volume factor used in superellipsoid recovery [**123, 153**].

In (5.18), the weighting constant λ, controls the contribution of the second term to the objective function. It is very difficult to determine this parameter on the basis of specified principles. If it is chosen too small, the second term, t_2, will almost not make its contribution to the objective function. If it is chosen too large, t_2 will dominate the objective function. In this case, the fitting procedure cannot obtain accurate size parameters because t_2 is independent of the model size. Yokoya et al. arbitrarily chose $\lambda = 1$ for their two-term objective function [**153**]. We would like the value selected for λ to permit the model fitting procedure to produce the most discriminative residuals. Since the objects may possess arbitrary shapes, there seems to be no general rule for selecting λ. Based on the ideal shape comparisions between certain models, we use $\lambda = 5$ [**144**].

3.3. Biasing the Objective Function with Different Norms We have suggested an L_1 norm in (5.19) and L_2 norm in (5.25) to measure differences in distance and orientation, respectively. It is known [**58**] that the sensitivity of an L_2 norm gradually increases. In other words, this norm is insensitive to small values of the objective function and becomes sensitive to outliers. On the other hand, the sensitivity of an L_1 norm is the same for all residual values. When the shape types or the pose of the objects and models are very different, the data which are far from the model surface can be viewed as outliers. Thus, the first term with its L_1 norm makes a much smaller contribution to the objective function than it would an L_2 norm. In this case, the second term, in the form of the L_2 norm, is very large and dominates the objective function. With the objective function defined in this way, we can achieve efficient model recovery which will be discussed in Section 5.

4. Minimising the Objective Function

4.1. Optimisation Technique The procedure for fitting parametric geons is a search for a particular set of parameters $\vec{a}$, which minimises the objective function in (5.18). This function has a few deep and many shallow local minima indicated in Figure 5.9. The deep local minima are caused by an inappropriate tapering, bending or rotation parameters of the model. The shallow minima are caused by noise and minor changes in object shape. In order to obtain the best fit of a model to an object, we need to find model parameters corresponding to the *global* minimum of the objective function. To accomplish this, we employ a stochastic optimisation technique, Very Fast Simulated Re-annealing

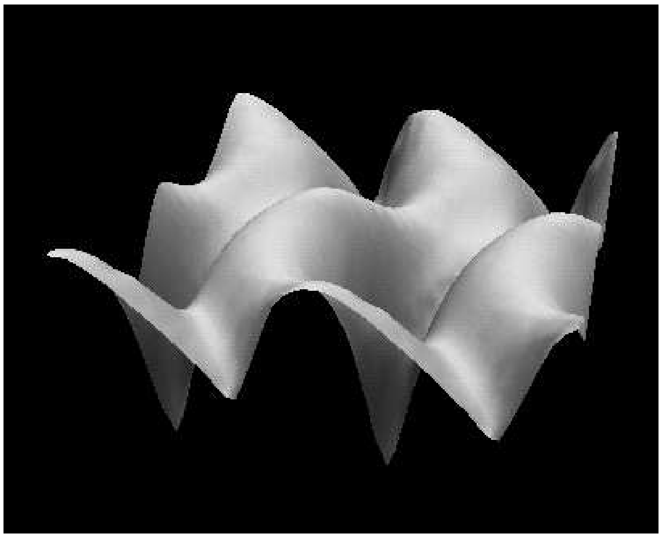


FIGURE 5.9. The logarithm of the objective function in terms of two rotation parameters. The actual parameter space is in from nine to eleven dimensions.

(VFSR) [**66**]. Motivated by an analogy to the statistical mechanics of annealing in solids, the simulated annealing technique uses a "temperature cooling" operation for non-physical optimisation problems, thereby transforming a poor solution into a highly optimised, desirable solution[**71**]. The salient feature of this approach is that it statistically finds a global optimal solution. VFSR uses an annealing schedule which decreases exponentially, making it much faster than traditional (Boltzmann) annealing [**71**], where the annealing schedule decreases logarithmly. The re-annealing property permits adaptation to changing sensitivities in the multidimensional parameter space. Using VFSR, we can reliably and efficiently obtain parameters which describe the best fit between models and data based on our objective function.

Some researchers have used a nonlinear least squares minimisation (Levenberg- Marquardt) method, adding random walks to escape local minima [**55, 123, 131**]. This is similar to simulated annealing but with an extremely fast annealing schedule. In some cases, where the properties of the objective functions are known or a good initial parameter estimation can be obtained, this approach will usually take much less time than the general global optimisation methods. However, using an inappropriate initial guess with an extremely fast annealing schedule may trap the algorithm at a local minimum. This is because global convergence cannot be assured. with VFSR, only a coarse range is needed for each parameter. In addition, methods which require a good initial estimate must also

parameter	lower bound	upper bound
a_x	$l/2$	$L/2$
a_y	$l/2$	$L/2$
a_z	$l/2$	$L/2$
t_x	$c_x - d$	$c_x + d$
t_y	$c_y - d$	$c_y + d$
t_z	$c_z - d$	$c_z + d$
r_x	$-\pi$	π
r_y	$-\pi$	π
r_z	$-\pi$	π
k_x	0	1
k_y	0	1
κ	0	$2/h$

TABLE 5.2. Parameter constraints.

assume that bending and tapering deformation takes place along only the longest side [**123**]. This assumption restricts the object shapes to be recovered. By using a global optimisation method, we do not necessarily need to impose this constraint. Therefore, volumetric primitive models can be recovered from more shapes.

A classical simulated annealing algorithm [**71**] has been used for parametric model fitting [**153**]. However, this algorithm is too slow because it decreases temperature logarithmically. The algorithm VFSR we used decreases temperature exponentially and is much faster than the classical simulated annealing.

4.2. Determining the Parameter Space For any optimisation problem, the range of the parameters defining the objective function must be known beforehand. Constraints for a total of 12 parameters are specified as shown in Table 5.2. In order determine the range of the size parameters, a_x, a_y, a_z, we have calculated a rectangular region in 3D space bounded by maximum and minimum x, y, z coordinates $\{X_{max}, X_{min}, Y_{max}, Y_{min}, Z_{max}, Z_{min}\}$ of range data shown in Figure 5.10. The maximum dimension in this space is

$$L = \sqrt{(X_{max} - X_{min})^2 + (Y_{max} - Y_{min})^2 + (Z_{max} - Z_{min})^2} \tag{5.26}$$

$l > 0$ is the minimum possible length of objects. The centroid (c_x, c_y, c_z) of the data set is calculated to estimate the translation parameters, t_x, t_y, t_z. d is the deviation from the

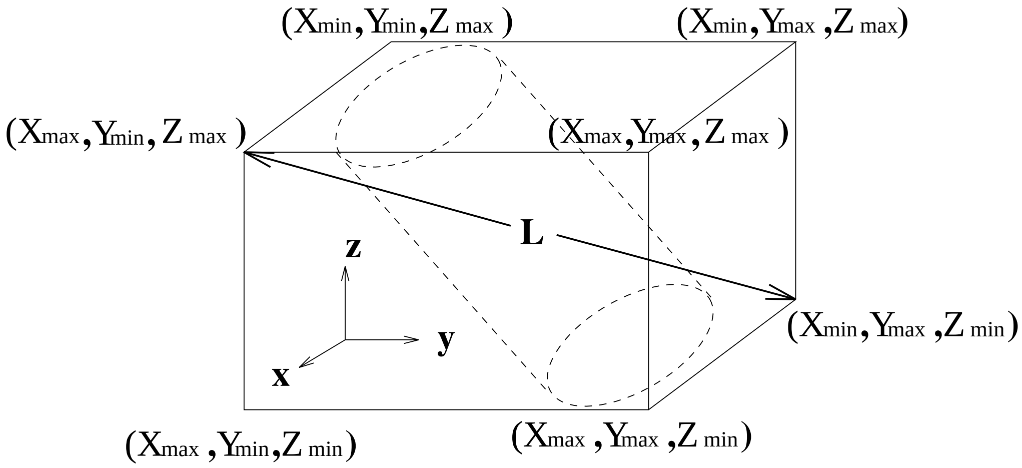


FIGURE 5.10. A cylindrical part enclosed by a rectangular region in 3D space which is used for estimating the range of the size parameters.

centroid. l and d are free parameters set according to *a priori* knowledge. Since the upper bound of the bending curvature κ can be set to the inverse of the minimum possible radius, we select $h = \min(X_{max} - X_{min}, Y_{max} - Y_{min}, Z_{max} - Z_{min})$ as the minimum diameter of the bent sector. Thus $h/2$ is the minimum possible radius. Rotation parameters, r_x, r_y and r_z are set to the range of $[0, 2\pi)$. Tapering parameters, k_x, k_y, are set to their valid range (see Section 1.2.2).

4.3. Stopping Conditions A practical issue in using simulated annealing is to select an appropriate condition for stopping the process. The parameter search procedure, done with VFSR, stops when any of the following conditions is reached.

(i) Smallest temperature value.
(ii) Minimum value of the objective function.
(iii) Maximum number of times sampling the same point.
(iv) Maximum number of times of state acceptance.
(v) Maximum number of evaluations of the objective function.
(vi) Approximate relative difference between two objective function values.

All parameter values are given in Chapter 6.

5. Discussion

In Section 3.2, we defined the objective function as a sum of a distance measure (t_1) and a normal measure (t_2) in terms of L_1 and L_2 norms, respectively. In Section 3.3, we indicated that the L_1 norm is less sensitive to outliers than the L_2 norm. It is also known that the absolute size of a model is independent of the measurement of the differences between normals. These properties can be used to construct an efficient parameter search during model fitting. Effectively, the procedure automatically endeavours to compute the correct result in what amounts to two successive 'stages'. In the first 'stage', when the fitting procedure begins, the models and objects are not well aligned, so most of the data can be viewed as outliers. Thus the second term is much larger than the first, thereby dominating the search. Obviously the exact size of the model has little effect on the second term. Hence, the actual search space mainly involves transformation and deformation parameters, as well as the ratio of the size parameters. Clearly, this search space will be smaller than the entire parameter space. As the fitting procedure progresses, the position, orientation and shape of the model will approach that of the object. Now the contribution of the second term gradually decreases and the first term becomes progressively 'larger'. When the value of the first term is similar to that of the second, the search enters the second 'stage' in which both terms will contribute equally to the objective function, and the search space becomes the full parameter space. Thus a search in full parameter space without good initial estimations is effectively achieved by a 'subspace' search followed by a full-space search with good initial estimates of transformation parameters, as shown in Figure 5.11.

Yokoya et al. have proposed a different two-term objective function [**153**], in which the first term is an L_2 norm. Accordingly, this term will contribute significantly to the objective function right from the beginning of the parameter search. Since this term depends on all of the model parameters, this method conducts a full-space search throughout the whole procedure. Therefore, their objective function is less efficient than ours.

6. Chapter Summary

This chapter describes an approach to qualitative volumetric shape representation by approximating object parts with minor shape variations by a finite set of primitives. We have proposed a new set of volumetric primitives, parametric geons, founded on the basic forms of sculpture and globally-deformed superellipsoids. Parametric geons provide distinctive qualitative shape classes as well as quantitative size and deformation information required for object recognition. The models impose constraints which facilitate shape approximation in the qualitative shape recovery process.

We have proposed an approach to recover parametric geon models from range data. An objective function involving a measure of distance and normal differences, and global

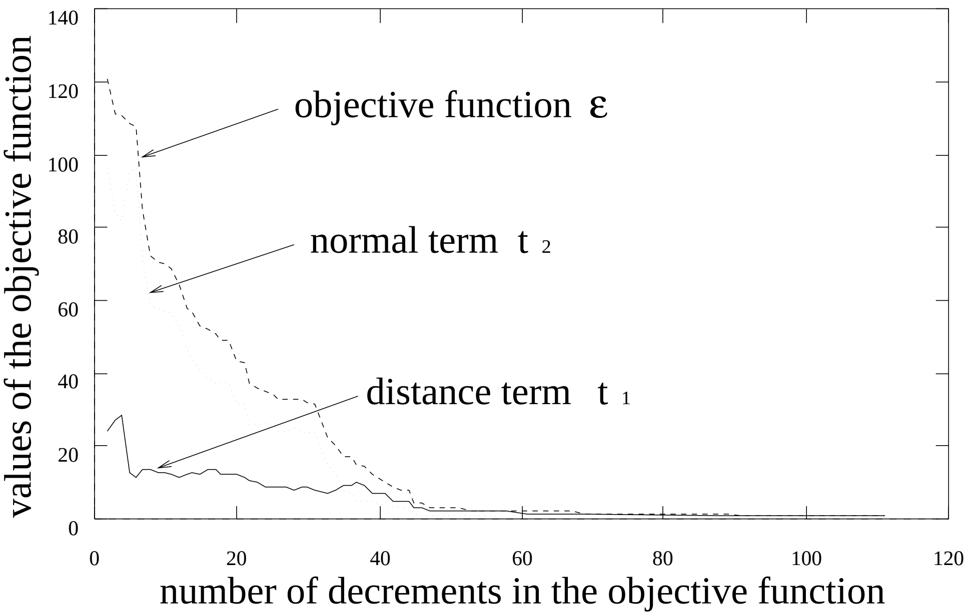


FIGURE 5.11. Values of distance measures (t_1) and normal measures ($\lambda\gamma t_1$) vs number of decrements in the objective function. The curves show the convergence of the distance measure t_1 and the normal measure $\lambda\gamma t_2$ as they change during the search. The solid line indicates values of the distance measure. The dotted line gives values for the normal measure. The dashed line presents the values of the complete objective function. These curves were obtained when a curved cylinder model was fit to data from the same type of object.

optimisation (VFSR) are all used to fit models to the data. The combination of the L_1 and L_2 norms in the objective function permits an efficient and hierarchical search of the model parameters resulting in more discriminative fitting residuals. The best model is selected based on the minimum fitting residual. The VFSR works very efficiently by allowing temperature decrease exponentially while the classical simulated annealing can only decrease temperature logarithmically to ensure the statistic global convergence. The experimental results will be reported in the next chapter.

CHAPTER 6

Experiments

In previous chapters we presented the motivation and techniques for computing part-based representations of 3D objects. In this chapter, we will provide experimental results to demonstrate the various aspects of the algorithms.

Section 1 describes the experimental setup, including the laser rangefinder, data acquisition and computer facilities. Section 2 explains the user-defined parameters required for our algorithms. Section 3 presents the results of the charge density computation. In order to illustrate some characteristics of the charge density distribution clearly, we also show its distribution over a 2D contour. Section 4 presents the results of object segmentation. Section 5 gives the experimental results of part identification. Different aspects of model recovery are investigated in this section. A chapter summary is provided in the last section.

1. Data Acquisition

Multiview and single-view range data were used throughout all of the experiments. Acquisition of the range images was accomplished using a NRC/McGill laser rangefinder which scanned objects supported by a turntable. The objects were placed from 30cm to 60cm from the rangefinder. Simple thresholding was used to remove the background data. This threshold was determined off-line by a calibration of the 3D workspace. For multiview range data, four views were obtained for each object. View transformation parameters were initially computed based on a calibration between the rangefinder coordinate system, and the turntable coordinate system and then refined by a method described in [**25**]. With these parameters, our method transformed the range data in each rangefinder coordinate system into a world coordinate system and redundant data which could be seen from more than one view were removed. The approach used for data transformation and redundant data deletion is described in Appendix B. Figure 6.1 shows the range images obtained from

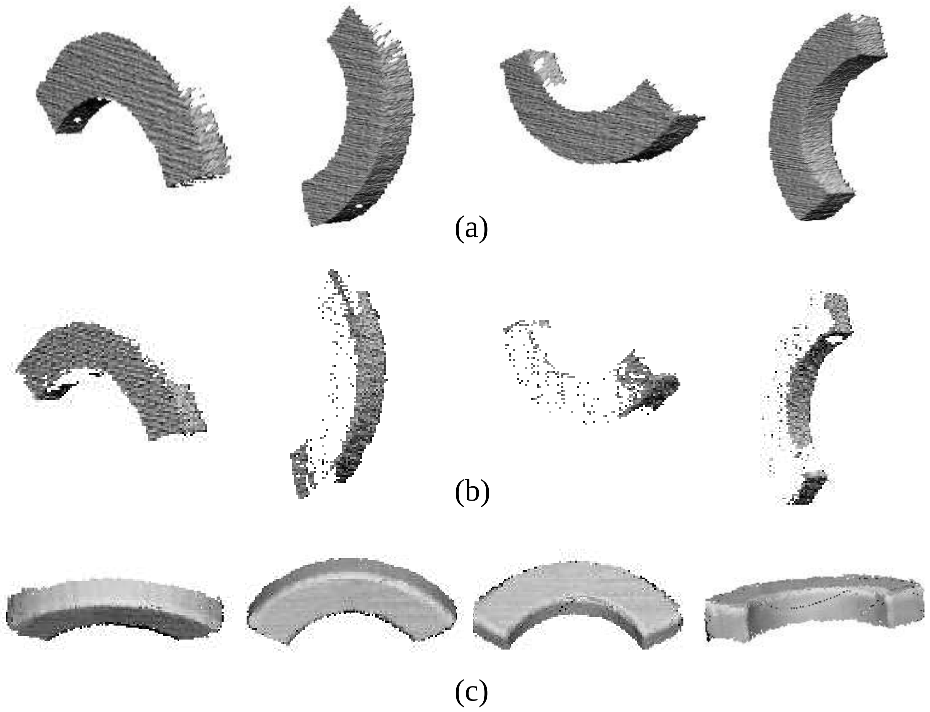


FIGURE 6.1. Multiview integration. (a) Four images of a curved wooden cuboid taken from different viewpoints. (b) Range data in each view after redundant data are removed. (c) Range data viewed in the four directions around a horizontal axis after merging the four data sets in (b).

four viewpoints, the range data after most redundant data are removed, and the range data integrated into the world coordinate system.

Objects used in experiments include machine-made wooden objects, carved stone objects, toy bowling pins and bananas. Beside the data we acquired in our laboratory, some data were also obtained from the GRASP Lab at the University Pennsylvania, the PRIP Lab at Michigan State University, and InnovMetric Software Inc.. All programs were written in C or C++ and were run on SPARC-10 or SGI R4000 and R8000 workstations.

2. Parameter Specifications

In this section, we list all user-defined parameters, some of which are data dependent and may not work in all circumstances.

(i) **Total charge** Q: $Q = 1000$. This parameter sets the total charge on a conducting object in Equation (4.12). Since we are interested in the simulated charge density, this parameter is data-independent.

(ii) **Separation between top and bottom patches**: 20% of the maximum range of data in the Z direction. This parameter, which is defined in Equation (C.2), sets the position of the bottom patch of a triangular mesh when triangulating single-view range data. The larger its value, the more triangles are created and the longer the charge density computation will be. Although changing the parameter affects the charge distribution on the visible surface, it has little effect on the ultimate position of the extrema of the charge density. Therefore, the segmentation results are not sensitive to this parameter.

(iii) **The charge density for determining a part boundary, ρ_t**: 120% of the minimum charge density over the object surface. This parameter, which is used in Algorithm 4.3, determines when an object should not be decomposed any further. Its selection depends on *a priori* knowledge of the surface concavity.

(iv) **Weighting factor λ in the objective function**: $\lambda = 5$. This parameter, which occurs in Equation (5.18), sets the contribution of the second term to the objective function. It depends on the similarities between the parametric geon shapes.

(v) **Maximum number of objective function evaluations in VFSR**: 10,000. This parameter sets the maximum number of objective function evaluations performed by the global optimisation algorithm, VFSR. The higher this number, the longer the algorithm runs and the more accurate the model parameters.

(vi) **Relative difference between two objective function values**: 0.003. This parameter is one of the stopping conditions for VFSR (see Section 4.3). It is defined as

$$\frac{ObjectiveFunction(i-STEP) - ObjectiveFunction(i)}{ObjectiveFunction(i-STEP)}$$

Here i is the decrement index of the objective function. $STEP$ is the number of times an objective function value is lower than all previous values, as shown in Figure 6.2. We set $STEP = 10$. This parameter reflects the coarse slope of the objective function. The larger its value, the less time the fitting procedure will take, the less accurate the model parameters.

3. Charge Density Distribution

The results of the charge density distribution are presented in this section. In order to explain the behaviour of charge density distributions in a convenient way we first discuss the algorithm by computing the charge density distribution for 2D objects.

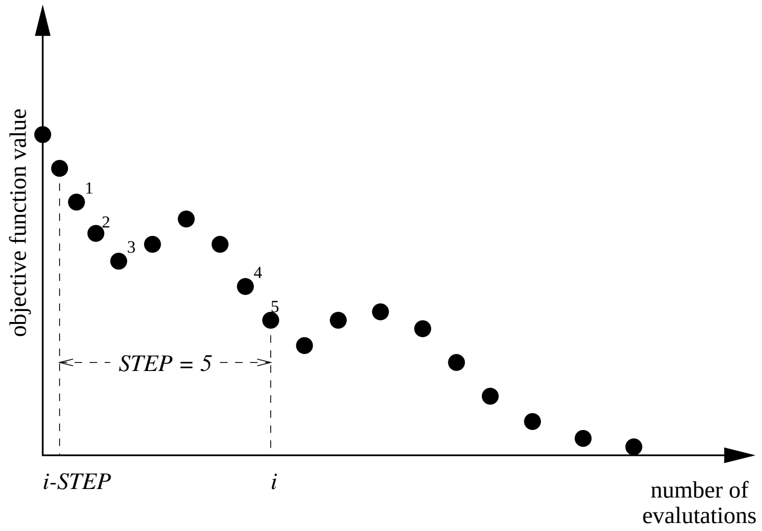


FIGURE 6.2. The decrement ($STEP$) of the objective function. The black dots represent the samples of the objective function. The figures beside the dots indicate the number of the decrements starting at the $(i - STEP)$th evaluation.

3.1. 2D Case

3.1.1. *Method* The algorithm for a 2D shape is slightly different from the 3D case. The physical model we have used is the charge density distribution on a 2D contour. To do this, we examine an *infinitely long* charged conductor having this 2D contour as a *constant* cross-section, as shown in Figure 6.3. Since the cross-sections have constant size and are parallel to each other, the charge density distribution is the same at any plane along the infinitely long conductor. The charge on the contour is treated as line of charge, that is, a uniformly charged infinitely long line perpendicular to the cross section. We have developed the algorithm for computing the 2D charge density distribution [**150**].

Since a contour in an image is composed of a sequence of pixel points, a polygonal approximation can be formed automatically by linking all consecutive pixels on the contour by line segments. This is different from the 3D case where a triangular mesh is constructed. The middle point of each segment is selected as the observation point. The actual charge density on each contour pixel is computed by taking the average of the charge densities at the observation points, on each side of this pixel along the contour.

3.1.2. *Using Synthetic Data* First, we tested our algorithm using a perfect ellipse to verify its correctness:

$$r(\theta) = d/(\cos^2\theta + d^2\sin^2\theta)^{1/2} \tag{6.1}$$

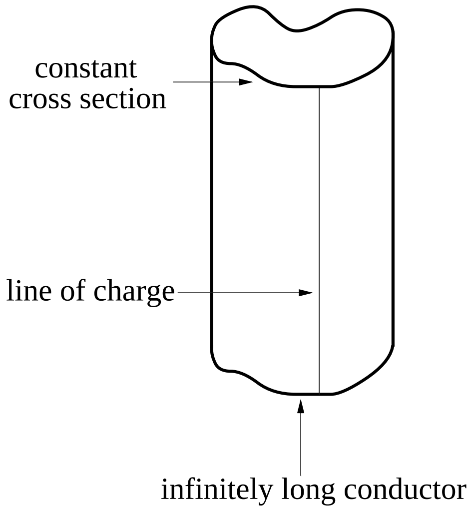


FIGURE 6.3. Configuration of charged 2D contours. The 2D charge density distribution is computed along a cross section of the infinitely long conductor. The thin vertical line indicates the line of charge.

Here d is the ratio of the major to minor axes of the ellipse. The discrete data are generated by sampling the variable $\theta \in [0, 2\pi]$ at N points. Here $N = 360$. We compared the result to the analytical expression for the charge density along an ellipse, known to be proportional to [**111**]:

$$\lambda(\theta) = \left(\frac{\cos^2\theta + d^2 \sin^2\theta}{\cos^2\theta + d^4 \sin^2\theta} \right)^{1/2} (2\pi)^{-1} \tag{6.2}$$

Figure 6.4 contrasts the computed (solid curve) and theoretical charge density distributions (dashed curve). It can be seen that the two curves are very similar. We also note that there exist two other algorithms in the physics literature for 2D charge density computation [**108, 111**]. Since both have ignored the potential produced at the observation point, we have found them not to be as accurate as the method proposed here. In addition, we speed up the computation by evaluating the line integral[1] analytically while they do this numerically.

Next we show results from two other contours of analytical functions, the hypocycloid and the generalised epicycloid, containing either sharp convexities or concavities. Figure 6.5 shows that the charge density is very high at the sharp convexities and is close to 0 at the sharp concavities. Figure 6.5 also reveals that the charge density curve is sharper at convexities than concavities. This suggests that the charge density is more sensitive to the convexities than concavities.

[1]In the 2D case, the integration domain is a line segment rather than a planar triangle.

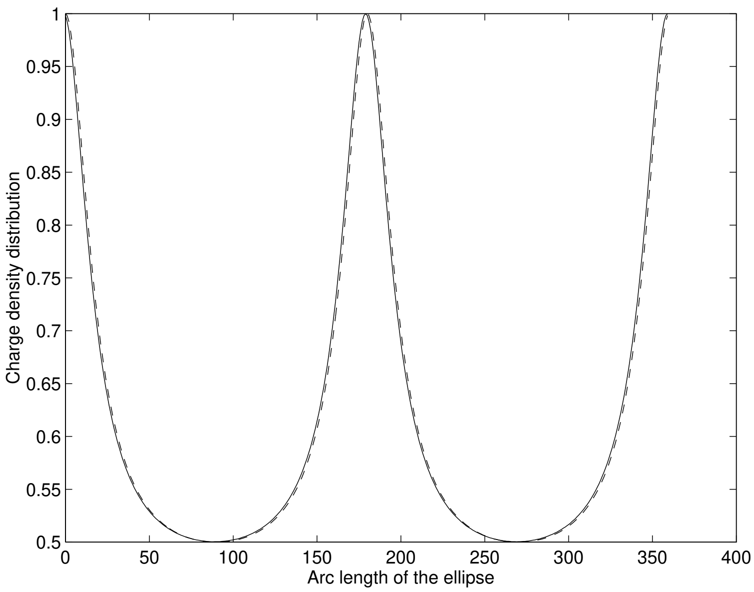
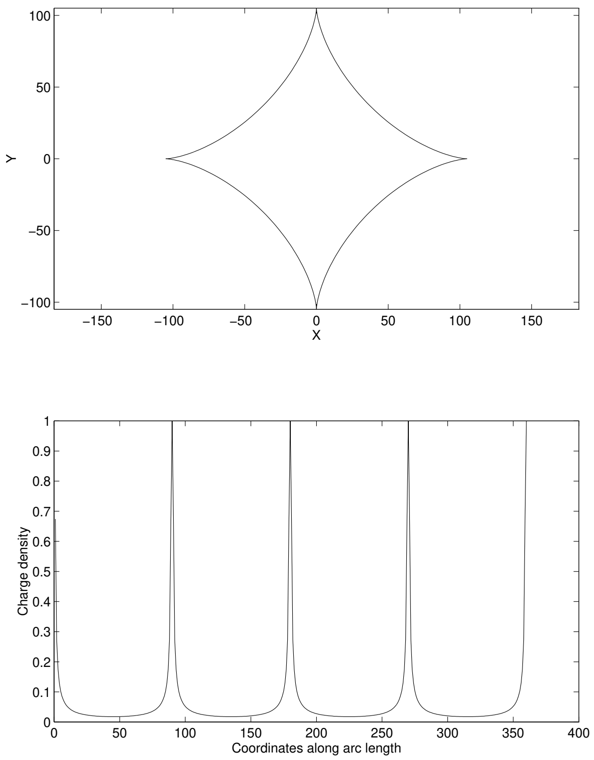


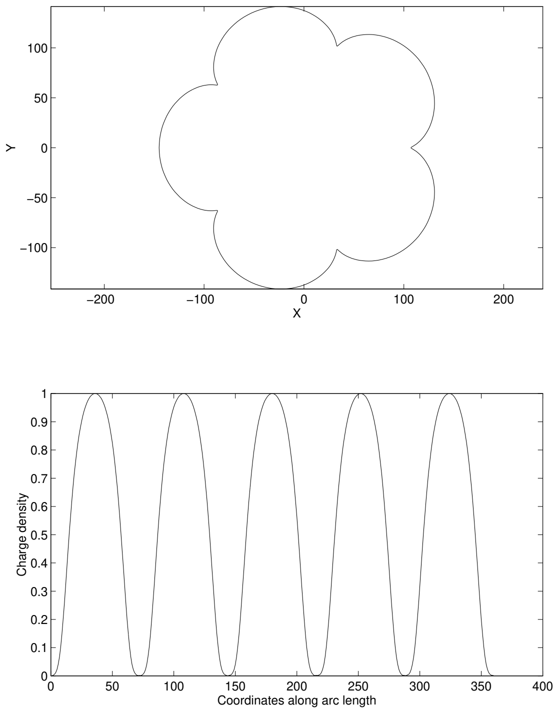
FIGURE 6.4. The theoretical (dashed line) and computed (solid line) charge density distributions along the contour of an ellipse. Both curves are normalised by their maximum values.

3.1.3. *Using Real Data* For data obtained from a real image, the actual contour is distorted by the image sampling process. Therefore, the high frequency noise contaminates the contour (see Figure 6.6 (a)) and also the computed charge density distribution (see Figure 6.6 (b)). However, we note that the noise affects the computation of the charge density distribution much less than it would a curvature computation.

Let us examine this hypothesis. In a similar fashion to computing the incremental curvature [**50**], we approximate the curvature of a contour based on the changing rate of the discrete tangent at a point on the contour. The increment is 1. A comparison of noise sensitivity between the charge density and the curvature for the object contour in Figure 6.6 (a) is given in Figure 6.7. We show the charge density distribution in the left column and the curvature distribution in the right column. Without any smoothing operation, all corners on the contour can be indicated by the charge density distribution (see Figure 6.7 (a)) but the concave corners are poorly indicated by the curvature distribution (see Figure 6.7 (f)). Next we applied lowpass filtering to the discrete Fourier transform of the polygon data to remove the high frequency components. The amount of smoothing was increased from 1% to 4% energy from the largest component. The results are shown in the rest of figures. It is clearly seen that the charge density computation is more robust to noise than the curvature computation.

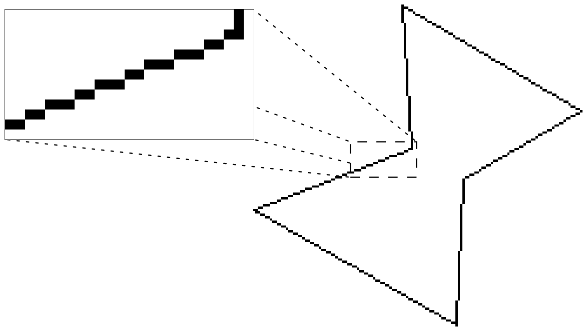


(a)

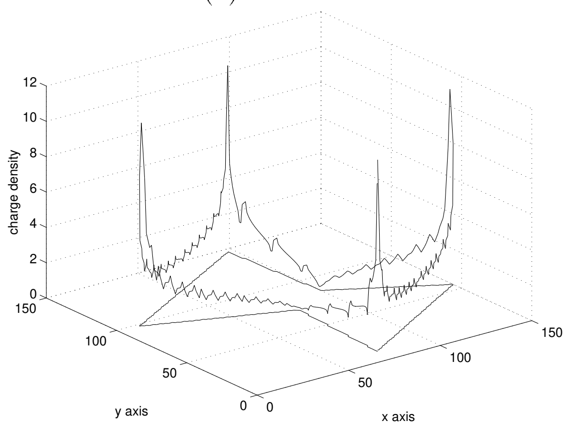


(b)

FIGURE 6.5. The charge density distribution along analytical contours. The arc length is referenced to the right most point. (a) A contour with sharp convexities and the charge density distribution along the contour. (b) A contour with sharp concavities and the charge density distribution along the contour. It can be seen that the charge density curve is sharper at convexities than concavities.



(a)



(b)

FIGURE 6.6. Effect of a noisy contour. (a) The smooth contour in an image is distorted by the image sampling process. (b) The charge density distribution along the contour given in (a).

In distinction to a local shape computation, such as curvature, the significance of the charge density distribution is its ability to reveal both fine and gross shape information. We demonstrate this with the following two examples. Figure 6.8 (a) illustrates a dumbbell-like object with wiggles superimposed. The gray levels indicate charge densities, which are normalised to the range between 20 (darkest intensity) and 255 (white). The object contains two kinds of structures. They are: (1) the fine structure, which is represented by small wiggles and (2) the gross structure, which is delineated by the two major components of the dumbbell. Figure 6.8 (b) shows the charge density distribution along the arc length of this object. This curve simultaneously indicates the fine and gross structures of the contour. The dashed line depicts the two gross components defined by the envelop of the charge density distribution. However, the incremental curvature of the contour only denotes the fine structure, as shown in Figure 6.8 (c).

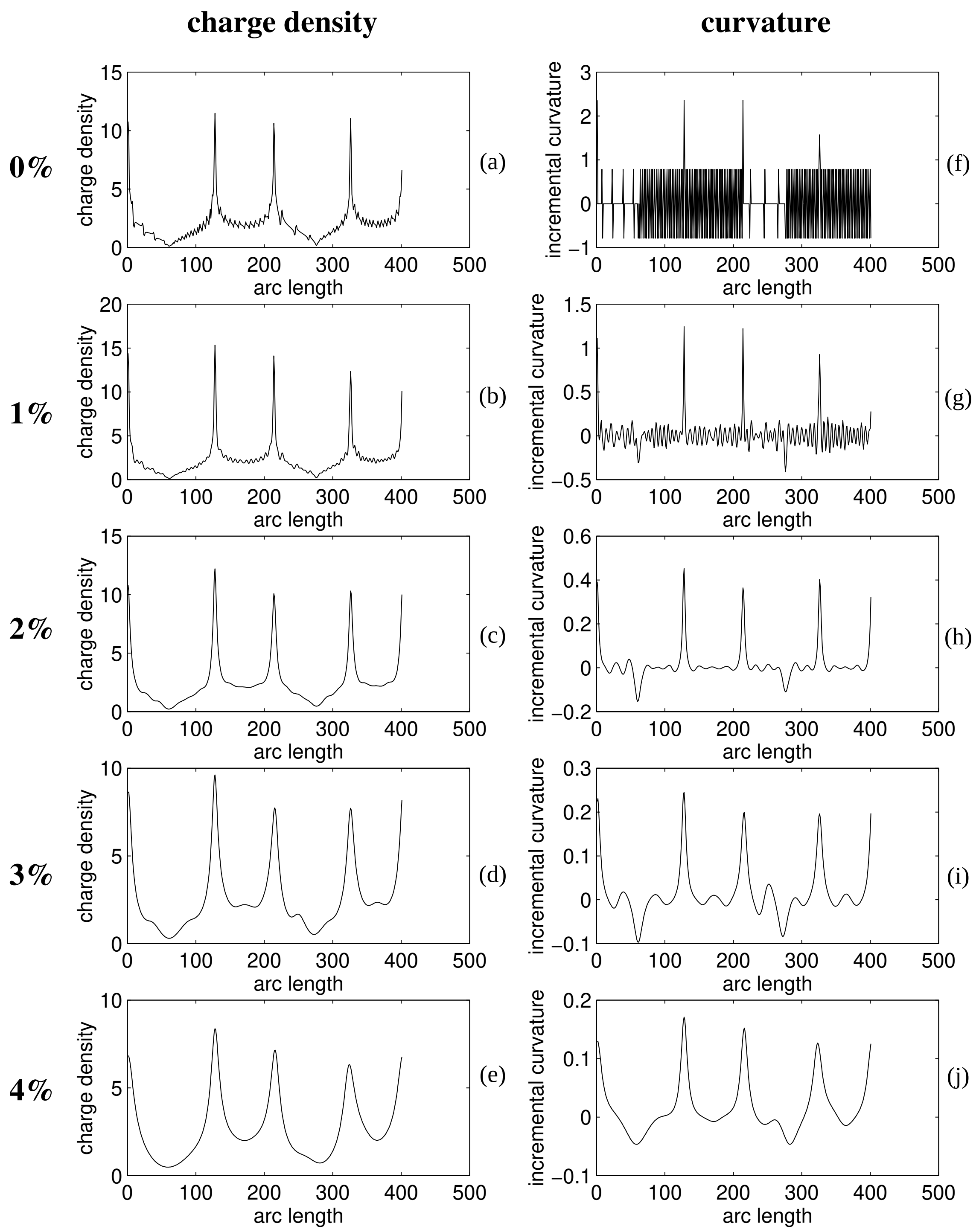


FIGURE 6.7. Comparison between charge density and curvature computations. The left and right columns show the charge density and curvature distributions, respectively. In the first row, no smoothing is applied. In the rest of rows, we have applied lowpass filtering to the Fourier components of the contour by removing from 1% to 4% of the energy in the largest Fourier component. At each level of smoothing, the charge density computation is more robust to high frequency noise than the curvature computation.

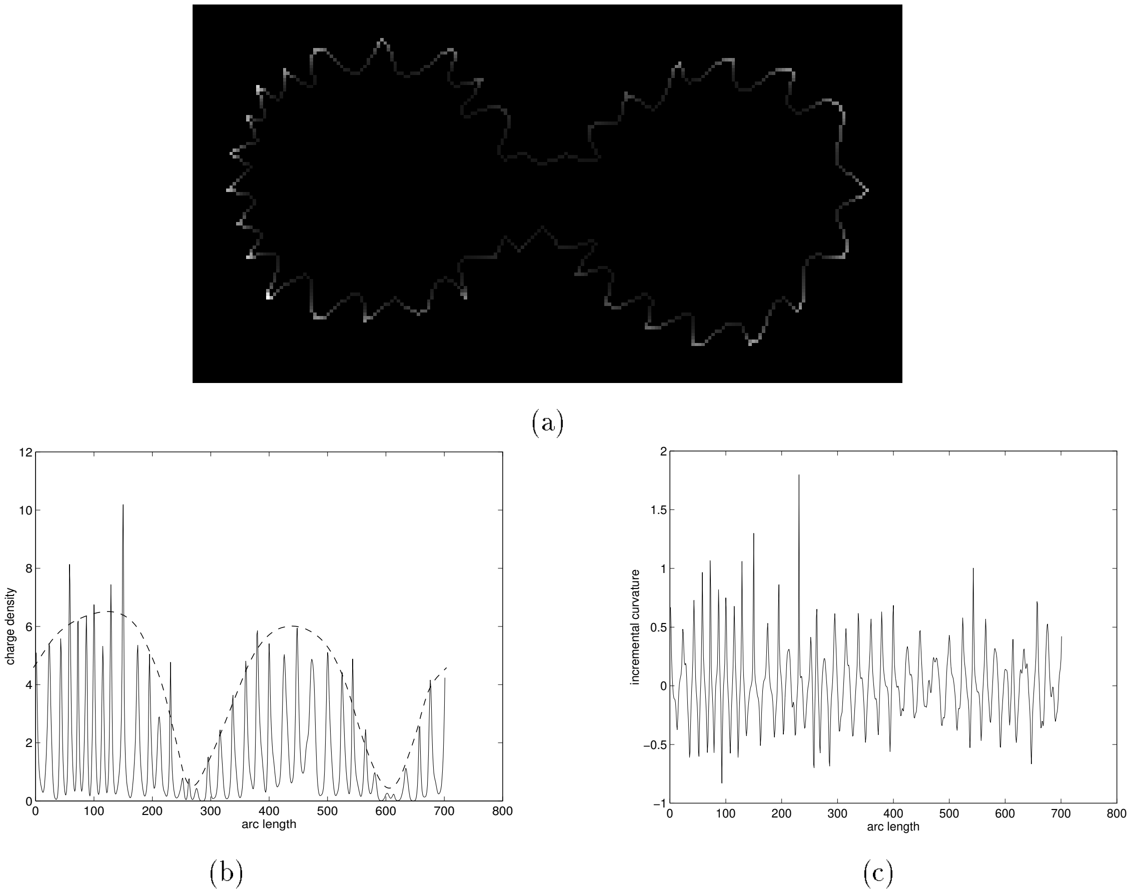


FIGURE 6.8. Fine and gross features. (a) The charge density for a dumbbell object with wiggles superimposed. The brightest and darkest intensities indicate the maximum and minimum charge densities, respectively. (b) The charge density distribution over the contour in (a). The arc length is referenced at the highest pixel on the contour and goes counterclockwise. The frequent peaks indicate small wiggles on the contour. The two peaks of the envelope (the dashed line) of the curve denote the two major parts of the dumbbell. (c) The incremental curvature distribution along the contour in (a). In this computation, the smoothing factor was chosen to be 2% and the increment for the curvature computation was equal to 1.

In another example, Figure 6.9 shows (a) the charge density on the object contour and (b) the charge density distribution along the arc length. We observe that the peaks and valleys in the charge density distribution can indicate not only the convexity and concavity of the contour shape but also the *significance* of protrusive parts. For example, the higher the charge density, the larger the part protrusion. We note that these are consistent with human intuition. However, the incremental curvature is not able to do so, as shown in Figure 6.9 (c).

To summarise, we have demonstrated in the 2D case that the charge density computation is less sensitive to high frequency noise than the curvature computation. Therefore,

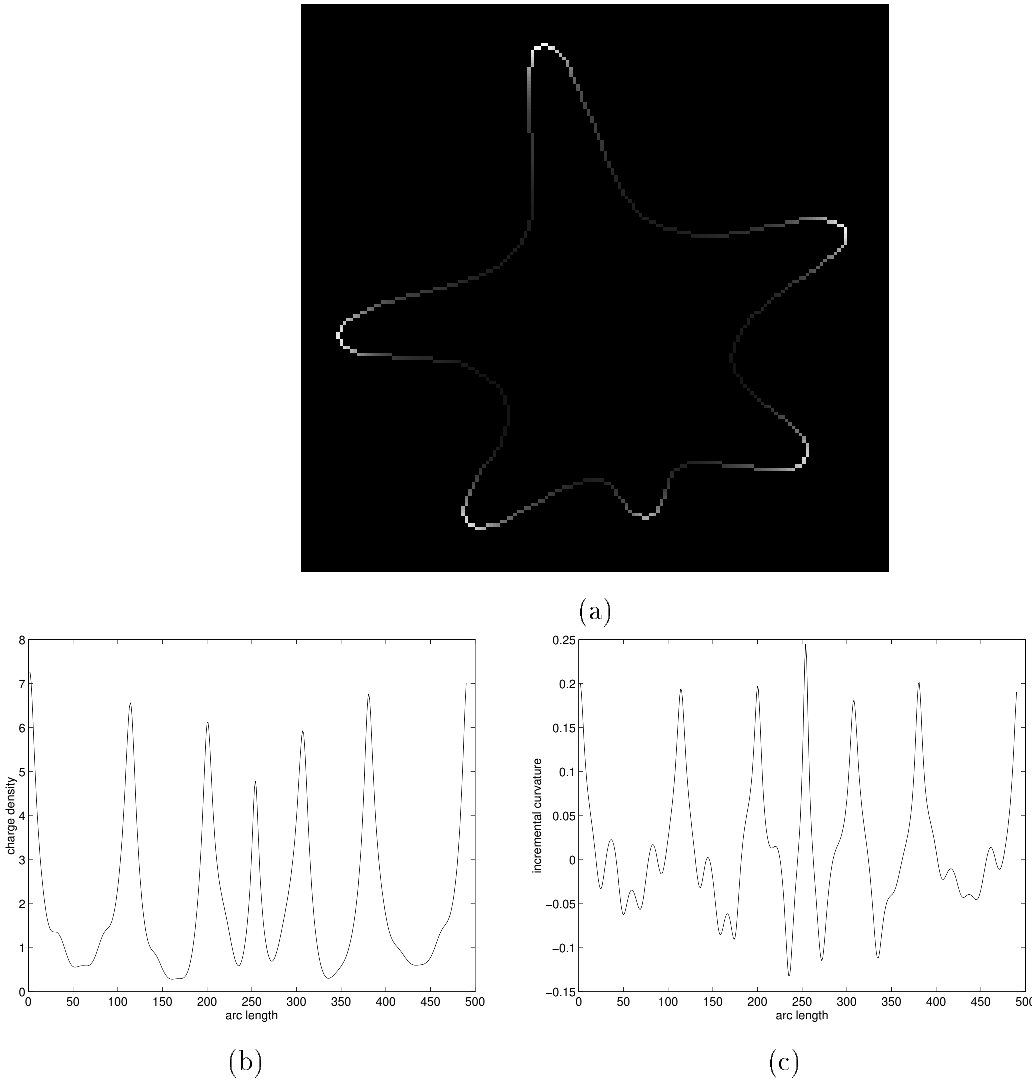
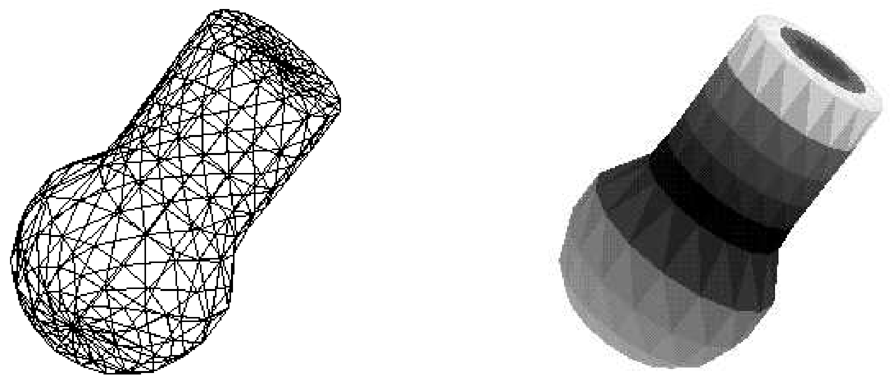


FIGURE 6.9. The charge density distribution on an image contour. (a) an object contour superimposed by the charge density distribution. (b) The charge density distribution along the arclength of the contour. The height of peaks indicate the significances of object parts. (c) The incremental curvature along the contour. In the computation, the smoothing factor is 2% and the increment for the curvature computation is 1.

detection of part boundaries using the charge density distribution is more robust than using curvature. Moreover, the charge density can reveal shape information at both fine and gross scales. It can also indicate the significance of parts. These features have not been demonstrated by any previous approach.

3.2. 3D Case In this section, we will show experimental results of the charge density computation for surfaces of 3D objects. The first object is a vase, consisting of sphere and a cylinder. The raw range image (sphere+cylinder.Z) originated from the PRIP Lab at



(a) (b)

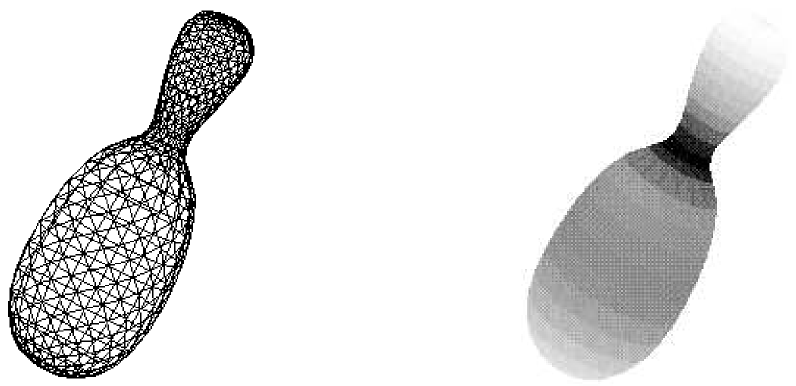
FIGURE 6.10. Charge density for a vase. (a) The triangular mesh tessellation; (b) the computed charge density distribution.

Michigan State University. The mesh data for the object, shown in Figure 6.10 (a), were obtained by a deformable model fitting algorithm developed by D. Decarlo and D. Metaxas at the University of Pennsylvania [**35**]. Figure 6.10 (b) gives the computed charge density distribution over the object surface. The gray levels indicate charge densities, which are normalised to the range between 0 (darkest intensity) and 250 (white). It can be clearly seen that the lowest charge densities are located at surface concavities, which are at the intersection of the spherical and cylindrical portions of the object. Conversely, since the edge on the top of the object is sharply convex, the charge density at these points reaches a maximum.

The second 3D object is a toy bowling pin. The range data were obtained by multiview integration, as described in Appendix B. The triangular mesh of the object in Figure 6.11 (a) was computed in the same way as the previous object. Figure 6.11 (b) shows the simulated charge density distribution. Again the charge density easily distinguishes between the locations of the deep surface concavities and convexities.

The third 3D object is a toy elephant. Multiview data were obtained from InnovMetric Software Inc. who also computed the tessellation [**128**], shown in Figure 6.12 (a). Figure 6.12 (b) shows the computed charge density distribution. Although the shape of the elephant is very complex, our method successfully indicates, by the bright gray levels, the protrusions caused by the legs, nose, ears and teeth of the elephant. The charge density also reveals the concave portions around the neck and between the legs (as dark gray levels).

So far we have illustrated the charge density distributions using triangular meshes, which model the complete 3D shape of the objects. Next we will show experimental results



(a) (b)

FIGURE 6.11. Charge density for the bowling pin. (a) The triangular mesh tessellation. (b) The computed charge density distribution.

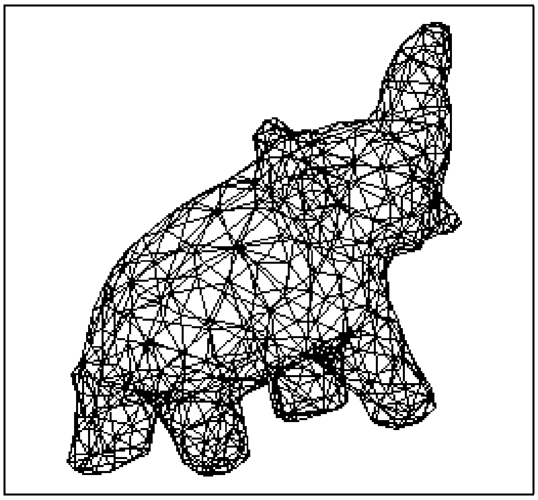
obtained from single-view range data. In this case, since only partial shape information is available, we construct a closed triangular mesh for the purpose of computation.

Figure 6.13 (a) shows the triangular mesh of the visible surface of a carved stone owl and (b) shows the computed charge density distribution over its surface. Figure 6.14 (a) illustrates the triangular mesh of the visible surface of a clock with two ringers on the top and (b) shows the computed charge density distribution over this surface. In both examples, dark and bright regions indicate surface concavity and convexity, respectively. Although only partial shape information of the complete objects are available in these experiments and the construction of the closed triangular meshes is rather arbitrary, our algorithm can still produce the desired results for the visible surfaces. The shape of the invisible portions of the object do actually affect the absolute values of the charge density distribution on the visible surfaces. However, the relative values of the high and low charge densities almost remain the same. We note that the size of triangles is not crucial to the charge density computation. During the experiments, we observed that even with a ratio of maximum to minimum triangle areas of about 200, our algorithm still produced satisfactory results.

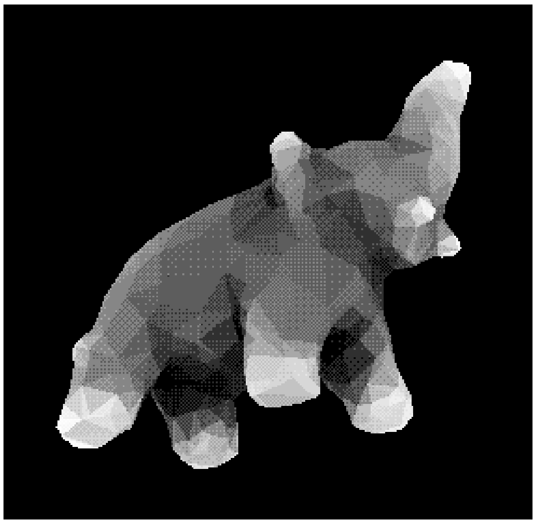
For the bowling pin, which consists of 864 triangles, the charge density computation takes 80 seconds on a SGI-R8000 workstation. The complexity of this computation of is $O(N^2)$. N is the number of triangles in the triangular mesh.

4. Object Decomposition

Given that the charge density distributions shown in Section 3.2, our algorithm decomposes an object into parts by locating and then deleting triangles on part boundaries. Examples are given in Figure 6.15. The bowling pin and the vase are segmented into two



(a)

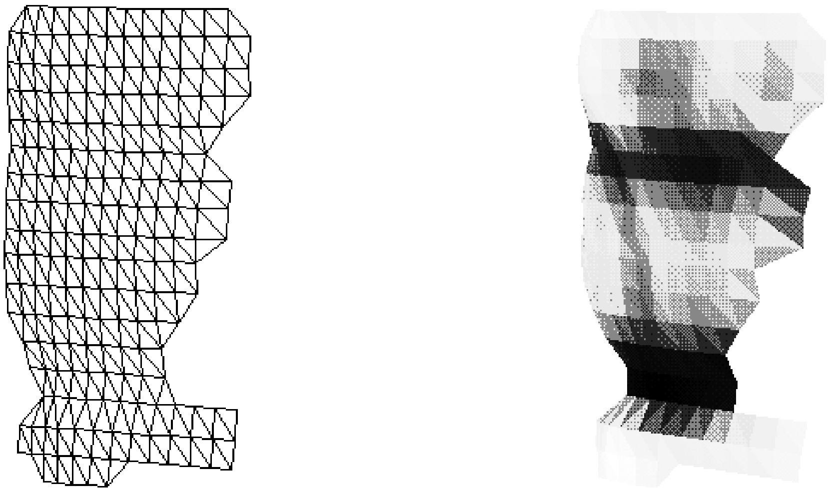


(b)

FIGURE 6.12. A toy elephant. (a) The triangular tessellation. (b) The charge density distribution.

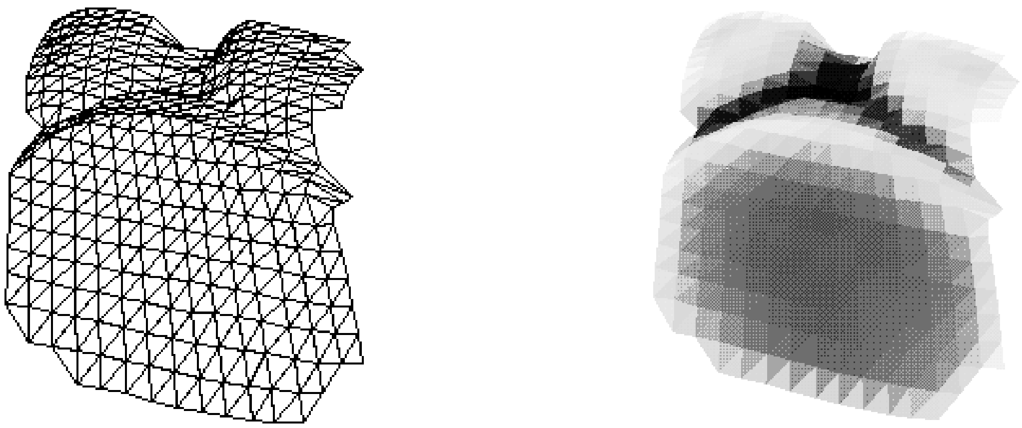
parts. The owl and the clock were segmented into three parts. These results are consistent with our intuition of object parts.

We note two issues regarding part segmentation. The first concerns the criterion for part decomposition. We have claimed that low charge density indicates part boundary points. The question is: How low should the charge density be to indicate a part boundary? In fact, it appears that, for human beings, there is no universal criterion for determining parts. It always depends on personal experience and preferences. Thus, it is also reasonable for users to set a threshold for the value of the low charge density. In order to determine a



(a) (b)

FIGURE 6.13. An owl sensed in a single view. (a) The triangular tessellation on the visible surface. (b) The charge density distribution.



(a) (b)

FIGURE 6.14. An alarm clock with two bells on top sensed in a single frontal view. (a) The triangular tessellation for the visible surface. (b) The charge density distribution.

threshold, one must know the absolute or relative depth of a concavity that forms a part boundary. We know that the charge density is not a pure local shape measure and therefore cannot provide absolute information. Hence, this threshold must be determined by relative information, originating from *a priori* knowledge of the surface concavity.

The second issue is related to the assumption about object shape. Although the transversality principle states that when two objects interpenetrate they intersect transversally with probability one [**11**], a surface concavity does not always appear for all objects.

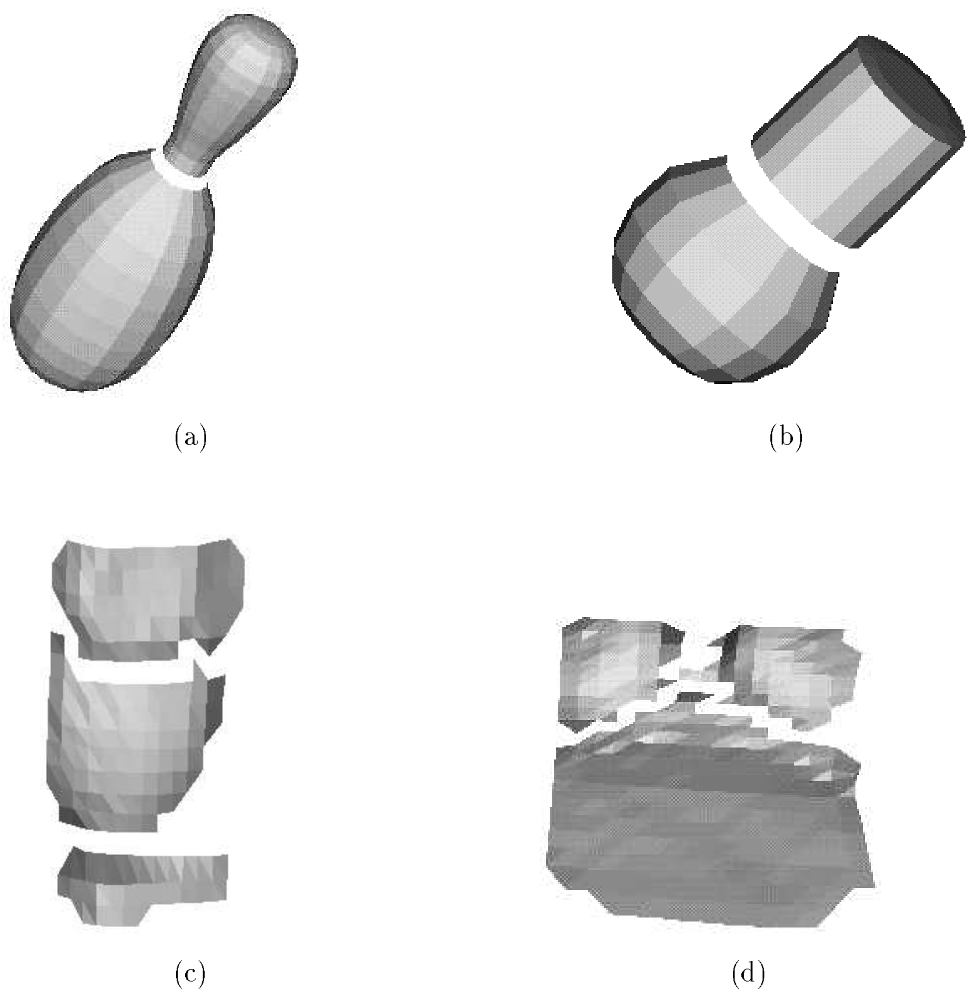
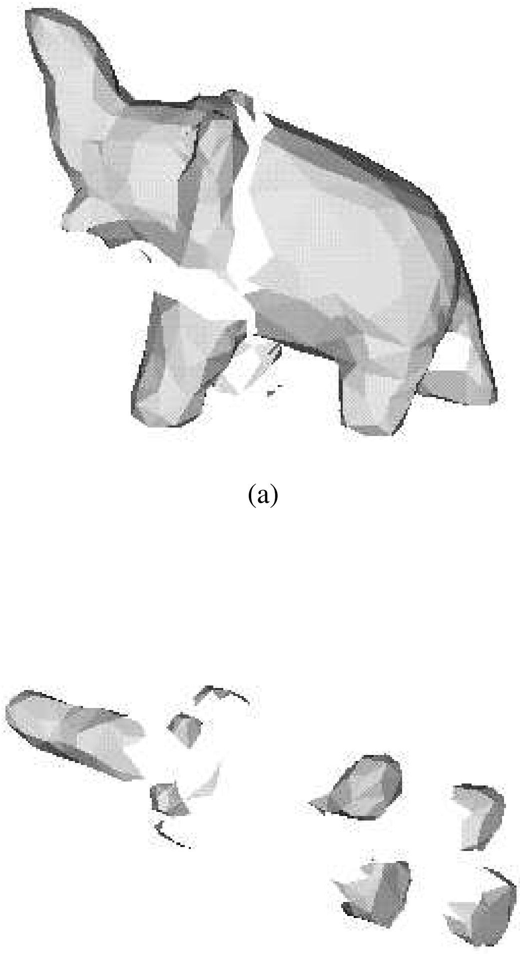


FIGURE 6.15. Results of part segmentation. (a) and (b) show the segmented parts of objects tessellated by a model fitting technique, which uses multiview data. (c) and (d) show the segmented parts of objects tessellated by a local triangulation technique, which takes single-view range data.

We stated in Chapter 3 that the part boundaries of objects segmented by this method must be delimited by a closed contour of surface concave points. This ensures that a part can be completely separated from the rest of the object. Here we show an example where the part segmentation assumption is violated. Figure 6.16 (a) illustrates the segmentation of the elephant. Although one can see some intuitive parts, such as nose, four legs and ears, etc., only the left front leg is separated from the object by our segmentation method. This occurs because that the part boundaries are not delineated by a closed contour of deep surface concave points.

Nevertheless, using a different decomposition strategy may produce a better segmentation. Figure 6.16 (b) shows the result obtained by simple thresholding. This method



(a)

(b)

FIGURE 6.16. The segmented result for the elephant. (a) shows the segmentation obtained by the part boundary tracing method. (b) shows the segmentation obtained by thresholding.

extracts these triangles that are located on protrusive parts. The charge density values on these triangles are greater than the selected threshold. We can see that all protrusive parts have been found, although they are not complete. This example again shows that the charge density is a good indicator for protrusive parts.

5. Part Identification

In the previous section, we presented the experimental results for object segmentation into parts. The next stage of our system derives a parametric geon model for the segmented part. The task is performed by fitting all models to the part data and selecting the best model based on the minimum fitting residual. In this section, we will discuss some experimental results and investigate the following issues:

(i) The efficiency of the objective function for model fitting

(ii) The discriminative properties of parametric geons

(iii) The effect of object shape imperfection

(iv) The importance of multiview data for shape approximation

We are interested in examining the residual differences for all of the models, especially when object data contain noise and object shapes do not exactly conform to the parametric geons. The inputs in the experiments are single-view and multiview 3D data of single-part objects and segmented parts. Part model recovery requires the data points and surface normals. Since each segmented part is represented in terms of a set of triangles, we use the centroids and orientations of all triangles as the data points and normals, respectively.

The execution time varied according to the data, models and stopping conditions. The approximate average time taken for obtaining acceptable fitting results was around 3 minutes on an SGI(Personal Iris) R4000 workstation. However, to achieve very accurate model parameters, the computation could require about two hours for most complex parametric geon shapes, i.e. tapered and curved models. The relative error of the objective function residuals produced is less than 5%.

5.1. Using Range Data of Geon-like Objects In this experiment, we matched each parametric geon model to multiview range data of seven machine-made wooden objects and examined the fitting residuals. Since they are single-part objects, no segmentation is needed. The shape of each object was similar to one of the parametric geons. Four views were used to collect the range images. Surface normals were computed by a least squares fitting method. After multiview integration, these dense 3D data were subsampled at a $50:1$ sampling rate for parametric geon recovery.

Table 6.1 shows that the residuals obtained by fitting models to their own object type are much smaller than those obtained by fitting to other object types. Thus the seven selected parametric geons are seen to be very discriminative. The types of objects used are listed in the first column. The bold figures on the diagonal are the residuals given by fitting a model to its own type of data. The underlined figures are residuals produced by fitting tapered and curved models to a cylinder or a cuboid. When this is done, k_x, k_y, or κ take values which are very close to 0. Thus, the data from the deformed models can be very nicely characterised by the non-deformed models (regular cylinder or cuboid). If two residuals are very close and much smaller than the others, the algorithm arbitrarily selects

OBJECTS	MODELS						
	1	2	3	4	5	6	7
1	**1.206**	12.075	19.368	16.511	26.449	14.512	24.162
2	22.535	**0.7968**	12.976	0.864	12.834	0.871	14.432
3	36.156	17.819	**1.313**	28.707	1.327	17.807	1.338
4	17.993	20.740	27.190	**2.339**	15.203	17.625	25.740
5	34.276	28.241	14.256	20.156	**1.667**	28.228	14.242
6	24.153	21.986	22.148	21.197	47.987	**3.300**	22.197
7	34.209	29.887	23.213	25.130	16.291	14.341	**2.949**

TABLE 6.1. Fitting models to range data of geon-like objects. Items in each row are the residuals from fitting different models to the data of a particular object, as listed in the first column. The numbers from 1 to 7 denote ellipsoid, cylinder, cuboid, tapered cylinder, tapered cuboid, curved cylinder and curved cuboid, respectively. The bold figures denote the residuals from fitting a model to its own object type. The underlined figures are the residuals from fitting tapered or curved models to a regular cylinder or cuboid.

the simplest of the two shapes. Figure 6.17 shows the results of fitting the seven parametric geons to the range data of a curved cuboid. The lighter shaded volumes are the models obtained by the fitting procedure and the darker sparse spots indicate the input data. (a) through (g) illustrate models of the ellipsoid, the cylinder, the cuboid, the tapered cylinder, the tapered cuboid, the curved cylinder and the curved cuboid superimposed on the 3D data, respectively. We indicate the residuals at the top left corner in each image. The algorithm selected the curved cuboid shown in (g) as the best model for the wooden object. This result is consistent with our expectations.

5.2. Using Range Data of Imperfect Geon-like Objects The purpose of this experiment was to examine the uniqueness of shape approximations using parametric geons when given multiview data of a set of single-part objects whose shapes varied. In this case, eleven real bananas were taken as the objects. Figure 6.18 shows four of the bananas used in the experiments. Their shapes cannot be exactly depicted by any of the parametric geons. The apparently noisy surfaces of the bananas shown in the figure were due to the rangefinder's sampling error. This was because the bananas had to be placed relatively far from the rangefinder in order for them to fit within its scanning field-of-view.

Figure 6.19 shows the results of fitting the seven parametric geons to the 3D data of a particular banana. The algorithm selected the curved cylinder shown in (f) as the best model. The numbers at the top left corner in each image indicate the fitting residuals.

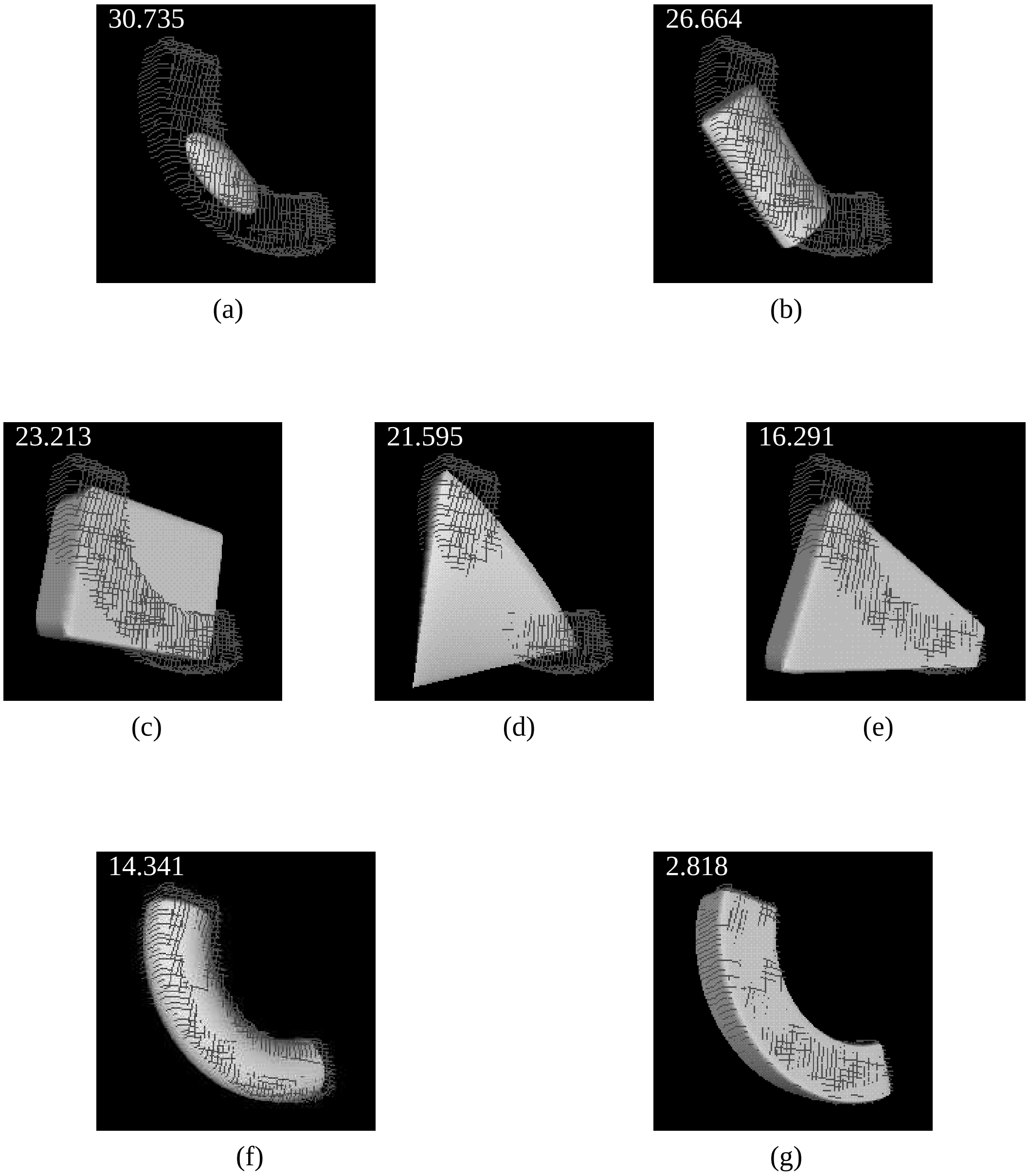


FIGURE 6.17. Fitted models superimposed on range data of a curved cuboid. As expected the curved cuboid in (g) yielded the minimum fitting residual. The values of fitting residuals are indicated at the top left corner in each image.

Clearly this result is consistent with our intuition of the banana's actual shape. Table 6.2 gives the average, maximum and minimum fitting residuals for all of the bananas. Since these are all of different size, we cannot make an absolute comparison of the fitting residuals. Thus, each residual was normalised by the minimum residual among those obtained for the same banana. The results show that the best model for all of the bananas is the curved cylinder, which gives the smallest average residual value. The results of parametric

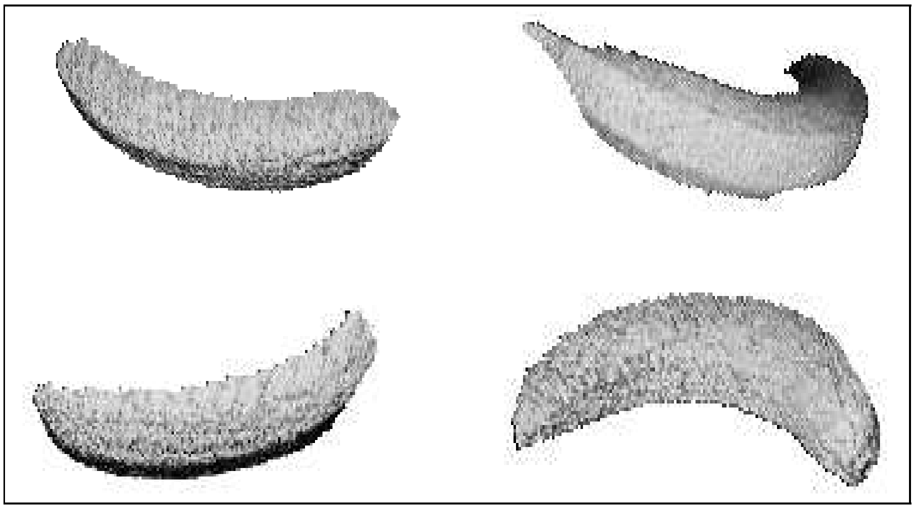


FIGURE 6.18. Four bananas used in the experiments.

	MODELS						
	1	2	3	4	5	6	7
Mean residual	3.255	2.889	3.851	3.324	3.611	1.000	2.987
Maximum residual	4.001	3.489	4.717	5.018	4.328	1.000	3.802
Minimum residual	2.656	2.458	3.102	2.464	3.073	1.000	2.385

TABLE 6.2. Results of fitting different models to range data of eleven bananas. The models are numbered in the same way as Table 6.1.

geon fitting can be also appreciated from Figure 6.20. The circles linked by dotted lines correspond to residuals from one particular set of data. In order to show the differences in the residuals clearly, they were normalised by the minimum residuals obtained for each banana. One can see that all minimum residuals are significantly lower than others. Therefore, the parametric geon models and recovery procedure demonstrate robust behaviour and uniquely represent the different banana shapes.

5.3. Comparing Different Objective Functions In this experiment, we recovered parametric geons from the same eleven bananas using just t_1 (see Equation (5.19) in Chapter 5) as the objective function for fitting. This objective function measures the sum of

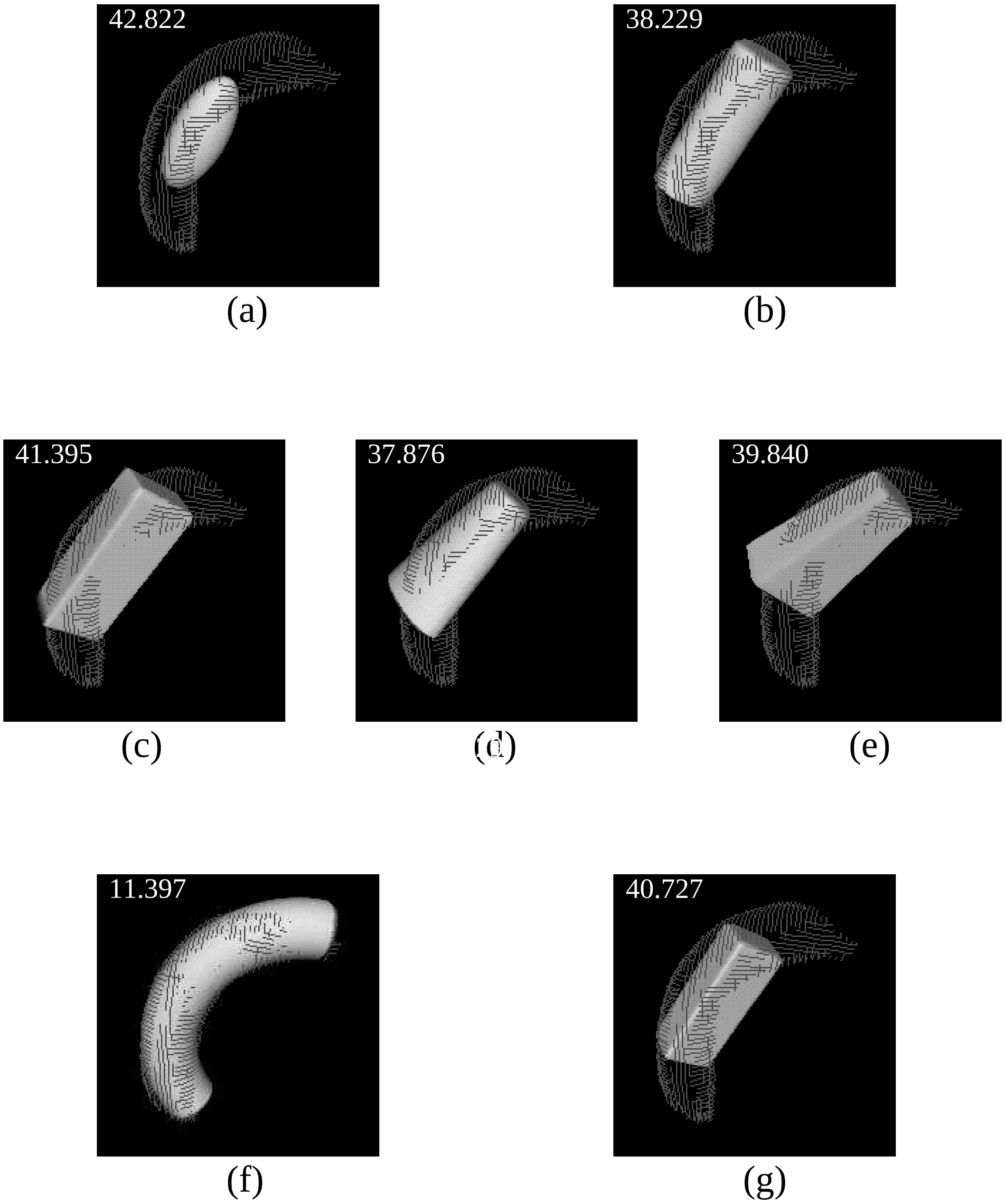


FIGURE 6.19. Fitted models superimposed on the range data obtained from a banana.

the spatial distances from the data points to the model surface along a line passing through a datum and the model centre. It has been shown to have a significant advantage over some others as an objective function for superellipsoid fitting [**55**]. Although our 3D data were obtained from multiple views, the data at the bottom of the bananas were still missing, a

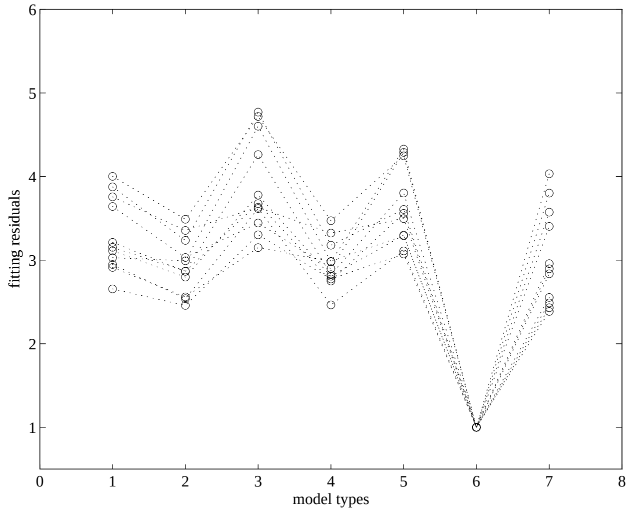


FIGURE 6.20. Fitting residuals obtained using eleven bananas. The horizontal axis represents each individual parametric geon model (1-ellipsoid, 2-cylinder, 3-cuboid, 4-tapered cylinder, 5-tapered cuboid, 6-curved cylinder, 7-curved cuboid). Circles connected by a dotted line correspond to residuals obtained from one set of data. All residuals are normalised by the minimum residuals. The objective function in Equation(5.18) was used.

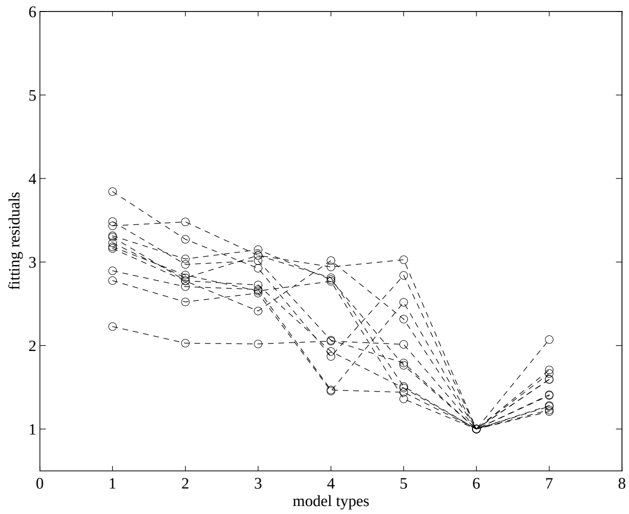


FIGURE 6.21. Fitting residuals obtained using a different objective function with eleven bananas. All residuals are normalised by the minimum residuals. In the objective function, only weighted t_1 is used.

situation which would cause the model size to be underconstrained. Thus we multiply t_1 by a size factor γ. The effect is same as that of γ presented in Section 3.2, Chapter 5.

Figure 6.21 shows the fitting residuals obtained using the weighted t_1 with eleven bananas. Although this simplified objective function actually produces the correct shape type, the minimum residuals are not significantly lower than the others. In addition, for some data sets the algorithm required many more evaluations to find the global minimum than the objective function proposed in this paper. The latter shows superior performance because the component due to the normal measure makes the global minimum deeper, as indicated in Figure 6.20). This facilitates global optimisation and makes the fitting residuals more discriminating.

5.4. Comparing Single-view and Multiview Data This experiment examined the quality of fitting when using single-view data of the same eleven bananas. Figure 6.22 shows the normalised fitting residuals. The algorithm again selected the curved cylinder as the model for all of the bananas. This is consistent with the results for multiview data. However, compared with Figure 6.20, the fitting results are now much more diverse, and the differences between the minimum residuals and the others are significantly reduced. Because the banana shape was not regular, the model estimated from single-view data was biased by the partial shape information in the data, as shown in Figure 6.23 (a) and (b). Using active vision, uncertainty associated with these model parameters could be analysed further to guide the collection of more data from other views [**140, 141**]. When multiview data are used, the algorithm obtains much more accurate models, as demonstrated in Figure 6.23 (c) and (d).

5.5. Comparing Perfect and Imperfect Geon-like Objects Here we compare the parameter dispersion obtained by fitting parametric geons to single-view data of perfect and imperfect geon-like objects. We obtained four sets of data by scanning a curved plastic tube whose shape resembled a perfect curved cylinder. The data of imperfect geon-like objects consisted of 44 sets of single-view data of the eleven bananas, each of which was scanned from four different views. Three scale parameters along the X, Y, Z axes and the bending curvature parameter were examined. We cannot compare the transformation parameters because the two types of objects are represented in terms of their individual views. Because of differences in size and axis curvature, we used the *coefficient of variation*, defined as the ratio of the standard deviation and the mean, as the measure of *relative* dispersion of each estimated parameter.

Figure 6.24 shows that the parameter dispersion is much larger for the bananas than the plastic tube. This is because the shape variations of imperfect objects in some views may

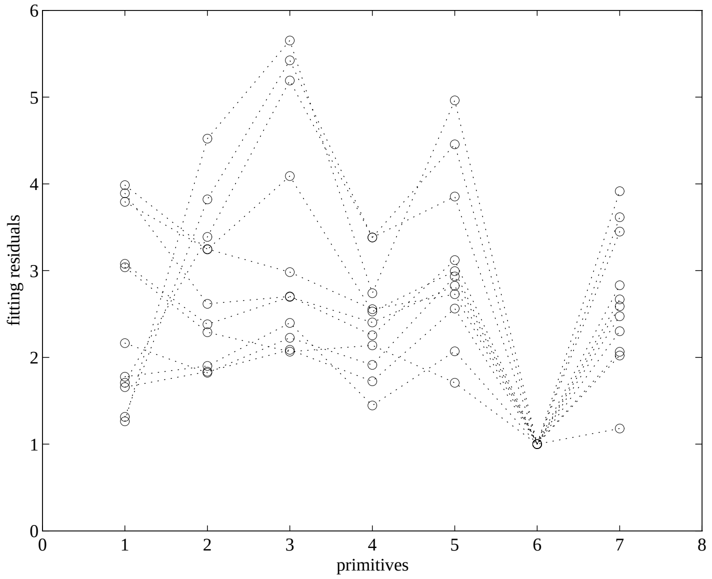


FIGURE 6.22. Fitting residuals obtained from single-view data of the eleven bananas.

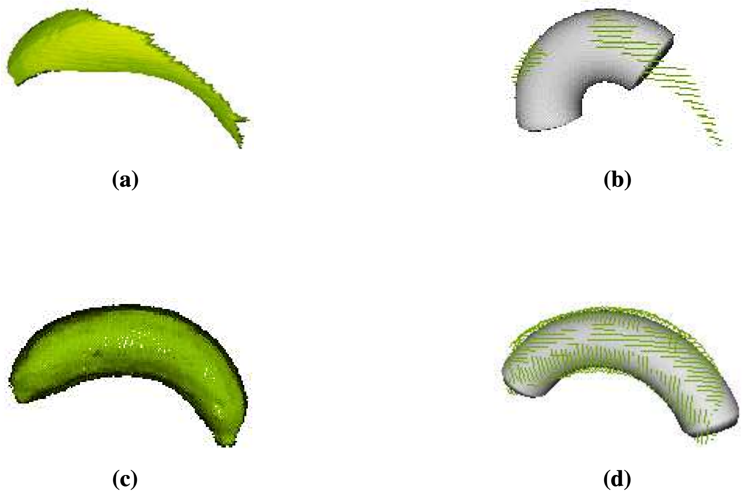


FIGURE 6.23. Fitting a model to single-view (a) and multiview data (b) of a banana. The left column shows the actual range data and the right column shows the model superimposed on the range data.

be more than in other views, and data from perfect geon-like objects in single views contain more consistent information. Thus, the imperfection in shape makes it much more difficult to obtain unique quantitative information using single-view data. This also suggests that employing multiview data is very important in parametric model recovery, especially when the object shapes are not highly consistent with the model shapes.

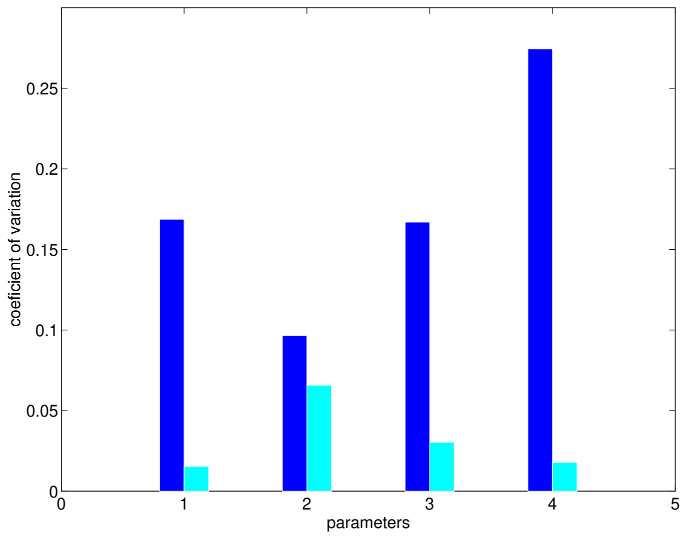


FIGURE 6.24. Comparing bananas and plastic tubes. The lighter bar denotes the coefficient of variation of four estimated parameters for the plastic tube. The darker bar indicates the coefficient of variation of four parameters *averaged over* eleven bananas. The horizontal axis indicates the parameters. 1, 2, 3 are the model size parameters along the X, Y, Z axes, respectively, and 4 is the bending curvature.

5.6. Using Multi-part Objects We have also conducted experiments with multi-part objects, which have been segmented into parts by the method described in Chapter 4. Figure 6.25 (a) is the side view of range data of an object and (b) is its parametric geon model consisting of a cylinder and an ellipsoid. Figure 6.25 (c) is the range image of a toy bowling pin and (d) is its model consisting of a tapered cylinder and an ellipsoid. Figure 6.25 (e) is the range image of a carved stone owl and (f) is its model. The head and torso are identified as two curved cylinders and the model of the feet is a tapered cylinder. These results again indicate that (1) our method works best when an object is composed of perfect geon-like parts and complete shape information is available (see (a) and (b)); (2) When complete shape information is available but the object is composed of imperfect geon-like parts, our method can also obtain a good result (see (c) and (d)); (3) when only partial shape information is available and the object is not composed of perfect geon-like parts, our method can still obtain the satisfactory qualitative results (see (e) and (f)).

6. Chapter Summary

We have conducted systematic experiments using multiview and single-view range data to test our part segmentation and identification methods. Experimental results demonstrate that charge density distributions possess both fine and gross shape information and can be computed efficiently and robustly. Part segmentation using the charge density distribution

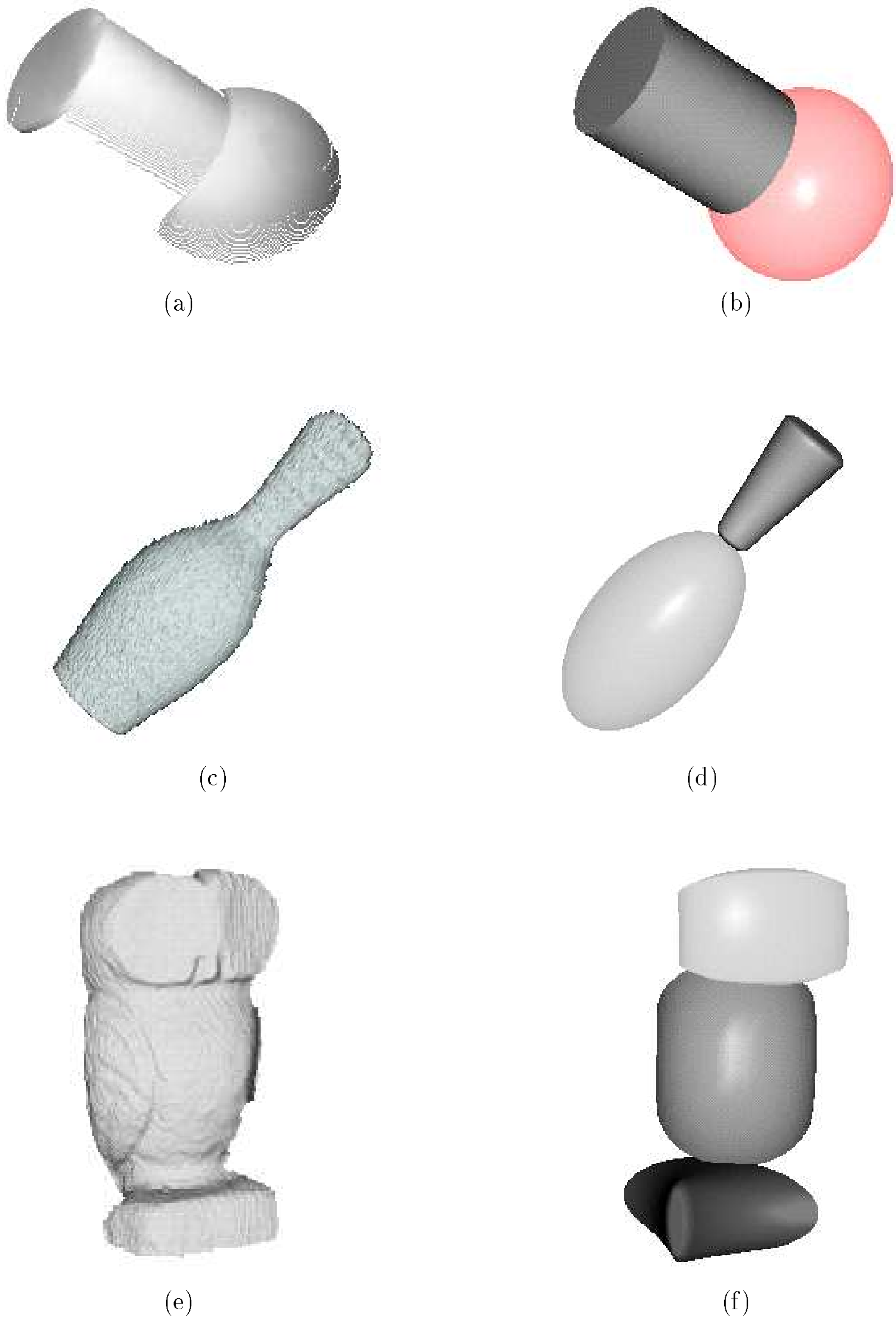


FIGURE 6.25. Part-based descriptions of objects. Range data of three objects are shown in the left column. The part-based descriptions of these three objects are presented in the right column.

as a surface feature indicator is successful if the object shape satisfies our segmentation assumption.

We have also demonstrated the strong performance of our approach to shape approximation of object parts by parametric geons. When using single-view data of objects which do not consist of geon-like parts, we can uniquely obtain qualitative shape information. However the quantitative information of parametric geons is often diverse and unreliable. When multiview data of the same objects are used, we can robustly recover the parametric geon models with much more consistent quantitative information. The newly defined objective function with both t_1 and t_2 terms produces much more discriminative fitting residuals than that with only the term t_1. Our fitting technique using global optimisation can reliably produce unique shape types of object parts. By directly comparing the part shape with model shapes, we accomplish explicit shape verification of the resultant part models.

CHAPTER 7

CONCLUSIONS

In the previous chapters, we have presented a new approach to qualitative volumetric shape-based representation of 3D objects sensed by a laser rangefinder. Here we summarise the thesis and consider: (1) the contributions made to knowledge in the field of computer vision, (2) the limitations of the current system, and (3) potential improvements.

1. Thesis Summary

A prerequisite for an autonomous robot to explore its environment is its ability to recognise objects perceived by a visual sensor. In general, image data acquired by a laser rangefinder only provide the distances from the sensor to object surfaces in the scene. Thus, range images alone cannot be directly used for identifying individual objects. Efficient object recognition by machine requires a symbolic description derived from image data of the object which can be matched to existing models in a database. To compute such descriptions, it is necessary to impose meaningful constraints derived from human vision studies. Object descriptions must be able to characterise a variety of the sensed objects and be insensitive to sensor noise and minor object shape variations. Moreover, the object representation process must be able to verify the resultant shape descriptions with the object shapes.

From nature we observe that many objects consist of parts. Consider the theory of human image understanding, Recognition-By-Components (RBC) [**15**], which addresses the issue of part-based recognition. To represent object parts, the RBC theory derives a finite set of generic volumetric primitives, called geons. It also postulates that if an arrangement of a few geons can be recovered from the image, objects can be recognised quickly even when they are occluded, rotated in depth or degraded. Inspired by RBC, considerable research has been carried out on geon-based object representation and recognition. However, nearly all of this work has focused on the recovery of geon models from complete edge maps or ideal line drawings depicting objects whose parts are instances of geon models. The problem of recovering a part description in terms of perfect models from a non-ideally shaped part has not been considered. This thesis addresses this challenging issue. Specifically, a novel physics-based approach is proposed for part segmentation and a new top-down strategy is introduced for recovering geon models from imperfect geon-like parts.

Object segmentation decomposes an object into parts at the part boundaries. In Chapter 4, we introduced a physics-based property, *the simulated electrical charge density distribution*, to characterise the object shape, and segment the object into parts at deep surface concavities where the charge density achieves a local minimum. This method is motivated by the analogy between the singularity in surface tangents and the singularity in the charge density distribution over object surfaces. The simulated charge density differs significantly from surface curvature, a commonly-used surface property. The former is computed by solving a set of integral equations which does not require smooth surfaces. However, surface curvature is derived from second derivatives which does require surface smoothness. Curvature computations depend on local data [**14**] while the charge density computation uses global data. Local computations based on differentiation are very sensitive to sensor noise, and a local scale must be specified for the computation. It is usually difficult to choose such a scale since it depends on the object structure which is not explicitly provided by range data. By contrast, the global electrical charge computation reduces the local noise effects and does not require that a particular scale be selected. In addition, the charge density can reveal the object's fine and gross structures simultaneously and indicate the significance of part protrusions. Although the charge density computation requires a complete triangulation of object surfaces, it does not work in a voxel-based coordinate system and furthermore does not perform computations in object interiors. Thus, it requires much less parameter selection and works more efficiently than a particle diffusion-based approach [**151**], which must compute the interior of objects in a voxel-based coordinate system.

Part model identification obtains shape approximations using *parametric geons*. In Chapter 5, we have defined parametric geons as seven qualitative shapes inspired by the volumetric primitives commonly used in sculpture. Each parametric geon is formulated by an implicit equation of *restricted* deformed superellipsoids. The attributes of the parametric geons such as pose, size, tapering rates and curvature of the axis are specified by parameters. We fit all of the parametric geons to an object part and select the best model based on the minimum fitting residual. The formulation of the models provides explicit global shape constraints, which restricts the resultant part descriptions to predefined models. Thus, this approach can recover part models even when objects consist of imperfect geon-like parts.

More importantly, since we directly examine the similarity between part and the model shapes, an explicit shape verification of the resulting descriptions can be accomplished. This is in contrast to all previous work on geon-based representations, which did not perform explicit shape verification. The new objective function and the use of global optimisation technique (Very Fast Simulated Re-annealing) have made the part model recovery more robust.

Chapter 6 examined our approach through systematic experimentation using both single- and multiple-view range data. We showed that the charge density computation is more robust to high frequency noise than the curvature computation. The charge density distribution can provide both gross and fine shape information and can indicate the significance of part protrusions. We successfully performed object segmentation into parts. We demonstrated that our approach can recover models from object parts, whose shapes were not consistent with the model shapes. Experimental results also revealed that (1) the newly-defined objective function produced more distinctive fitting residuals for shape discrimination than that used in previous work [**55**]; (2) for single-view data, the model fitting results were more stable when using perfect geon-like parts than imperfect geon-like ones; (3) for imperfect parts, the model fitting procedure using multiview data produced much more robust results than using single-view data.

2. Thesis Contributions

We have introduced a new paradigm for qualitative volumetric shape-based representation of 3D objects found in range data. The major contribution that this research makes to knowledge in the field of computer vision is the physics-based approach to object segmentation into parts. It is based on an analysis of the *simulated electrical charge density distribution on the object surface*. To our knowledge, this is the first time such an analogy has been used to characterise an object's shape and segment an object into parts. This approach provides a superior alternative to the traditional geometry-based approaches and creates a new direction for object shape representation. The novel aspects of our research are described as follows:

Shape Characterisation: We propose a physical property, *the simulated electrical charge density distribution*, to characterise object shapes. The charge density exhibits the relative contrast of concavity and convexity of object shapes. It is computed by using global data but clearly manifests local features. The computation of charge densities is unique for a given shape, invariant to translation and rotation, robust to sensor noise and not sensitive to surface triangulation.

Part Segmentation: We segment an object into parts at the part boundaries which are characterised by the local minima of the simulated charge densities. This approach is motivated by an analogy between the singularity of surface tangent planes and the singularity of the charge density distribution on the surface of a perfect conductor. All traditional approaches have employed geometrical properties, such as curvature or volumetric shape models, for part segmentation.

Model Recovery: We introduce parametric geons as the part models by explicitly specifying the qualitative shapes of deformed superellipsoids. Parametric geons convey qualitative shape and quantitative pose and deformation information. We directly compare the part shape with all parametric geons and select the model whose shape is most similar to the part shape. In this way, we can achieve an explicit shape verification of the resultant descriptions of parts. All previous approaches on geon recovery do not perform explicit shape verification. A new objective function defined in terms of mixed L_1 and L_2 norms and a fast global optimisation technique (Very Fast Simulated Re-Annealing) are employed to obtain models robustly.

Experimentation: We have developed a software system for achieving above and successfully obtained parametric geon descriptions of multi-part objects. We have also studied the properties of charge density distributions on 2D contours and 3D surfaces and compared the characteristics of the charge density and curvature. In addition, we have investigated the performance of model recovery affected by different objective functions, shape imperfection and the amount of input data.

3. Limitations

3D object representation is obviously a complex problem, and this research has attempted to address only certain issues. Those omitted must be dealt with in future research in order to produce a usable object recognition system. We elaborate on some of the problems in the following:

- We have assumed that the segmentation of an object from its background could be easily performed. In practice, we achieved this figure-ground separation by using *a priori* information of 3D space obtained from an off-line calibration process. If a robot navigates in an unknown environment, this calibration would not be available. Other context free approaches are required to segment the object from its background.

- We have assumed that the part boundary on the object surface must be explicitly indicated by deep surface concavities. This is the requirement for all boundary-based approaches which follow the principle of transversality regularity [**60, 56**]. However, in practical situations, many objects do have obvious parts, which are not completely separated by deep surface concavities. Examples are the "elbow" in Figure 2.2 and the elephant in Figure 6.12. A different part segmentation theory is needed for segmenting these kinds of objects.
- We have assumed that the object of interest consists of multiple parts. This is because the charge density alone is insufficient to distinguish between multi-part and single-part objects. The charge density only indicates the *contrast* between concavity and convexity and cannot measure the absolute convexity and concavity. In practice, an approach to object representation should first examine whether a sensed object is composed of multiple parts. If it is, then part segmentation is needed.
- In order to compute the charge density distribution, a triangular mesh tessellation on the object surface is required. In the case of multiview data whose local spatial relations are not explicitly specified, it is still difficult to do surface tessellation automatically for complex shapes. In the case of single-view range data whose local spatial relations are specified by the range image grid, surface tessellation can be done easily. However, since this operation is based on local data, dense range data is required.

By taking these limitations into consideration, a more powerful object representation system can be developed.

4. Future Work

To improve the current system and extend our research, we suggest the following:

- An investigation can be carried out for a part-based *object recognition system* using parametric geons. Matching the object descriptions computed from range data against parametric geon models in a database can be conducted in the following way. Qualitative shape and the number parts should be examined first. If these are not enough to identify the object, quantitative attributes, such as pose, size, tapering rates and curvature of the axis should be consulted. Such a recognition system also possesses another feature. Although the number of parametric geons totals only seven, various spatial arrangements of these primitives can make up a large number of object models. Therefore, such a system will be able to recognise many objects efficiently [**15**].

- More research can be carried out on the *applications of simulated charge density*. In this thesis, we only use the simulated charge density distribution to extract surface concavity. In a similar fashion, surface convexity could also be detected. Since the simulated charge density distribution contains both gross and fine object shape information, it is very useful for characterising shape. In addition, the significance of parts, which is revealed in the simulated charge density distribution, can also be used to characterise object parts. Moreover, local feature detections based on other similar integration mechanisms would be worth studying.
- Our approach provides a useful tool to derive *higher-level descriptions of objects from triangulation models*. We note that research on object shape reconstruction has extensively used such triangular mesh descriptions. However, there is almost no work which directly uses the triangular meshes as the model for recognition tasks. This is because surface triangulation models lack uniqueness with respect to scale and viewpoints. The number and positions of triangles will vary when an object is sensed from different directions and distances. Thus, triangulation models do not appear particularly well for defining object classes. By simulating charge density distributions on the complete mesh, the object shape could be characterised and higher-level shape descriptions could be derived.
- Another important future research direction is the integration of different sources of information to perform *figure-ground separation*. Range images contain only depth information, which may sometimes be insufficient for the extraction of an object from its background. Consider a cup on the top of a table and a laser rangefinder scanning the scene from a 45 degree angle. The distances from data points on the table top to the sensor may be larger, or equal to or smaller than the distances from the cup to the sensor. In this case, the cup cannot be completely separated from the table top based only on depth information. Thus, other sources of information, such as colour or gray level intensity, may provide superior contrast for the object and its background and, therefore, be useful for figure-ground separation. Of course, this requires a video camera and a calibration between range and intensity images.

- Future research can be extended to *decrease the computational time* required. The most time-consuming stage in the charge density computation is the construction of the coefficient matrix A in Equation (4.13). Since the computation of each matrix element is independent of others, parallel computation could be used. Similarly, the most time-consuming stage in the model fitting procedure is the evaluation of the complex objective function in Equation (5.18). Again the evaluation at each data point is independent of the others. Thus, parallel computation could be useful here as well.

In conclusion, this thesis presents a new approach to qualitative part-based representations of 3D objects sensed by a laser rangefinder. The research is a theoretical investigation of a generic object recognition system rather than being limited to a single application. We focus on the issues of part segmentation and part model identification. The experimental results demonstrate that our approach can recover qualitative part-based descriptions even when objects consist of imperfect geon-like parts. This research provides a basis for future study in qualitative shape recognition and autonomous robot task performance.

REFERENCES

[1] G. H. Abdel-Hamid and Y.-H. Yang. Electrostatic field-based detection of corners of planar curves. In *Proceedings of the 1993 Canadian Conference on Electrical and Computer Engineering*, pages 767–770, Vancouver, Canada, September 1993.

[2] G. H. Abdel-Hamid and Y.-H. Yang. Multiresolution skeletonization: An electrostatic field-based approach. In *Proceedings of The IEEE International Conference on Image Processing*, pages 13–16, Austin, USA, November 1994.

[3] G. J. Agin and T. O. Binford. Computer description of curved objects. In *Proceedings of the 3rd International Joint Conference on Artificial Intelligence*, pages 1113–1115, Stanford, California, August 1973.

[4] T. Arbel, P. Whaite, and F. P. Ferrie. Recognizing volumetric objects in the presence of uncertainty. In *Proceedings 12th International Conference on Pattern Recognition*, pages 470–476, Jerusalem, Israel, Oct 9-13 1994. IEEE Computer Society Press.

[5] H. Asada and M. Brady. The curvature primal sketch. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(1):2–14, January 1986.

[6] R. Bajcsy and F. Solina. Three dimensional object representation revisited. In *Proceedings of the First International Conference on Computer Vision*, pages 231–241, London, England, 1987.

[7] R. Bajcsy, F. Solina, and A. Gupta. Segmentation versus object representation-are they separable? In P. Jain and A. Jain, editors, *Analysis and Interpretation of Range Images*, pages 207–223. Springer–Verlag, 1990.

[8] A. H. Barr. Superquadrics and angle-preserving transformations. *IEEE Computer Graphics Applications*, 1:11–23, 1981.

[9] A. H. Barr. Global and local deformations of solid primitives. *Computer Graphics*, 18(3):21–30, 1984.

[10] R. Barrett, M. Berry, T. F. Chan, et al. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods.* SIAM, Philadelphia, 1994.

[11] B. M. Bennett and D. D. Hoffman. Shape decompositions for visual shape recognition: The role of transversality. In S. Ullman and W. Richards, editors, *Image Understanding II*, pages 215–255. Ablex Publishing Corporation, N.J., 1989.

[12] R. Bergevin and M. D. Levine. Part decomposition of objects from single view line drawings. *CVGIP: Image Understanding*, 55(1):73–83, January 1992.

[13] R. Bergevin and M. D. Levine. Generic object recognition: Building and matching coarse descriptions from line drawings. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(1):19–36, January 1993.

[14] P. J. Besl and R. C. Jain. Invariant surface characteristics for three dimensional object recognition in range images. *Computer Vision, Graphics, and Image Processing*, 33(1):33–88, 1986.

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

[16] I. Biederman. Aspects and extensions of a theory of human image understanding. In Z. W. Pylyshyn, editor, *Computational Processes in Human Vision*, chapter 14, pages 370–428. Ablex Pub. Corp., Norwood, NJ, 1988.

[17] I. Biederman. Visual object recognition. In A. I. Goldman, editor, *Readings in Philosophy and Cognitive Science*, pages 9–21. MIT Press, Cambridge, MA, US, 1993.

[18] I. Biederman and E. Cooper. Priming contour-deleted images: Evidence for intermediate representations in visual object recognition. *Cognitive Psychology*, 23:394–419, 1991.

[19] I. Biederman and E. E. Cooper. Evidence for complete translational and reflectional invariance in visual object priming. *Perception*, 20:585–593, 1991.

[20] I. Biederman and E. E. Cooper. Size invariance in visual object priming. *Journal of Experimental Psychology: Human Perception & Performance*, 18(1):121–133, February 1992.

[21] I. Biederman and P. C. Gerhardstein. Recognizing depth-rotated objects: Evidence and conditions for three-dimensional viewpoint invariance. *Journal of Experimental Psychology: Human Perception & Performance*, 19(6):1162–1182, 1993.

[22] T. O. Binford. Visual perception by computer. In *IEEE Conference on Systems and Control*, Miami, FL, 1971.

[23] T. O. Binford. Survey of model-based image analysis system. *The International Journal of Robotics Research*, 1(1):18–63, 1982.

[24] J. Van Bladel. *Singular Electromagnetic Fields and Sources*. Oxford University Press, Oxford, 1991.

[25] G. Blais and M. D. Levine. Registering multiview range data to create 3D computer objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):820–830, August 1995.

[26] J. D. Boissonnat. Geometric structures for three-dimensional shape representation. *ACM Transactions on Graphics*, 3(4):267–286, October 1984.

[27] T. Boult and A. Gross. Recovery superquadrics from depth information. In *Proceedings of the AAAI Workshop on Spatial Reasoning and Multisensor Integration*, pages 128–137. American Association for Artificial Intelligence, 1987.

[28] M. Brady. Criteria for representations of shape. In J. Beck, B. Hope, and A. Rosenfeld, editors, *Human and Machine Vision*, pages 39–84. Academic Press, New York, 1983.

[29] F. J. Bueche. *Introduction to Physics for Scientists and Engineers*. McGraw-Hill Book Company, New York, 3rd edition, 1980.

[30] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, November 1986.

[31] Y. Chen and G. Médioni. Object modeling by registration of multiple range images. In *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, pages 2724–2729, Sacramento, CA, 1991.

[32] Y. Chen and G. Medioni. Surface description of complex objects from multiple range images. In *Proceedings of IEEE Conference on Computer Vision & Pattern Recognition*, pages 153–158, Seattle, June 1994.

[33] E. E. Cooper, I. Biederman, and J. E. Hummel. Metric invariance in object recognition: A review and further evidence. *Canadian Journal of Psychology*, 46(2):191–214, June 1992.

[34] T. Darrell, S. Sclaroff, and A. Pentland. Segmentation by minimal description. In *Proceedings of the Third International Conference on Computer Vision*, pages 112–116, Osaka, Japan, December 1990.

[35] D. DeCarlo and D. Metaxas. Adaptive shape evolution using blending. In *IEEE Proceedings of International Conference on Computer Vision*, pages 834–839, 1995.

[36] S. J. Dickinson, R. Bergevin, I. Biederman, et al. The use of geons for generic 3-D object recognition. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, volume 2, pages 1693–1699, Chambery, France, September 1993.

[37] S. J. Dickinson, H. I. Christensen, J. Tsotsos, et al. Active object recognition integrating attention and viewpoint control. In Jan-Olof Eklundh, editor, *Proceedings of Third European Conference on Computer Vision*, volume II, pages 3–14, Stockholm, Sweden, May 1994. Springer-Verlag.

[38] S. J. Dickinson and D. Metaxas. Integrating qualitative and quantitative shape recovery. *International Jounral of Computer Vision*, 13(3):311–330, 1994.

[39] S. J. Dickinson, D. Metaxas, and A. Pentland. Constrained recovery of deformable models from range data. In *Proceedings of Internaitional Workship on Visual Form*, Capri, Italy, May 1994.

[40] S. J. Dickinson, A. P. Pentland, and A. Rosenfeld. A representation for qualitative 3D object recognition integrating object-centered and viewer-centered models. In K. N. Leibovic, editor, *Vision: A Convergence of Disciplines*. Springer Verlag, New York, 1990.

[41] S. J. Dickinson, A. P. Pentland, and A. Rosenfeld. 3D shape recovery using distributed aspect matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):174–198, February 1992.

[42] S. J. Dickson, A. P. Pentland, and A. Resenfeld. From volumes to views: An approach to 3-D object recognition. *CVGIP: Image Understanding*, 55(2):130–154, March 1992.

[43] G. Dudek and J. K. Tsotsos. Shape representation and recognition from curvature. In *Proceedings of the 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 35–41, Hawaii, June 1991. IEEE Computer Society.

[44] R. Fairwood. Recognition of generic components using logic-program relations of images contours. *Image and Vision Computing*, 9:113–122, 1991.

[45] O. D. Faugeras, M. Hebert, P. Mussi, and J. D. Boissonnat. Polyhedral approximation of 3-D objects without holes. *Computer Vision, Graphics, and Images Processing*, 25:169–183, 1984.

[46] F. P. Ferrie, J. Lagarde, and P. Whaite. Recovery of volumetric object descriptions from laser rangefinder images. In *Proceedings of the First European Conference on Computer Vision*, pages 387–396, Antibbes, France, April 1990.

[47] F. P. Ferrie, J. Lagarde, and P. Whaite. Darboux frames, snakes and superquadrics: Geometry from the bottom up. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(8):771–784, August 1993.

[48] F. P. Ferrie and M. D. Levine. Integrating information from multiple views. In *Proceedings of the IEEE Computer Society Workshop on Computer Vision*, pages 117–122, Miami, Florida, Dec. 1987.

[49] F. P. Ferrie and M. D. Levine. Deriving coarse 3D models of objects. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 345–353, Ann Arbor, Michigan, June 1988.

[50] H. Freeman. Shape descriptions via the use of critical points. *Pattern Recognition*, 10(3):159–166, 1978.

[51] M. Gardiner. The superellipse: A curve that lies between the ellipse and the rectangle. *Scientific American*, 213:222–234, 1965.

[52] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, November 1984.

[53] P. J. Giblin. *Graphs, Surfaces and Homology.* Chapman and Hall Ltd, 2nd edition, 1981.

[54] G. Godin and M. D. Levine. Structured edge map of curved objects in a range image. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 276–281, 1989.

[55] A. D. Gross and T. E. Boult. Error of fit measures for recovering parametric solids. In *Proceedings of the 2nd International Conference on Computer Vision*, pages 690–694, Tampa, Florida, 1988.

[56] V. Guillemin and A. Pollack. *Differential Topology.* Prentice-Hall, Englewood Cliffs, NJ, 1974.

[57] A. Gupta and R. Bajcsy. Volumetric segmentation of range images of 3D objects using superquadric models. *CVGIP: Image Understanding*, 58(3):302–326, November 1993.

[58] F. R. Hammpel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel. *Robust Statistics: The Approach Based on Influence Functions.* Wiley, New York, 1986.

[59] G. Hatfield and W. Epstein. The status of the minimum principle in the theoretical analysis of visual perception. *Psychological Bulletin*, 97(2):155–186, 1985.

[60] D. Hoffman and W. Richards. Parts of recognition. *Cognition*, 18:65–96, 1984.

[61] R. Hoffman and A. K. Jain. Segmentation and classification of range images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):608–620, 1987.

[62] T. Horikoshi and S. Suzuki. 3D parts decomposition from sparse range data using information criterion. In *Proceedings of 1993 IEEE Computer Society Conerence on Computer Vision and Pattern Recognition*, pages 168–173, New York City, NY, June 1993. IEEE Computer Society Press.

[63] B. K. P. Horn. *Robot Vision.* The MIT Press, Cambridge, MA, 1986.

[64] L. S. Horowitz and T. Pavlidis. Picture segmentation by direct split-and-merge procedure. In *Proceedings of the 2nd International Joint Conference on Pattern Recognition*, pages 424–433, Copenhagen, August 1974.

[65] J. Hummel and I. Biederman. Dynamic binding in a neural network for shape recognition. *Psychological Review*, 99(3):480–517, July 1992.

[66] L. Ingber. Very fast simulated re-annealing. *Mathematical and Computer Modelling*, 12(8):967–973, 1989.

[67] J. D. Jackson. *Classical Electrodynamics.* Wiley, New York, 1975.

[68] A. Jacot-Descombes and T. Pun. A probabilistic approach to 3-D inference of geons from a 2-D view. In K. W. Bowyer, editor, *Applications of Artificial Intelligence X: Machine Vision and Robotics, Proceedings of SPIE*, volume 1708, pages 579–588, Orlando, Florida, April 1992.

[69] A. R. Jarvis. A perspective on range finding techniques for computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5(2):122–139, March 1983.

[70] D. Keren, D. Cooper, and J. Subrahmonia. Describing complicated objects by implicit polynomials. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(1):38–52, Januray 1994.

[71] S. Kirkpatrick, Jr. C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, May 1983.

[72] J. J. Koenderink and A. J. Van Doorn. The shape of smooth objects and the way contours end. *Perception*, 11:129–137, 1982.

[73] E. Koh, D. Metaxas, and N. Badler. Hierarchical shape representation using locally adaptive finite elements. In Jan-Olof Eklundh, editor, *Lecture Notes on Computer Sience, Computer Vision-ECCV'94*, volume 800, pages 441–446. Springer-Verlag, 1994.

[74] S. Kumar, S. Han, D. Goldgof, and K. Bowyer. On recovering hyperquadrics from range data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(11):1079–1083, November 1995.

[75] E. Leeuwenberg and P. van der Helm. Unit and variety in visual form. *Perception*, 20:595–622, 1991.

[76] A. Lejeune and F. Ferrie. Partioning range images using curvature and scale. In *Proceedings of 1993 IEEE Computer Society Coonference on Computer Vision and Pattern Recognition*, pages 800–801, New York City, NY, June 1993. IEEE Computer Society Press.

[77] A. Leonardis, F. Solina, and A. Macerl. A direct recovery of superquadric models in range images using recover-an-select paradigm. In Jan-Olof Ekklundh, editor, *Proceedings of Third European Conference on Computer Vision*, volume I, pages 309–318, Stockholm, Sweden, May 1994. Spring-Verlag.

[78] R. G. Lerner and G. L. Trigg. *Encyclopedia of Physics*. VCH Publishers, Inc., New York, 2nd edition, 1991.

[79] M. D. Levine. *Vision in Man and Machines*. McGraw-Hill, Inc., New York, 1985.

[80] M. Leyton. Inferring causal history from shape. *Cognitive Science*, 13:357–389, 1989.

[81] D. G. Lowe. *Perceptual Organization and Visual Recognition*. Kluwer Academic Publishers, Hingham, MA, USA, 1985.

[82] D. Marr and H. K. Nishihara. Representation and recognition of spatial organization of three-dimensional shapes. *Proceedings of the Royal Society*, B200:269–294, 1978.

[83] T. McInerney and D. Terzopoulos. A finite element model for 3D shape reconstruction and nonrigid motion tracking. In *Proceedings of Fourth International Conference on Computer Vision*, pages 518–523, Los Alamitos, CA, May 1993. IEEE Computer Society Press.

[84] R. Mohr and R. Bajcsy. Packing volumes by spheres. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5:111–116, 1983.

[85] R. C. Munck-Fairwood and L. Du. Shape using volumetric primitives. *Image & Vision Computing*, 11(6):364–371, July 1993.

[86] P. J. Besl nad R. C. Jain. Intrinsic and extrinsic surface characteristics. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 226–233, San Francisco, CA, June 1985.

[87] R. Nevatia and T. O. Binford. Descriptions and recognition of curved objects. *Artificial Intelligence*, 8:77–98, 1977.

[88] Q. L. Nguyen and M. D. Levine. 3D object representation in range images using geons. In *Proceedings of the 11th International Conference on Pattern Recognition*, Hague, Netherlands, August 1992.

[89] B. O'neill. *Elementary Differential Geometry*. Academic Press, Nork and London, 1966.

[90] J. O'Rourke and N. Badler. Decomposition of three-dimensional objects into spheres. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1:295–305, 1979.

[91] S. Palmer, E. Rosch, and P. Chase. Canonical perspective and the perception of objects. In J. Long and A. Baddeley, editors, *Attention and Performance IX*, pages 135–151. Erlbaum Associates, Hillsdale, N.J., 1981.

[92] A. Pentland. Part segmentation for object recognition. *Neural Computation*, 1:82–91, 1989.

[93] A. P. Pentland. Perceptual organization and the representation of natural form. *Artificial Intelligence*, 28:293–331, 1986.

[94] A. P. Pentland. Recognition by parts. In *Proceedings of the First International Conference on Computer Vision*, pages 8–11, London, June 1987.

[95] A. P. Pentland. Automatic extraction of deformable part models. *International Journal of Computer Vision*, 4:107–126, 1990.

[96] A. P. Pentland and S. Sclaroff. Closed-form solutions for physically based shape modeling and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:715–729, 1991.

[97] M. Potmesil. Generating models of solid objects by matching 3D surface segments. In *Proceedings of the 8th International Joint Conference on Artificial Intelligence*, pages 1089–1093, Karlsruhe, West Germany, August 1983.

[98] R. H. Price and R. J. Crowley. The lightning-rod fallacy. *American Journal of Physics*, 53(9):843–848, September 1985.

[99] B. Putnam. *The Sculptor's Way*. Farrar & Rinehart, INC., 1939.

[100] N. S. Raja and A. K. Jain. Recognizing geons from superquadrics fitted to range data. *Image and Vision Computing*, 10(3):179–190, April 1992.

[101] N. S. Raja and A. K. Jain. Obtaining generic parts from range data using a multi-view representation. *CVGIP: Image Understanding*, 60(1):44–64, July 1994.

[102] K. Rao and R. Nevatia. Describing and segmenting scenes from imperfect and incomplete data. *CVGIP: Image Understanding*, 57(1):1–23, January 1993.

[103] S. M. Rao. *Electromagnetic scattering and radiation of arbitrarily-shaped surfaces by triangular patch modeling*. PhD thesis, University of Mississippi, August 1980.

[104] L. R. Rogers. *Sculpture*. Oxford University Press, 1969.

[105] H. Rom and G. Medioni. Part decomposition and description of 3D shapes. In *Proceedings of the 12th International Conference on Pattern Recognition*, volume I, pages 629–632, Jerusalem, Israel, October 1994. IEEE computer Society, IEEE Computer Society Press.

[106] C. Ronse and P. A. Devijver. *Connected Components in Binary Images: the Detection Problem*. Research Studies Press, England, 1984.

[107] A. Rosenfeld and J. J. Pfaltz. Sequential operations in digital picture processing. *Jounral of the Association of for Computing Machinery*, 13(4):471–494, October 1966.

[108] J. B. Ross. Plotting the charge distribution of a closed-loop conducting wire using a microcomputer. *American Journal of Physics*, 55:948–950, 1987.

[109] B. Sabata, F. Arman, and J. K. Aggarwal. Segmentation of 3D range images using pyramidal data structures. *CVGIP: Image Understanding*, 57(3):373–387, May 1993.

[110] P. T. Sander and S. W. Zucker. Inferring surface trace and differential structure from 3-D images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(9):833–854, September 1990.

[111] W. L. Schaich and S. Mahmud. On the distribution of charge over the surface of a conductor. *American Journal of Physics*, 59:656–658, 1991.

[112] R. A. Serway. *Physics for Sciences & Engineers.* Saunders College Publishing, 2nd edition, 1986.

[113] S. A. Shafer and T. Kanade. The theory of straight homogeneous generalised cylinders. In *DARPA83-Image Understanding Workshop*, pages 210–218, 1983.

[114] L. G. Shapiro. Recent progress in shape decomposition and analysis. In L. N. Kanal and A. Rosenfeld, editors, *Progress in Pattern Recognition 2*, pages 113–123. Elsevier Science Publishers B. V. (North-Holland), New York, 1985.

[115] L. G. Shapiro, J. D. Moriarty, R. M. Haralick, and P. G. Mulgaonkar. Matching three-dimensional objects using a relational paradigm. *Pattern Recognition*, 17(4):385–405, 1984.

[116] J. S. Shieh. Relational fuzzy model: A representation for object identification. In *Proceedings of International Conference on Intelligent Robotic System*, pages 965–974, Piscataway, NJ, 1993. IEEE.

[117] K. Siddiqi and B. B. Kimia. Parts of visual form: Computational aspects. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 17(3):239–251, March 1995.

[118] K. Siddiqi, K. J. Tresness, and B. B. Kimia. Parts of visual form: Psychophysical aspects. *Perception*, Submitted, 1994.

[119] P. P. Silvester and R. L. Ferrari. *Finite Elements for Electrical Engineering.* Cambridge University Press, Cambridge, 2nd edition, 1990.

[120] O. Skliar and M. H. Loew. A new method for characterization of shape. *Pattern Recognition Letters*, 3:335–341, 1985.

[121] D. R. Smith and T. Kanade. Autonomous scene description with range imagery. *Computer Vision, Graphics and Image Processing*, 31(3):322–334, September 1985.

[122] W. R. Smythe. *Static and Dynamic Electricity.* McGraw-Hill, Inc., New York, third edition edition, 1968. page 124.

[123] F. Solina and R. Bajcsy. Recovery of parametric models from range images: the case for superquadrics with global deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(2):131–147, 1990.

[124] F. Solina, A. Leonardis, and A. Macerl. A direct part-level segmentation of range images using volumetric models. In *Proceedings of 1994 IEEE International Conference on Robotics and Automation*, pages 2254–2259, San Diego, CA, May 1994. IEEE Robotics and Automation Society, IEEE Computer Society Press.

[125] B. I. Soroka. Generalized cones from serial sections. *Computer Graphics and Image Processing*, 15:154–166, 1981.

[126] M. Soucy, A. Croteau, and D. Laurendeau. A multi-resolution surface model for compact representation of range images. In *Proceedings of IEEE International Conference on Robotics and Automation*, pages 1701–1706, May 1992.

[127] M. Soucy and D. Laurendeau. Generating non-redundant surface representations of 3D objects using multiple range views. In *Proceedings of the 10th Conference on Pattern Recognition*, pages 198–200, Atlantic city, New Jersey, June 1990.

[128] M. Soucy and D. Laurendeau. A general surface approach to the integration of a set of range views. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(4):344–358, April 1995.

[129] L. Stark and K. Bowyer. Achieving generalized object recognition through reasoning about association of function to structure. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 13(10):1097–1104, October 1991.

[130] L. Stark and K. W. Bowyer. Function-based generic recognition for multiple object categories. *CVGIP - Image Understanding*, 59(1):1–21, January 1994.

[131] S. Sullivan, L. Sandford, and J. Ponce. On using geometric distance fits to estimate 3D object shape, pose, and deformation from range, CT, and video images. In *Proceedings of 1993 IEEE Conference on Computer Vision and Pattern Recognition*, pages 110–115, New York City, NY, June 1993.

[132] D. Terzopoulos, , A. Witkin, and M. Kass. Symmetry-seeking models for 3D object reconstruction. *International Journal of Computer Vision*, 1(3):211–221, 1987.

[133] D. Terzopoulos and D. Metaxas. Dynamic 3D models with local and global deformations: Deformable superquadrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(7):703–714, 1991.

[134] E. Trucco. Part segmentation of slice data using regularity. *Signal Processing*, 32:73–90, 1993.

[135] E. Trucco and R. B. Fisher. Experiments in curvature-based segmentation of range data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(2):177–181, February 1995.

[136] B. Tversky and K. Hemenway. Objects, parts, and categories. *Journal of Experimental Psychology*, 113(2):169–191, June 1984.

[137] M. Vasilescu and D. Terzopoulos. Adaptive meshes and shells: Irregular triangulation, discontinuities, and hierarchical subdivision. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 829–832, Urbana-Champaign, IL, June 1992.

[138] B. C. Vemuri and J. K. Aggarwal. 3D model construction from multiple views using range and intensity data. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 435–438, Miami, Florida, June 1986.

[139] A. M. Wallace, E. E. Brodie, and S. Love. Discrimination between visual stimuli by variation of shape and relative position of volumetric primitives. *Image and Vision Computing*, 11(6):372–388, 1993.

[140] P. Whaite and F. P. Ferrie. From uncertainty to visual exploration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(10):1038–1049, October 1991.

[141] P. Whaite and F. P. Ferrie. Uncertain views. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 15–18, Champaign, Illinois, June 1992.

[142] D. Wilton, S. M. Rao, A. W. Glisson, et al. Potential integrals for uniform and linear source distributions on polygonal and polyhedral domains. *IEEE Transactions on Antennas and Propagation*, AP-32(3):276–281, 1984.

[143] R. J. Woodham. Stable representation of shape. In Z. W. Pylyshyn, editor, *Computational Progresses in Human Vision: An Interdisciplinary Perspective*. Ablex Publishing Corporation, Norwood, NJ, 1988.

[144] K. Wu and M. D. Levine. 3-D object representation using parametric geons. Technical Report TR-CIM-93-13, Centre for Intelligent Machines, McGill University, Montreal, Quebec, Canada, September 1993.

[145] K. Wu and M. D. Levine. Parametric geons: A discrete set of shapes with parameterized attributes. In *SPIE International Symposium on Optical Engineering and Photonics in Aerospace Sensing: Visual Information Processing III*, volume 2239, pages 14–26, Orlando, FL, April 1994. The International Society for Optical Engineering.

[146] K. Wu and M. D. Levine. Recovering parametric geons from multiview range data. In *Proceedings of IEEE Conference on Computer Vision & Pattern Recognition*, pages 159–166, Seattle, June 1994.

[147] K. Wu and M. D. Levine. Shape approximation: From multiview range images to parametric geons. In *Proceedings of 12th International Conference on Pattern Recognition*, volume 1, pages 622–625, Jerusalem, Israel, October 1994.

[148] K. Wu and M. D. Levine. 3D part segmentation: A new physics-based approach. In *Proceedings of IEEE International Symposium on Computer Vision*, pages 311–316, Coral Gables, FL, November 1995.

[149] K. Wu and M. D. Levine. Segmenting 3D objects into geons. In *The 8th International Conference on Image Analysis and Processing*, Lecture Notes in Computer Science 974, pages 321–334, Sanremo, Italy, September 1995. International Association for Pattern Recognition, Springer-Verlag.

[150] K. Wu and M. D. Levine. 2D shape segmentation: A new approach. *Pattern Recognition Letters*, 17(2):133–140, February 1996.

[151] Y. Yacoob and L. S. Davis. Labeling of human face components from range data. *CVGIP: Image Understanding*, 60(2):168–178, September 1994.

[152] Y. Yacoob and Y. I. Gold. 3D object recognition via simulated particles diffusion. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 442–449, San Diego, CA, June 1989. IEEE Computer Society Press.

[153] N. Yokoya, M. Kaneta, and K. Yamamoto. Recovery of superquadric primitives from a range image using simulated annealing. In *Proceedings of International joint Conference on Pattern Recognition*, volume 1, pages 168–172, 1992.

[154] W. Zorach. *Zorach Explains Sculpture: What It Means and How It Is Made*. Tudor Publishing Company, New York, 1960.

[155] S. W. Zucker. Early vision. In S. C. Shapiro, editor, *Encyclopedia of Artificial Intelligence*, volume 1, pages 394–420. John Wiley & Sons, Inc., 1992.

APPENDIX A

Integral Evaluation

In order to solve the set of linear equations given in Equation (4.13) in Chapter 4, we must first compute the integral in (4.16), which is referred to as the *potential integral* [**142**]. It is an integration of a 3D function over a polygonal domain. The problem can be stated formally as:

PROBLEM A.1. *Given a function $F = 1/|\mathbf{r} - \mathbf{r}'| : R^3 \to R$ and three vertices ($\mathbf{v}_i \in R^3, i = 1, 2, 3$) of a triangle T, compute the integration of F over T.*

Fortunately, the analytical expression of the potential integral has been obtained [**142**]. This significantly improves the accuracy and efficiency of the numerical computation. For the sake of completeness, we present the integral evaluation here. The derivation of this expression can be found in [**103**].

Figure A.1 shows the geometrical quantities used in the computation. The notation is similar to that used in [**142**]. The darker shaded triangle is the integration domain, T. The quantities used for computation are listed below:

P: a plane shown by a lighter shaded region containing the triangle, T

$\mathbf{n}$: the normal of P and T

i: the index of the sides of T

$\mathbf{l}_i$: the vector containing the ith side (thick solid line) of T

$\mathbf{u}_i$: the normal vector of $\mathbf{l}_i$ in P

$\mathbf{r}$: the vector position of the observation point

ρ: the vector position of the orthogonal projection of the observation point onto P

$\mathbf{R}$: the vector from the observation point to a point on T

$\mathbf{P}_i^0$: a vector perpendicular to the ith side of T and passing through the point at ρ

R_i^0: the distance from the observation point to the vector $\mathbf{l}_i$

$R_i^{\pm}$: distances from the observation point to two endpoints of the ith side of T

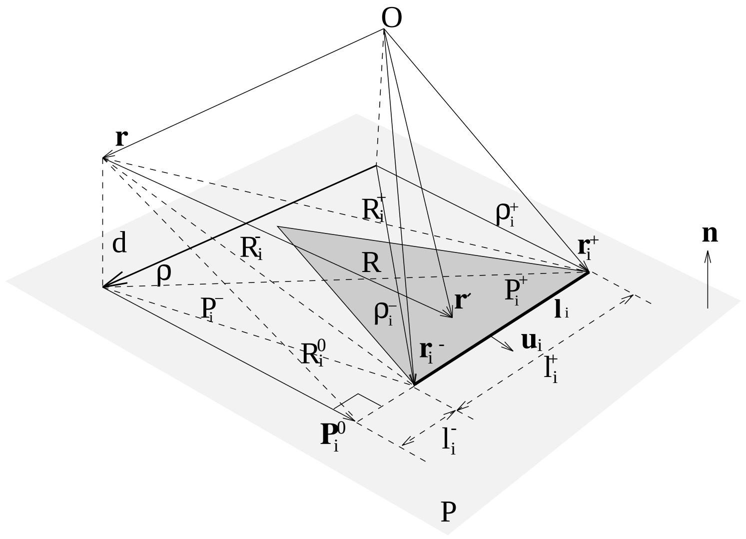


FIGURE A.1. The triangle patch of the charge source. Vector quantities and scalar quantities are denoted by solid lines with arrows and dashed lines, respectively. This is an adaptation of Figure 2 in [**142**].

d: the distance from the observation point to P

$P_i^{\pm}$: distances from a point at ρ to two endpoints of the ith side of T

$\mathbf{r}_i^{\pm}$: vector positions of two endpoints of the ith side of T

$l_i^{\pm}$: distances from two endpoints of the ith side to the projection of the point at ρ onto $\mathbf{l}_i$

$\rho_{\mathbf{i}}^{\pm}$: vectors from the projection of the origin onto P to two endpoints of the ith side.

Let

$$\rho_i^- = \mathbf{v}_1, \qquad \rho_i^+ = \mathbf{v}_2, \qquad \mathbf{n} = \frac{(\mathbf{v}_1 - \mathbf{v}_3) \times (\mathbf{v}_2 - \mathbf{v}_3)}{|(\mathbf{v}_1 - \mathbf{v}_3) \times (\mathbf{v}_2 - \mathbf{v}_3)|}$$

The evaluation of the integral in Equation (4.16) is given as follows [**142**]:

$$\begin{aligned} \int_T \frac{dS'}{|\mathbf{r} - \mathbf{r}'|} &= \sum_{i=1}^{3} \mathbf{P}_i^0 \cdot \mathbf{u}_i \Bigg[P_i^0 \ln \frac{R_i^+ + l_i^+}{R_i^- + l_i^-} \\ &\quad - |d| \left(\tan^{-1} \frac{P_i^0 l_i^+}{(R_i^0)^2 + |d| R_i^+} - \tan^{-1} \frac{P_i^0 l_i^-}{(R_i^0)^2 + |d| R_i^-} \right) \Bigg] \end{aligned} \tag{A.1}$$

where

$$\mathbf{l}_i = \frac{\mathbf{r}_i^+ - \mathbf{r}_i^-}{|\mathbf{r}_i^+ - \mathbf{r}_i^-|}, \qquad \mathbf{u}_i = \mathbf{l}_i \times \mathbf{n}, \qquad \rho_i^{\pm} = \mathbf{r}_i^{\pm} - \mathbf{n}(\mathbf{n} \cdot \mathbf{r}_i^{\pm}),$$

$$\rho = \mathbf{r} - \mathbf{n}(\mathbf{n} \cdot \mathbf{r}), \quad P_i^0 = |(\rho_i^{\pm} - \rho) \cdot \mathbf{u}_i|, \quad P_i^{\pm} = |\rho_i^{\pm} - \rho|,$$

$$l^{\pm} = (\rho_i^{\pm} - \rho) \cdot \mathbf{l}, \qquad d = \mathbf{n} \cdot (\mathbf{r} - \mathbf{r}^{\pm}), \qquad R_i^{\pm} = \sqrt{(P_i^{\pm})^2 + d^2},$$

$$\mathbf{P}_i^0 = \frac{(\rho_i^{\pm} - \rho) - l^{\pm}\mathbf{l}}{P^0}, \quad R_i^0 = \sqrt{(P_i^0)^2 + d^2}.$$

Note that in (A.1), the first term within the square brackets is equal to zero if either $R_i^+ + l_i^+$ or $R_i^- + l_i^-$ is equal to 0. This is because P_i^0 equals zero. With the analytical evaluation of the integral in (A.1), we can compute the charge density distribution over a 3D surface by solving the set of linear equations in Equation (4.13).

APPENDIX B

Multiview Integration

Our approach to parametric geon-based shape representation starts with either single- or multi-view range data. Construction of multiview data of 3D objects is described in this section. Multiview data are produced by a three-step procedure, called multiple view integration [**48, 127, 97, 138, 31, 25**]. In the first step - *data acquisition* - the range data from different views are collected as viewer-centred data descriptions specified in each camera coordinate system. In the second step - *view registration* - a transformation between a camera coordinate system and the world coordinate system is calculated. In the last step - *view integration* - range data in each camera coordinate system are transformed into the world coordinate system and usually the redundant data seen in more than one view are removed. Here we present a simple and straightforward method for view integration.

Acquisition of multiview range images was accomplished with a laser range finder which scans objects supported by a turntable. The registration among images taken from different views was obtained by a method developed in our laboratory [**25**]. View integration was performed by using the view transformation, surface normals and residuals of the normal computation as follows:

ALGORITHM B.1.

(i) *Input images of view m= 1,2,...M.*

(ii) *For each view, compute the normals and the angles between normals and visual lines.*

(iii) **FOR** *each view m,*

  FOR: *each data point D(i,j) in the image,*

    FOR: *each successive view, n = m+1, m+2, ..., M,*

      (a) *Transform D(i,j) onto this view n.*

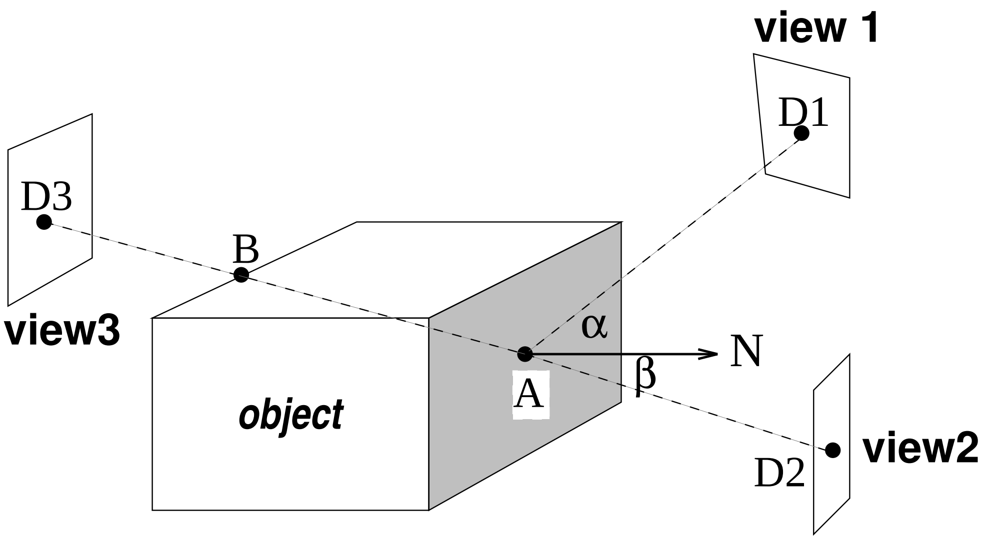


FIGURE B.1. Point A on the shaded object surface can be projected onto *view*1, *view*2 and *view*3 resulting in data points D_1, D_2 and D_3, respectively. Only D_1 and D_2 depict the same point(A) and are treated as redundant data. D_3 is not redundant with D_1 and D_2 since point A cannot appear in *view*3 due to object self-occlusion. The dashed lines are the projection lines.

(b) **IF** *the data D(i,j) is redundant with data in this view,* **THEN** *mark this D(i,j) and the corresponding data with RD (ReDundant).* **ELSE** *Mark this data NR (Not Redundant).*

END: *for each successive view.*

IF: *data are redundant,* **THEN** *select the best data in a specific view according to its normal and visual angle. Then mark this best data with NR.*

END: *for each data.*

END *for each view.*

(iv) *Convert data marked with NR in all views onto a common world coordinate system.*

The principle of selecting the best data is shown in Figure B.1. Point A generating a data point D_1 on the image plane in *view*1 can be *mathematically* projected on image planes in both *view*2 and *view*3 giving data points D_2 and D_3. If there exist data points at D_2 and D_3 on the original image planes, we can compute the world coordinates for these two points based on camera calibration. If the position of a point in the world coordinate system is spatially overlapped[1] by point A, we mark this data point in its image plane as a redundant one. In Figure B.1, D_2 is redundant with D_1. However, D_3 is not redundant

[1]Due to errors in the estimated transformation parameters, multi-view data for the same point in 3D do not exactly overlap. Here a threshold in distance is employed.

with D_1 and D_2 because point A cannot be *optically* projected onto the image plane in $view3$ due to object self-occlusion. The world coordinates for D_3 is at point B which is not overlapped with point A in 3D space. Given redundant data D_1 and D_2, the angles, α and β, between the surface normal N at point A and scan lines are examined. In general, if a surface point faces the rangefinder, and the angle between its normal and the scan line is small, the rangefinder obtains good reflection of the laser beams from the surface, and the quality of the image data is good. Thus, D_1, which gives a larger angle - or a smaller cosine value of β - is removed. If the cosine values of angles associated with a few data are very close, we keep the data point with the smallest residual resulting from the normal computation. In a few cases, both the cosine values and the residuals are very close; in this case we choose data collected at an earlier stage. The integrated data is expressed as a sequence of 3D points and used for 3D part-based representation.

APPENDIX C

Surface Triangulation

Surface triangulation is required by the charge density computation presented in Chapter 4. In this section, we describe in detail the surface triangulation of single-view range data. The charge density computation requires a triangular mesh to be tessellated on the complete surface of the object. However, the range data obtained from a particular view only reflect the visible surfaces, as shown in Figure 4.7. Thus it is impossible to perform the mesh tessellation based on the *actual shape* of the invisible parts of the object. In practice, we *artificially* construct a mesh on the invisible side in order to make up a closed triangular mesh. This permits us to compute the charge density. We note that the actual shape of the invisible surface only affects the absolute value of the charge density on the visible surfaces. The position of the extrema of the charge density distribution remains almost the same and thus it makes sense to construct an artificial mesh on the invisible surface. This argument is later justified by experiments.

Single-view range data are defined as a collection of M discrete samples of an underlying function $\mathbf{f} : \mathcal{I}^2 \rightarrow \mathcal{R}^3$

$$\mathbf{d}_i = \mathbf{f}(\mathbf{u}_i) \tag{C.1}$$

where $\mathbf{u}_i \in \mathcal{I}^2$ is the index of the 2D image grid, $\mathbf{d}_i = \{x_i, y_i, z_i\} \in \mathcal{R}^3$ is the 3D coordinate of a data point, $i = 1, ..., M$ is the index of the data and M is the total number of pixels in a range image. The object of interest can be segmented from its background by a simple thresholding operation. It is represented by data $\mathbf{d}_i, i = 1, ..., N, N < M$, as shown in Figure 4.7. It is noted that not all image pixels contain data since the object size is smaller than the image size. The closed triangular mesh for the object is composed of three patches of triangular meshes, as shown in Figure 4.8. The first, called the *top patch*, is obtained by triangulating the range data on the visible surface. The second, called the *bottom patch*, is planar, and is actually the (spatial) projection of the top patch onto an arbitrary plane

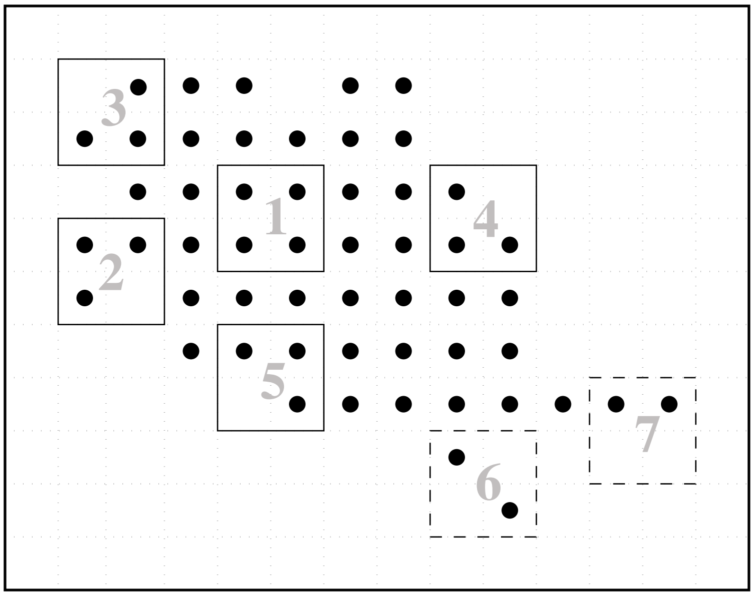


FIGURE C.1. Pixel relations in a 2×2 window.

perpendicular to the Z axis. These two patches are illustrated in Figure 4.8 (a). The third one , called the *side patch*, fills the gap between the top and the bottom patches, as shown in (b). The complete closed triangular mesh in Figure 4.8 (c) is obtained by merging the patches in (a) and (b).

1. Top and Bottom Patch Construction

We construct top and bottom patches based on the range image grid. Since the image grid is rectangular, the number of local spatial relations among data points is limited. Thus, we can enumerate all possible relations and derive a rule-based algorithm to tessellate these two patches. Consider a 2×2 window in a range image grid. We specify the left top pixel as the currently-considered pixel. For this pixel, there are four possible spatial configurations [1] for making triangles, as shown in Figure C.1. They are in Windows $1, 2, 4$ and 5. In addition, there may be no data point at the top left pixel but exits one at the top right, as indicated in Window 3. This configuration is also considered. Triangles can be constructed for each configuration. We use these five arrangements to tessellate the top and bottom surfaces by detecting the specific data configuration and making the appropriate triangles. The algorithm, where the boarder effects are not considered, is as follows:

ALGORITHM C.1.

[1] The number of pixels in the window must be either three or four in order to make a triangle. The data configurations in Windows 6 and 7 in Figure C.1 are invalid and no triangle will be formed.

(i) *Start from I(1,1),*

(ii) *Scan the image I(i,j), i,j = 1,...,N, and consider a* 2×2 *window positioned at (i,j),*

(iii) **If** *I(i,j)* $\neq$ *0,*

If: *I(i+1,j), I(i,j-1), I(i+1,j+1) are not equal to zero*[2], *make two triangles using* {*I(i,j), I(i+1,j), I(i,j+1)*} *and* {*I(i,j+1), I(i+1,j+1), I(i,j+1)*}*, respectively;*

Else: *if I(i+1,j), I(i,j+1) are not equal to zero, make a triangle using* {*I(i,j), I(i+1,j), I(i,j+1)*}.

Else: *if I(i,j+1), I(i+1,j+1) are not equal to zero, make a triangle using* {*I(i,j), I(i,j+1), I(i+1,j+1)*}.

Else: *if I(i+1,j), I(i+1,j+1) are not equal to zero, make a triangle using* {*I(i,j), I(i+1,j), I(i+1,j+1)*}.

Else *if* $I(i+1,j), I(i,j+1), I(i+1,j+1)$ *are not equal to zero, make a triangle using* {*I(i+1,j), I(i,j+1), I(i+1,j+1)*}.

(iv) **If** *I(N,N) has been visited, stop;* **else** *goto* (ii).

When constructing the bottom patch, we must specify the separation between the top and bottom patches. Note that the distance between these two patches does not crucially affect the position of the extrema of the charge density distribution on the visible surface. To illustrate this, we define a parameter ψ as follows:

$$\psi = \frac{z_{plane} - z_{max}}{z_{max} - z_{min}}. \tag{C.2}$$

As demonstrated in Figure C.2, ψ is the ratio of the distance from the data point having the largest Z value to the bottom planar patch and the spatial range of all data points in the Z direction. Figure C.3 shows the computed charge density distributions on the visible surface of an object when (a) $\psi = 10\%$, (b) $\psi = 50\%$ and (c) $\psi = 100\%$. The gray levels indicate charge densities, which are normalised in the range between 0 (darkest intensity) and 255 (white). It can be seen that the minima of the charge density distributions are not changed in these cases. In other words, the artificially constructed mesh on the invisible surface does not seem to affect the charge density extrema on the visible surface. A similar strategy has been proposed for generating a closed surface in 3D space for a diffusion-based shape analysis [**151**].

[2] In range images, if there are no data points at $I(i,j)$, it is customary to set $I(i,j) = 0$.

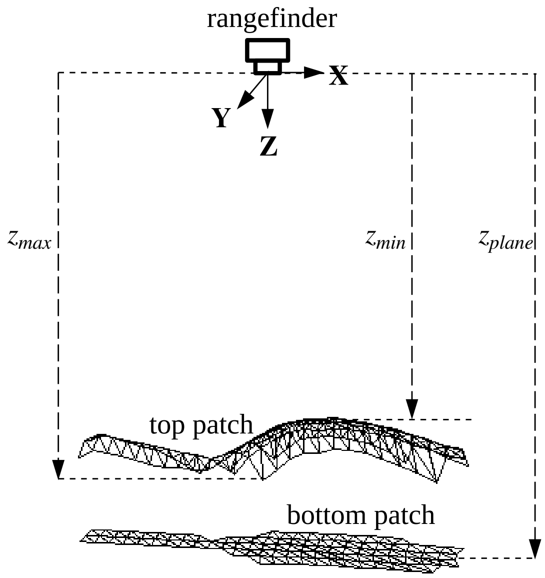


FIGURE C.2. Configuration of the top and the bottom patches.

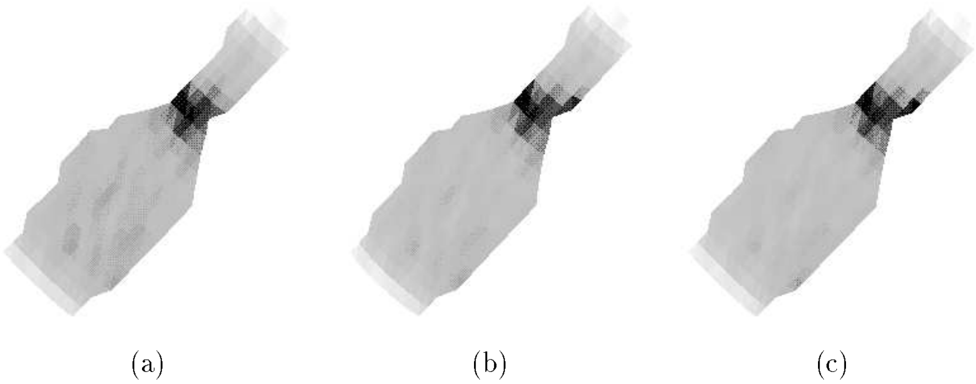


FIGURE C.3. Charge density distributions on visible surfaces. The charge density is normalised between gray levels 0 and 255. The brightest and darkest intensities indicate the highest nd lowest charge densities, respectively. When (a) $\psi = 10\%$, (b) $\psi = 50\%$ and (c) $\psi = 100\%$, the positions of the extrema of the charge density distribution on the visible surface are almost not affected.

2. Side Patch Construction

The characteristic of the side surface is that it is a cylindrical surface, which can be developed into a plane. Thus, this surface can be also triangulated in a planar domain. We first generate points of triangle vertices on the plane and then create the corresponding triangles.

Let the length of a triangle side along the boarder of the top patch (see Figure C.4) be $s_i, i = 1, ..., K$. In order to ensure that the triangles on all patches have more or less the

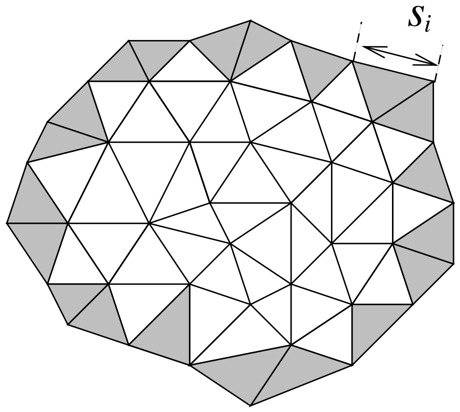


FIGURE C.4. Triangles on the boundary of the top patch are shaded. The length of a side on the boundary is s_i.

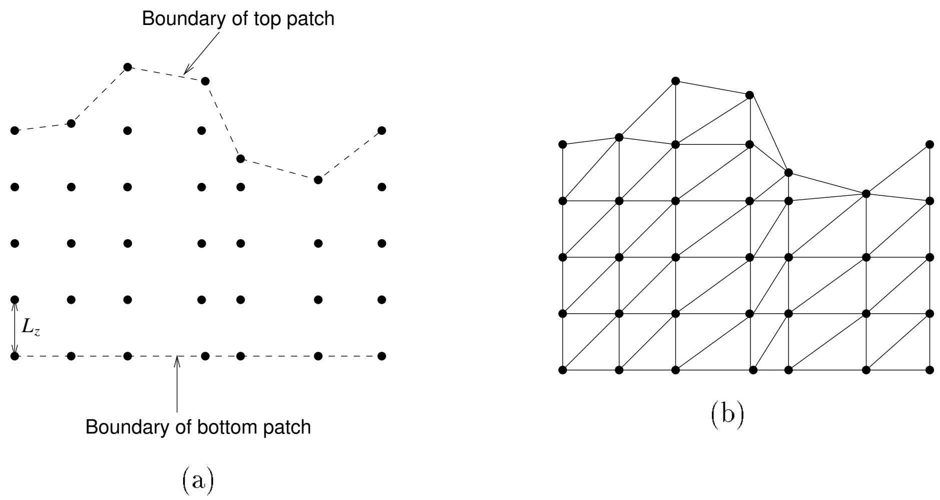
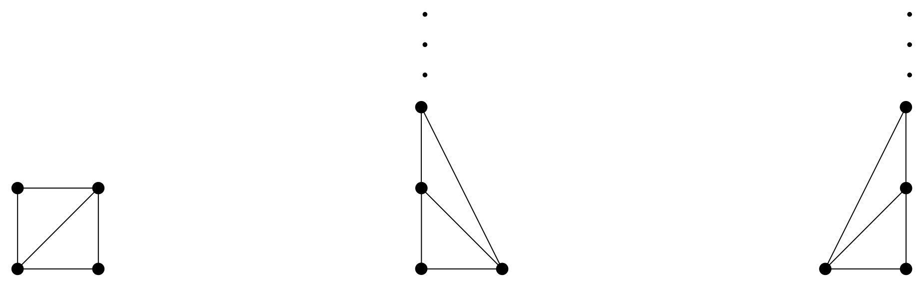


FIGURE C.5. The side patch of triangulated range data. (a) Vertices of triangles on the side patch, which has been developed in a plane. The distance between rows of data is set to L_z. The vertices linked by the dashed lines are on the boundaries of the top or bottom patches. (b) Triangles on the side patch.

same size, we choose the average of triangle side lengths on the top patch as the length of a triangle side on the side patch in the direction of Z as follows:

$$L_z = \frac{1}{K}\sum_{1}^{K} s_i$$

That is, the interval between two data points along the Z direction is L_z, as shown in Figure C.5. Next we determine the actual locations of triangle vertices on the side patch. Along the straight lines connecting the nodes on the bottom and top patches, we position vertices every L_z in a bottom-to-top order, as shown in Figure C.5.



(a) (b) (c)

FIGURE C.6. Configuration of triangle vertices on the side patch. (a) A common configuration having four vertices. (b) and (c) are special cases occurring at the top of a side patches. The circles and the dashed lines indicate the possible vertices and triangles.

In a similar fashion to the top and bottom patch triangulation, all possible local configurations of triangle vertices on the side patch are used for side patch triangulation. There are only three types of configurations of vertices of triangles as shown in Figure C.6. (a) represents the most common relation appearing at the bottom and middle portions of the side patch. Here there are four vertices, two on each column. (b) and (c) indicate the cases occurring at the top of the side patch. There is one vertex in one column and more than one vertex on the other. When (a) appears, two triangles are made. When (b) or (c) appear, the number of triangles is determined according to the number of vertices on the column containing more than one vertex. Figure 4.8 gives a complete example of a surface triangulation for the single-view range image in Figure 4.7.

APPENDIX D

Part Decomposition

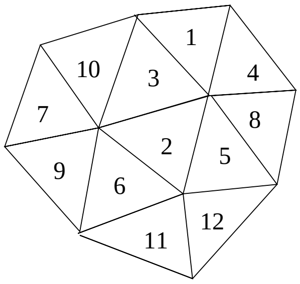
This section describes a technique, in detail, for decomposing a whole object into parts. Since the object surface is represented by a closed triangular mesh, part segmentation is actually accomplished by decomposing the mesh into several components based on the simulated charge density distribution. Each component is a connected triangular mesh which is a subset of the closed triangular mesh of the whole object. For a single-view range image, only that portion of the triangular mesh which represents the visible surface of the object is considered. This method is based on a so-called *Direct Connection Graph* (DCG), which serves as a specific coordinate system defined on the triangular mesh. We will first describe the method of DCG construction and then give the part decomposition algorithm.

1. DCG Construction

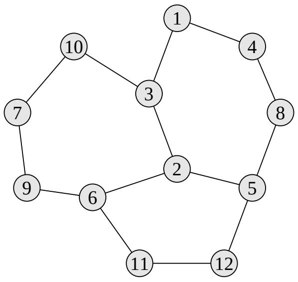
As defined in Chapter 4, Direct Connection Graph is one kind of representation of the triangular mesh. Its nodes represent the triangles and its branches represent the connections between a node and its direct neighbours. Figure D.1 (a) shows a triangle mesh and (b) shows its DCG.

We denote an array of triangles by $T_i = \{v_{ij}, j = 1, 2, 3\}, i = 1, ..., N$. Here i is the index of triangles and j is the index of triangle vertices. v_{ij} is the jth vertex of the ith triangle. N is the number of triangles. We represent the DCG of a triangular mesh by an 2D array $D[i, j]$, where $i = 1, ..., N; j = 1, ..., 5$. $D[i, 1]$ stores all triangles, $T_i, i = 1, ..., N$ in the mesh. $D[i, 2], D[i, 3], D[i, 4]$ consist of a list of direct neighbours of the triangle given in $D[i, 1]$. $D[i, 5]$ indicates the total number of the direct neighbours of $D[i, 1]$. For a closed mesh, $D[i, 5] = 3$; and for an open mesh, $D[i, 5]$ could be either 2 or 3. An example of an open triangular mesh and the process of its construction is given in Figure D.1[1]. Since the mesh in (a) is not closed, the triangles at the boundary have only two direct neighbours.

[1]Here we only show the DCG of the first six triangles. The complete DCG for this mesh contains 12 triangles.



(a)



(b)

T1	T3			1
T2				
T3	T1			1
T4				
T5				
T6				

(c)

T1	T3	T4		2
T2				
T3	T1			1
T4	T1			1
T5				
T6				

(d)

T1	T3	T4		2
T2	T3			1
T3	T1	T2		2
T4	T1			1
T5				
T6				

(e)

T1	T3	T4		2
T2	T3	T5		2
T3	T1	T2		2
T4	T1			1
T5	T2			1
T6				

(f)

T1	T3	T4		2
T2	T3	T5	T6	3
T3	T1	T2		2
T4	T1			1
T5	T2			1
T6	T2			1

(g)

T1	T3	T4		2
T2	T3	T5	T6	3
T3	T1	T2	T10	3
T4	T1	T8		2
T5	T2	T8	T12	3
T6	T2	T9	T11	3

(h)

FIGURE D.1. Construction of the Direct Connection Graph (DCG). (a) A triangular mesh. (b) The first step. Column 1 shows the list of triangles in the mesh. Column 2 to column 4 show the first to the third neighbour, respectively, of the triangle given in the first column. The last column shows the number of direct neighbours so far. (c) to (f) indicate successive steps of DCG construction. (7) shows the final result for T_1 to T_6.

For an example, in the case of T_1, $D[1,5] = 2$. The algorithm for constructing this array is described as follows:

ALGORITHM D.1.

(i) *Load all triangles in the mesh into* $D[i,1]$ *and set* $D[i,5] = 0$.

(ii) *Start with T_1, for each T_i,*

   (a) *Search downward in the first column for a direct neighbour of T_i, based on Definition 4.3. When a direct neighbour T_i of T_j is found,*

      (i) *Put T_j in the neighbour list of T_i,*

      (ii) *Put T_i in the neighbour list of T_j,*

      (iii) *Increase both $D(i,5)$ and $D(j,5)$ by 1.*

   (b) *From T_j, repeat the search procedure in (a) until T_N is reached.*

(iii) *End for each i.*

Since the DCG provides an explicit relationship between individual triangles on the surface of the object, it serves as a convenient coordinate system over the object surface. It permits the tracing of the part boundaries on the triangular mesh without employing a voxel-based coordinate system. This significantly reduces the required memory space for describing the object and increases the computational speed.

2. Finding Part Boundaries

The transversality principle states that when two surfaces intersect, they intersect transversally with probability one[**11**]. This means that the tangent planes to the two intersecting surfaces are of different orientations at *all* points where the surfaces intersect (see Figure 2.5). Following this, we have assumed that a part boundary is explicitly defined by deep surface concavities. For a complete object, the part boundary is a *closed* surface. This ensures that the decomposition algorithm will be able to segment a part from the rest of the object. The assumption also provides a stopping criterion for the boundary tracing procedure. Since the part boundary is located at local charge density minima, it can be traced along the 'valley' of the charge density distribution. We note that the tracing algorithm applied to a closed mesh is slightly different from that applied to an open mesh. The latter is constructed for representing single-view range data. Since the part boundary on an open mesh is not closed, the stop criterion for tracing on the closed mesh is modified. The new criterion is that when the tracing algorithm reaches a triangle on the boundary of the open mesh, it stops. The triangle on the mesh boundary is defined as the one with only two direct neighbours. Since two algorithms are very similar, in the following, we only describe the algorithm for tracing the closed triangular meshes.

The algorithm examines the charge density on all triangles to find a starting triangle for tracing each boundary. A starting triangle must satisfy the following conditions:

(i) It must be a *concave* extremum; that is, its charge density must be a local minimum.

(ii) It must be located at a *deep* concavity. Thus the charge density on the triangle must be lower than a preselected threshold[2].

(iii) It and its neighbours must not have been visited before. This ensures that the same boundary will not be traced again.

Beginning at the starting triangle, the algorithm proceeds to the neighbour with the lowest charge density. During the tracing procedure, all triangles detected on the boundary are marked. The marked ones will not be checked again and eventually will be deleted from the DCG. The process continues until it returns to the starting triangle. As a result of the assumption stated at the beginning of this section, this means that all triangles on this part boundary have been visited. Next the algorithm finds a new starting triangle and traces another boundary. It repeats the same tracing procedure, and finally stops when the charge density at a starting triangle is higher than the preselected threshold. After all triangles on part boundaries have been found, the nodes of the DCG representing these triangles are deleted. Thus the original DCG is now divided into a set of disconnected subgraphs, as shown in Figure 4.10 (c). Physically the object has been broken into parts. Each object part can be obtained by applying a component labelling algorithm to a subgraph of the DCG. The resultant part is ready for part model recovery.

In the following, we describe the boundary tracing algorithm in detail. The variables used in the algorithm are defined as follows:

ρ_{min}: the lowest charge density over all triangular patches, i.e., triangles.

$\rho(j)$: the charge density of the jth patch

ρ_t: the threshold for a starting patch

B_i: A set of triangles that belong to boundaries of parts, where i is the number of boundaries.

T_{start}: A set of triangles with which the boundary tracing procedure starts.

$NewPatch$: A flag indicating whether a new patch on the part boundary is found ($NewPatch = 1$) or not ($NewPatch = 0$).

$PminiCharge$: the minimum charge density of a direct neighbour patch.

$MiniCharge$: the charge density for T_{start}.

During the tracing, we label all patches as follows:

P_0: the patches belong to B_i

[2]This threshold determines when an object should not be decomposed any further. If the charge density at a starting triangle is greater than this threshold, we assume that all boundary points have been found. The selection of the threshold depends on a priori knowledge of the surface concavity and there is no universal rule for selecting it. Obviously, the higher the threshold, the more segmented parts will be found . Currently we choose 120% of the lowest charge density on the object surface as the threshold.

P_1: the patches have been visited

P_2: the neighbours of T_{start} which have been visited

P_3: the patches that have not been visited

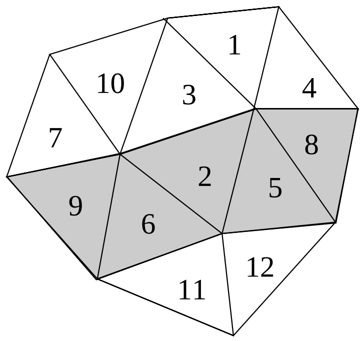
The boundary tracing algorithm is described as follows:

ALGORITHM D.2.

(i) *Mark all triangular patches as* P_3.
(ii) **Loop** *for each part boundary contour* i
  (a) *Initially set* $MiniCharge = \rho_t$.
  (b) **Loop** *for* j
    (i) *Find a patch* T_j *at a local charge density minimum.*
    (ii) **If** $\rho(j) < MiniCharge$, *assign* $\rho(j)$ *to* $MiniCharge$ *and mark* T_j *as* T_{start}.
  (c) **end** *of loop*
  (d) **if** $MiniCharge < \rho_t$
    (i) *set all neighbours of* T_{start} *to* P_2.
    (ii) **Loop** *for* k, *each neighbour of* T_{start}
      (A) **If** NB_k *is a* P_3, *assign the charge density of* NB_k *to* $PminiCahrge$, *set* $NewPatch = 1$, *mark this patch as* T_n *and* **escape** *the loop for* k.
    (iii) **end** *of loop for* k.
    (iv) **if** $NewPatch = 1$, *mark* T_n *as* P_1.
    (v) **Loop** *for* k
      (A) **If** $T(k)$ *is a* P_3, *mark it as* P_1.
      (B) **If** $\rho(k) < PminiCharge$, *assign* $\rho(k)$ *to* $PminiCharge$.
    (vi) **End** *of Loop for* k
    (vii) *Mark* $T(k)$ *as* P_0.
    (viii) **If** $T(k)$ *is a neighbour of* T_{start} *or a neighbour of* B_i, **stop** *tracing the contour* i.
(iii) **End** *of loop for* i

3. Classifying Patches into Parts

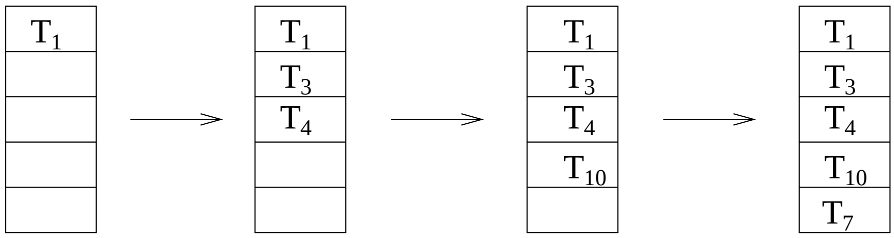
In this section, we describe a component labelling algorithm for a triangular mesh. We would like to obtain all triangular patches in a particular subgraph of the DCG, which



(a)

patch	neighbors		
T_1	T_3	T_4	
T_2	T_3	T_5	T_6
T_3	T_1	T_2	T_{10}
T_4	T_1	T_8	
T_5	T_2	T_8	T_{12}
T_6	T_2	T_9	T_{11}
T_7	T_9	T_{10}	
T_8	T_4	T_5	
T_9	T_6	T_7	
T_{10}	T_3	T_7	
T_{11}	T_6	T_{12}	
T_{12}	T_5	T_{11}	

(b)



(c)

FIGURE D.2. Component labelling. (a) a DCG containing a part boundary, as indicated by the shaded patches. (b) A DCG array. (c) the Updated PART array from left to right. The indexes in the rightmost array indicate patches that belong to the same part.

represents one object part. The existing component labelling algorithms [**107, 106**] are mainly for 2D binary images, in which each pixel has four direct neighbours. The difference between a 2D image and a triangular mesh is that the latter is described as a 1D array with three explicitly specified neighbours. An example of this algorithm is shown in Figure D.2. The algorithm finds each part in sequence and is illustrated as follows:

ALGORITHM D.3.

(i) **Loop** *for each object part i*

(a) *Open a temporary space, called PART, to store the triangle index for the ith part and initialise a counter for the number of triangles on each part.*

(b) *Starting with the first triangle in the DCG array, find the first unchecked triangle which is not on a part boundary. Add it to PART and mark it in the DCG array as a checked triangle. Specify a pointer in the DCG array, pointing to this triangle.*

(c) *In the DCG array, add the neighbours of this triangle, which (1) are not on a part boundary and (2) have not already been in PART, into PART.*

(d) *In the DCG array, move the pointer down one element.* **Repeat** *step (c)* **until** *no more triangles can be either checked or are on boundaries.*

(e) *Mark the triangles in PART as being on the same part i. Thus triangles belonging to part i have been found.*

(ii) **End** *of Loop for i*

(iii) **Repeat** *from step (a)* **until** *all triangles have been checked.*

The result of this algorithm is several lists of triangles. Each list contains triangles which belong to the same object part. These triangle lists are used for part model identification.

Document Log:

Manuscript Version 0

Typeset by $\mathcal{AMS}$-LaTeX — 24 April 1996

Kenong Wu

Centre for Intelligent Machines, McGill University, Montréal, Québec, Canada

E-mail address: wu@cim.mcgill.ca

Typeset by $\mathcal{AMS}$-LaTeX