Exploring Three-Dimensional Objects by Controlling the Point of Observation

Kiriakos N. Kutulakos

Technical Report #1251

October 1994

EXPLORING THREE-DIMENSIONAL OBJECTS BY CONTROLLING THE POINT OF OBSERVATION

by

KIRIAKOS NEOKLIS KUTULAKOS

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

(Computer Sciences)

at the

UNIVERSITY OF WISCONSIN – MADISON

1994

Abstract

In this thesis we study how controlled movements of a camera can be used to infer properties of a curved object's three-dimensional shape. The unknown geometry of an environment's objects, the effects of self-occlusion, the depth ambiguities caused by the projection process, and the presence of noise in image measurements are a few of the complications that make object-dependent movements of the camera advantageous in certain shape recovery tasks. Such movements can simplify local shape computations such as curvature estimation, allow use of weaker camera calibration assumptions, and enable the extraction of global shape information for objects with complex surface geometry. The utility of object-dependent camera movements is studied in the context of three tasks, each involving the extraction of progressively richer information about an object's unknown shape: (1) detecting the occluding contour, (2) estimating surface curvature for points projecting to the contour, and (3) building a three-dimensional model for an object's entire surface. Our main result is the development of three distinct active vision strategies that solve these three tasks by controlling the motion of a camera.

Occluding contour detection and surface curvature estimation are achieved by exploiting the concept of a *special viewpoint*: For any image there exist special camera positions from which the object's view trivializes these tasks. We show that these positions can be deterministically reached, and that they enable shape recovery even when few or no markings and discontinuities exist on the object's surface, and when differential camera motion measurements cannot be accurately obtained.

A basic issue in building three-dimensional global object models is how to control the camera's motion so that previously-unreconstructed regions of the object become reconstructed. A fundamental difficulty is that the set of reconstructed points can change unpredictably (e.g., due to self-occlusions) when *ad hoc* motion strategies are used. We show how global model-building can be achieved for generic objects of *arbitrary* shape by controlling the camera's motion on automatically-selected surface

tangent and normal planes so that the boundary of the already-reconstructed regions is guaranteed to "slide" over the object's entire surface.

Our work emphasizes the need for (1) controlling camera motion through efficient processing of the image stream, and (2) designing provably-correct strategies, i.e., strategies whose success can be accurately characterized in terms of the geometry of the viewed object. For each task, efficiency is achieved by extracting from each image only the information necessary to move the camera differentially, assuming a dense sequence of images, and using 2D rather than 3D information to control camera motion. Provable correctness is achieved by controlling camera motion based on the occluding contour's dynamic shape and maintaining specific task-dependent geometric constraints that relate the camera's motion to the differential geometry of the object.

Acknowledgements

I am greatly indebted to my advisor, Professor Chuck Dyer. His continuous guidance and encouragement have been instrumental to this work. He taught me how to do research in computer vision and how to best communicate my ideas. Professor Dyer provided an excellent work environment that was productive, friendly, and intellectually stimulating. I consider myself very fortunate in having the opportunity to work with him. I would also like to thank him for financially supporting my work.

Professor Vladimir Lumelsky has been another source of inspiration and advice during the last three years of my graduate studies. He introduced me to the field of robotics and generously provided constructive criticism at several stages of my work. He also broadened my perspective beyond the field of computer vision by helping me apply the work in this thesis to problems in robotic motion planning.

Special thanks are due to Professors Roland Chin, Rajit Gadh, and Jude Shavlik for taking the time to review this work as members of my thesis committee, and to Professor Amir Assadi for his input as a guest committee member. I have also benefited from discussions of research ideas with several former and current members of the vision and robotics groups, including Brent Seales, Mark Allmen, Steve Seitz, Dan Reznik, and Sanjay Tiwari.

I would like to thank all my friends, in the US and in Greece, for their love and support. I especially want to thank Sean Selitrennikoff for providing lots of necessary distractions during my first years in Madison; Dionisis Pnevmatikatos for being a great friend for the last five years and for being an endless source of friendly arguments; and Thanasis Diplas, Manolis Tsangaris, Spiros Kontogiorgis, Dimitris Tsangaris, Dimitris Fotakis, Thanos Tsiolis, Lakis Karmirantzos, Giannis Christou, Minos Garofalakis, Andreas Moshovos, and the rest of the members of the "Greek gang" of Madison who made my stay in Madison lots of fun.

I have no words to express my gratitude toward my wonderful fiancee, Sarah. Her love and affection,

her spirit of optimism, her enthusiasm, and her belief in me were my only sources of strength during the many difficult stages of life as a graduate student. Finally, I would like to dedicate this thesis to my parents, Michael and Bogusha. They encouraged me to pursue my aspirations and go to graduate school even though this meant we would be thousands of miles apart. Without their love and support none of this work would have been possible.

Contents

List of Figures

Chapter 1

Introduction

Visual object exploration is the process of performing controlled camera movements to extract an unknown object's geometrical and physical properties from the images obtained, or to inspect an object's surface. This thesis focuses on one aspect of the general visual object exploration problem: How can controlled changes of a camera's *viewpoint*, i.e., changes in the relative position of a camera and an unknown object, be used to infer properties of the object's three-dimensional shape. Among others, properites of interest include surface convexity and curvature.

Much emphasis in computer vision has been placed on exploring an object's shape using predefined viewpoint changes that do not take into account the object being explored (e.g., moving the camera along straight lines parallel to the image plane to simplify shape computations [32]). However, the unknown geometry of an environment's objects, the effects of self-occlusion, the depth ambiguities caused by the projection process, and the presence of noise in image measurements are just a few of the complications that make *object-dependent* viewpoint changes advantageous in certain shape recovery tasks. For example, when measuring the height of a coffee cup, near-top views should generally be avoided since the foreshortening of the cup's wall can lead to unreliable measurements; moving to a near-side view, on the other hand, can be very useful for this purpose because this motion cancels out the effect of foreshortening, making height computations more reliable and, possibly, simpler. As a second example, when using images taken by a moving camera to build a three-dimensional model of a complicated object with dents, holes, and protrusions, parts of the object may be missed due to occlusion if the camera moves in some predefined way in front of the object; reaching viewpoints where those

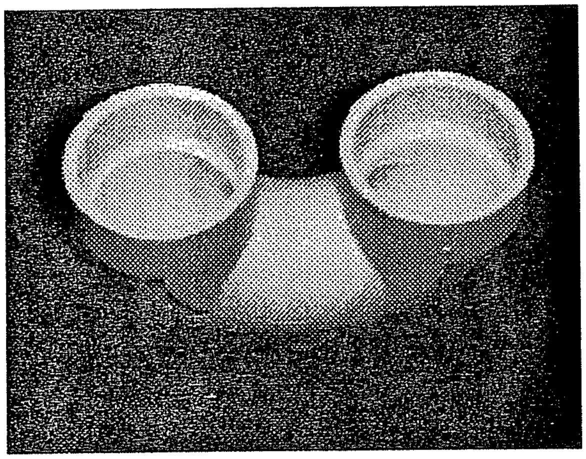


Figure 1: Example of an object of interest in this thesis.

parts are visible can lead to the construction of more accurate representations of the object's shape.

Object-dependent viewpoint changes must by driven by image data in order to be useful in exploring unknown objects, since no *a priori* shape information is available about the objects. Very little is currently known about how the cooperation of vision processing and viewpoint control can be used for visual object exploration either in shape recovery or in inspection tasks. In this thesis we show how this cooperation can be exploited for extracting geometric properties of objects that contain curved surfaces of complex shape, i.e., they can even contain multiple non-convex regions that occlude each other (Figure 1).

While psychologists have long advocated the active nature of perception [64–66, 87], the combination of vision and action in order to simplify and enhance vision processing is a relatively new development in computer vision research [9, 11, 12, 14, 16, 154]. A key idea in *active vision* is that there are computational advantages in interleaving the control of a camera's parameters with vision processing when operating in a three-dimensional, unstructured world. For example, Aloimonos *et al.* [9] showed that by controlling camera parameters such as the direction of gaze and the vergence of a stereo camera pair in task-driven and image-driven ways, mathematically ill-posed problems can become well-posed, geometric computations can be linearized, solutions can become stable against noise, and assumptions about the environment such as object smoothness can be relaxed. Partly motivated by the behavior of biological organisms, a number of active vision paradigms have been introduced to emphasize the close connection that exists between the tasks of animate organisms (e.g., manipulating objects, navigating, pursuing), the actions they have to take in order to perform them, and the vision processing they use [6, 15, 124, 144, 165]. The recent development of "anthropomorphic" vision platforms [47, 56, 100, 119, 123] equipped with rotating "necks," binocular cameras that can automatically change vergence, focus, and zoom, and the advances in general-purpose hardware design, have led to a number of demonstrations of the feasibility, generality, and power of integrating vision and action (e.g., see the collections of papers in [7, 10, 30, 73, 155, 156]).

Although a growing literature acknowledges the need for image-driven and task-driven control of a camera's parameters (e.g., fixating to keep a moving object in the field of view [47], verging and focusing to keep a moving object within the depth of field [125]), the use of image-driven viewpoint control for exploring unknown objects has received limited attention [112, 149].

The goal of this thesis is to study how image-driven viewpoint control can be used in the visual exploration of curved objects. Its main contribution is the development of a framework whereby vision processing and viewpoint control are formulated as two continuous, interdependent processes that are tightly coupled and occur simultaneously. We show that such a coupling has a significant impact on the shape recovery tasks that can be performed and on the computations needed to perform them. In particular, we show that the tight coupling of viewpoint control and vision processing can lead to localized processing of images, improved resistance to noise, and the ability to explore curved objects with complex surface geometry.

1.1 Motivation and Problems Considered

Visual object exploration can be used to autonomously extract information about the shape of an unknown object in order to manipulate it, inspect its surface, and perform shape measurements. Such a process is particularly useful when humans cannot be used to perform the task directly. Transmission delays, the difficulty of tele-operation and remote manipulation, and the time-critical nature of operating in hazardous, unstructured environments (space, undersea, or toxic) emphasize the need for high degrees of autonomy.

Visual object exploration for three-dimensional model-building is becoming increasingly important in the context of computer graphics and manufacturing as well. The realism of computer-generated images is contingent upon the realism and accuracy of the object models used as input. The recent interest in virtual reality and the design of systems for interactively navigating within three-dimensional "virtual worlds" [111] brings up the question of how models of the objects furnishing these worlds can be acquired. A method for automating this model acquisition process is clearly desirable, especially for objects with complex surface geometry (e.g., natural objects such as orchid flowers) for which manual construction can be time-consuming. In manufacturing, construction of new Computer-Aided Design (CAD) models can be simplified by "reverse engineering" already existing industrial parts, reducing the need for manual construction or use of time-consuming Coordinate Measuring Machines [150]. Currently, no general method exists for automating the model acquisition of complicated objects (e.g., with dents and holes), although some initial studies have been reported [117, 177] and commercial products for building models from real objects have appeared [166].

Visual object exploration does not have a single purpose. Rather, the geometric properties extracted about an object are defined by the particular task. Our claim is that the proposed framework of tightly coupling viewpoint control and vision processing is a general one. To demonstrate this we apply our framework to three shape recovery tasks, each involving the extraction of progressively richer information about an object's surface geometry:

- Determining the silhouettes of individual objects in an image, which can be used to grasp one of the objects
- Computing surface curvature at individual points on an object's surface
- Building a three-dimensional model of an object's entire surface

To solve these tasks we study the relationship between the differential geometry of a surface (e.g., curvature, tangent planes, the nesting configuration of parabolic curves), the surface's projected shape, and the manner in which this projected shape changes when viewpoint is controlled relative to the surface itself. The result is a collection of task-driven and image-driven exploration strategies for controlling viewpoint that have distinct functionalities and impose progressively greater demands on the resources of the vision system.

While the purpose of visual object exploration varies from task to task, the ability to explore curved objects of unspecified geometry can be very valuable. Even objects constructed from simple primitives such as spheres can exhibit complicated geometries and topologies, causing partial or complete occlusion of some of these primitives. Consequently, *a priori* assumptions about the structure of an image (e.g., no self-occlusions) are too restrictive in the context of visual exploration.

The approach taken in this thesis is to develop visual exploration strategies that have *provable* properties. In particular, the first two of the tasks we consider are studied in the context of arbitrarily-shaped curved objects, and a precise geometrical analysis of the conditions under which the strategies succeed and fail is given. Furthermore, our study of the third task shows how to combine viewpoint control and vision processing in a way that can provably guarantee the success of the visual exploration process for curved objects of "almost arbitrary" shape: The developed strategy is shown to provably achieve the task for arbitrary *generic* objects, an object class whose members can approximate arbitrarily closely any curved object with no surface discontinuities.

1.2 Overview of the Approach

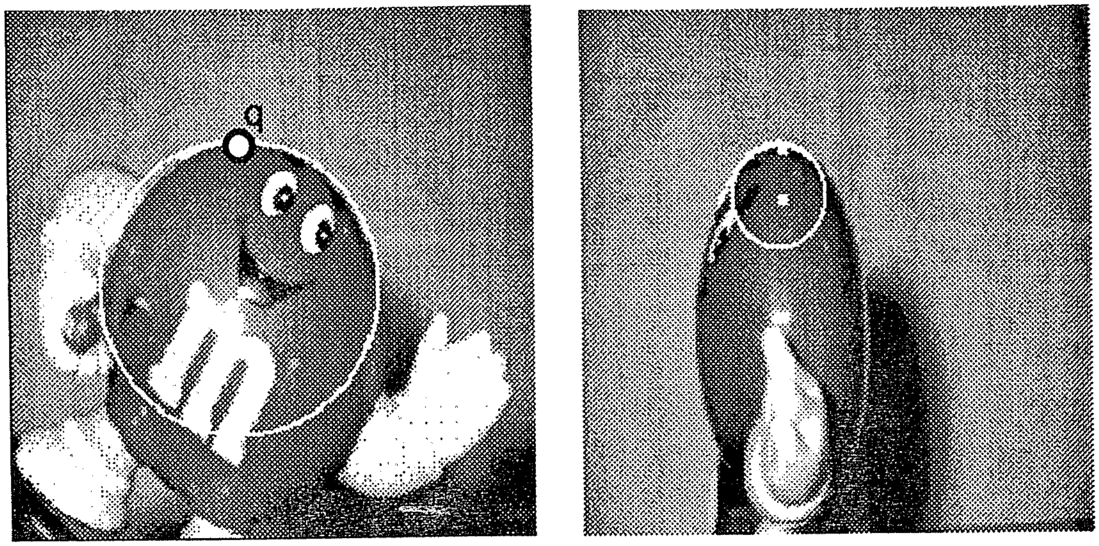
The main question studied in this thesis is how image-driven viewpoint control can be exploited to infer properties of an object's three-dimensional geometry. A key issue is what image data to use to extract shape information and to control viewpoint. Flexibility and generality can only be achieved if the image features used have a strong connection to the shape of the viewed object. Many different types of features are potentially useful for this purpose, including projected surface texture, markings, discontinuities, shadows, specularities, and silhouettes, to name just a few. Among these, texture, markings, and discontinuities have proved to be extremely valuable for accurately recovering an object's shape as a set of rigid 3D points, even when the camera undergoes completely arbitrary motion [162]. Unfortunately, relying on such image features can make visual exploration difficult for curved objects with sparsely distributed markings and discontinuities. Our work is specifically targeted toward exploring curved objects with little texture and few or no markings and discontinuities on them.

In our framework, visual exploration of curved objects is performed using the *occluding contour* [91] both to control viewpoint and to recover shape information. Roughly, the occluding contour can be viewed as a generalization of the silhouette; it corresponds to the depth discontinuities in an object's projection. The occluding contour is ideal as an input to the exploration of curved objects because its shape is highly constrained by the surface giving rise to it [116, 136] and can be recovered from images [170]. All of the work in this thesis revolves around the study of relationships between viewpoint control, surface shape, and their effect on the shape of the occluding contour.

The proposed *continuous, contour-driven viewpoint control* framework has three characteristic features: (1) Viewpoint control and vision processing are modeled as continuous, dynamic processes that occur simultaneously, (2) viewpoint control is driven directly by the image data, and in particular, by the shape of the occluding contour, and (3) the exploration strategies are based on the differential geometry of surfaces and have provable properties. Efficiency is achieved by changing viewpoint in a continuous fashion, assuming a dense sampling of images, relying on 2D rather than 3D shape information to control viewpoint, and extracting from each image only the information necessary to move the viewpoint differentially. Provable correctness is achieved by maintaining specific task-dependent geometric constraints between the viewpoint's motion and the object itself. Figure 2 gives an example of a simple exploration strategy for computing surface curvature, which is studied in Chapter 5.

When viewpoint is controlled in a continuous fashion, the occluding contour deforms. The key idea in our approach is that continuous viewpoint control allows us to design visual exploration strategies that constrain the deformation of the occluding contour in a task-specific manner by controlling the viewpoint's motion relative to the object. In this formulation, the main question posed for a given shape recovery task is: How should the viewpoint's motion be controlled so that the constrained deformation of the occluding contour solves the task? Our claim is that this formulation can be very useful for solving shape recovery tasks. For example, surface curvature can be computed under weaker assumptions than those required by existing methods, and tasks such as building a model for an object's entire surface can be provably achieved for objects with complex surface geometry.

In this thesis we show that we can force the contour's constrained deformation by controlling viewpoint directly from the image data, i.e., without first recovering 3D shape information about the object. Furthermore, we show that by basing the dynamic feedback loop that controls viewpoint on the occluding contour's deforming shape, this constraining process can be performed and analyzed locally in both space and time, even when performing global shape recovery tasks (e.g., building a model for



(a) (b)

Figure 2: We show in Chapter 5 that surface curvature for a point projecting to the silhouette becomes trivial to compute from a special viewpoint which, for point q in (a), corresponds to the side view of the toy. Results from the differential geometry of surfaces tell us that we can indeed reach that viewpoint through a continuous, contour-driven viewpoint control process: The viewpoint is reached from (a) by rotating viewpoint on a plane perpendicular to the image and tangent to the contour at q (i.e., a horizontal plane), *until* the curvature of the occluding contour at its uppermost point is maximized, as shown in (b).

an object's entire surface). This leads to the design of viewpoint control strategies that have minimal computational requirements, perform local processing in each image, and have provable properties.

Figure 3 shows the architecture of our framework. It consists of two computational processes that occur simultaneously: (1) The vision processing needed to control viewpoint amounts to tracking curves in the image, a subset of which corresponds to the occluding contour, and (2) the viewpoint control computations generate a continuous stream of differential motion commands. These commands can be used to move a camera relative to a stationary object, or to rotate the object in front of a stationary camera. Their goal is to achieve or maintain specific geometric constraints between the viewpoint's motion and the object. Drawing from results on the differential geometry of surfaces, these constraints are defined in terms of the occluding contour's shape. As a consequence, differential motion decisions

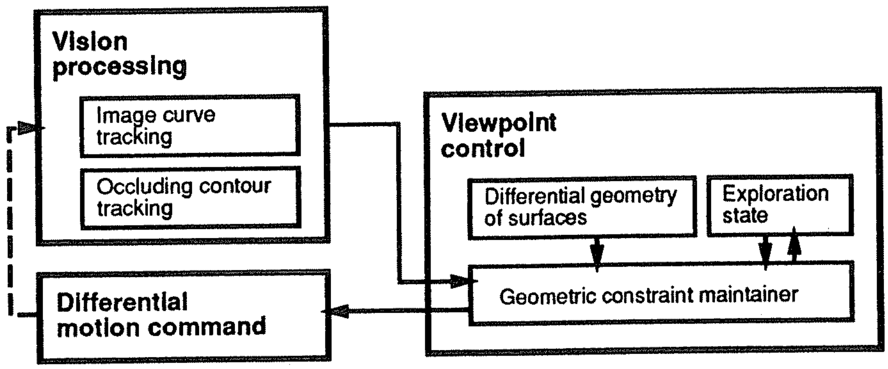


Figure 3: Organization of the continuous, contour-driven viewpoint control framework developed in this thesis. Solid arrows in this and in subsequent figures indicate computational dependencies; dashed arrows indicate indirect dependencies (e.g., motion commands indirectly affect the input of the vision processing module).

are designed to force the contour's shape to "evolve" in a well-defined and constrained way.

We claim that the proposed framework has advantages over existing approaches for viewpoint control when recovering information about the shape of curved objects. These approaches can be broadly categorized as following a *search-based framework*, a *gradient-based framework*, or a *constraint-based framework*. The three frameworks differ in the underlying mechanism that controls viewpoint, as well as the tasks to which they have been applied.

The assumption underlying the search-based framework is that both vision processing and physical camera motions are resource-intensive operations that must be executed as few times as possible. This framework has been applied primarily to three-dimensional model-building [45, 117, 177] and to object recognition [69, 81, 86]. The main idea is to try to extract as much information as possible about the object's shape from the current viewpoint and then use that information to decide where to move next (Figure 4). Viewpoint selection is performed by searching through the space of possible target viewpoints using an appropriate optimization criterion. For example, Connolly [45] constructed a volumetric model of the visible regions on an object from a single viewpoint and subsequently searched for the viewpoint that would maximize the area of the visible and unexplored portions of the object.

In the search-based framework, the single-viewpoint shape recovery computations imply that mechanisms other than viewpoint control are used to recover the shape of an object's visible areas. For

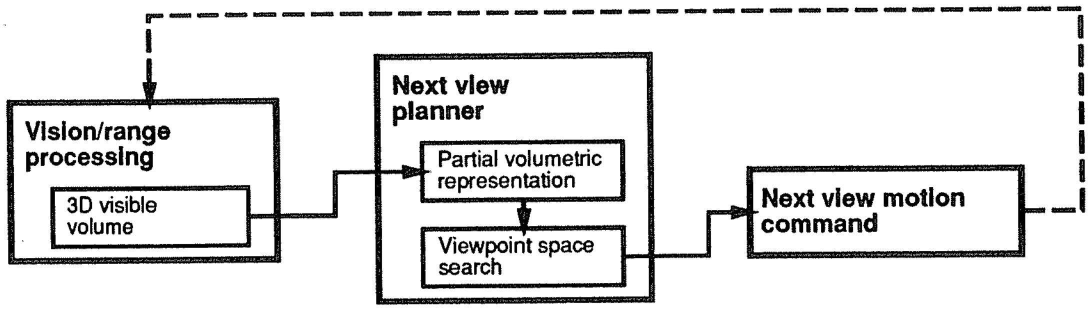


Figure 4: Search-based viewpoint control framework.

example, Connoly [45] used a laser range-finder. Furthermore, in the context of three-dimensional model building, the optimization part of the process can be computationally-intensive. Determination of a viewpoint's utility requires forming hypotheses about the appearance of the environment from the new viewpoint. For distant viewpoints and unknown objects, accurate prediction of an object's appearance is not possible. While heuristic evaluation criteria can be used, guarantees are not provided in existing methods about which parts of a complicated object will or will not be reconstructed during its exploration.

Recent advances in computer hardware design as well as in real-time vision techniques suggest that neither vision processing nor viewpoint control need be time-consuming. The gradient-based framework was designed to exploit this new technology. In the gradient-based framework, viewpoint is controlled by executing a differential motion command that optimizes a gradient-based optimization criterion. The gradient-based framework has been applied primarily to the task of visual servoing, i.e., keeping the image of a moving object stationary by maintaining a fixed geometric relationship with respect to the object [55, 71, 73, 176]. In this task, the camera's position is controlled reactively, in order to match the "external disturbances" causing motion of the object being viewed (Figure 5(a)). The framework has recently been used for obstacle avoidance in environments containing curved objects by first extracting 3D shape information using predefined linear camera movements and a shape-from-motion module [25, 31, 160]. In the context of shape recovery, the gradient-based framework has been restricted to cases where easily identifiable object features such as markings or corners always exist to drive the viewpoint control process by following an appropriate control law [149] (Figure 5(b)).

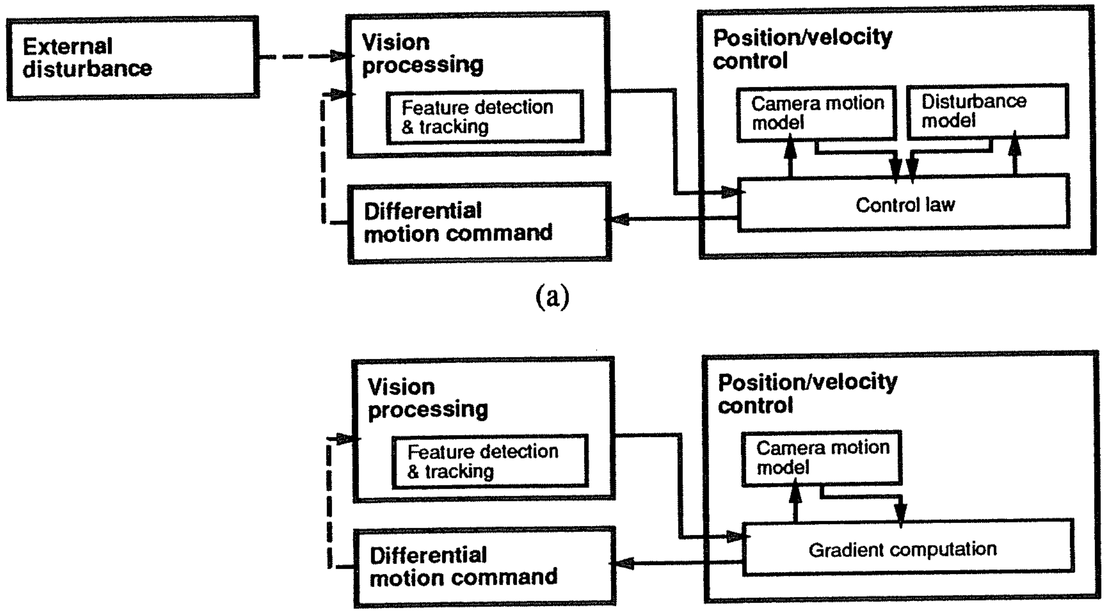


Figure 5: Gradient-based viewpoint control framework. (a) Organization for visual servoing. (b) Organization for recovering shape.

Our framework is similar to the gradient-based framework and was partly inspired by it. Both frameworks consider vision processing and viewpoint control as two continuous processes that are tightly coupled. However, in their current formulation, gradient-based approaches either rely on the existence of markings or discontinuities to drive the viewpoint control process, or use predefined camera movements to recover three-dimensional shape from multiple images. Hence, the question of how image-driven viewpoint control can be used to recover shape information for curved objects with few or no markings and discontinuities has not been investigated. Furthermore, gradient-based approaches are by definition local; it is unclear how they can be used to control viewpoint for performing global recovery tasks such as building a three-dimensional model of an object's entire surface.

At its most general formulation, our continuous, contour-driven viewpoint control framework constrains the viewpoint's motion to impose a specific structure on the evolution of an object's projection. This general principle was recently used to control viewpoint in the context of object recognition

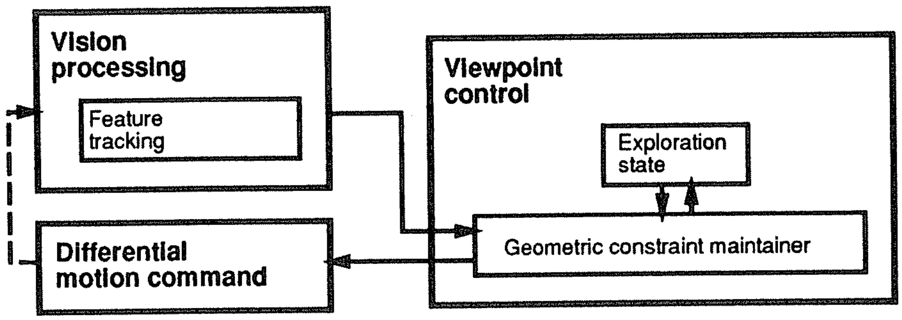


Figure 6: Constraint-based viewpoint control framework.

[182–184], and shape recovery [112]. Together with these recent studies, our work can be classified as following a more general *constraint-based framework* in which image-driven viewpoint control is used constrain the evolution of an object's projection. Unlike gradient-based approaches where the viewpoint's motion is completely unconstrained and is controlled locally by computing a gradient direction, constraint-based approaches force the viewpoint to move on well-defined constraint surfaces [80] (e.g., specific planes [183]), that impose a specific evolution on the object's projection. This simplifies the analysis of the viewpoint's motion by effectively reducing the degrees of freedom of this motion. Strategies based on the constraint-based framework have only been developed for controlling viewpoint around polyhedral objects. For example, Madsen and Christensen [112] used a image-driven viewpoint control strategy to measure the angle between two edges on a polyhedron's face by moving to viewpoint at which the image plane was parallel to that face.

Our continuous, contour-driven viewpoint control framework applies the more general projective-structuring framework to the exploration of curved objects, providing answers to the questions of what features to extract from an image, how to relate them to surface shape, and how to use their dynamic evolution to control viewpoint. From an analytical point of view, the occluding contour's shape does not change arbitrarily under differential viewpoint changes. These changes can be characterized in terms of the differential geometry and topology of curved surfaces. This makes tractable the problem of predicting how the contour's shape will change by constrained viewpoint changes, allows shape computations to be simplified, and leads to provably-correct exploration strategies for curved objects of complex geometry. From a computational point of view, continuous motion leads to effective filtering of the image stream. Since differential viewpoint changes produce small changes in the image, processing can focus on those areas where the object's projection does change, i.e., around the occluding contour.

No single sensor is ideal for exploring objects in every environment. Other sensors such as touch and laser range-finders, if available, can play an important complementary role in this process, and other types of visual information, such as markings, texture, specularities, and shading are likely to be very useful in specific contexts, and when more information about the environment's objects is available. Our goal was to choose an instance of the general object exploration problem, and develop a coherent framework that can be used to study issues related to the cooperation and coordination of sensing and action in the context of visually exploring curved objects, while imposing minimal restrictions on the objects that can be explored.

1.3 Major Contributions

The main contributions of this thesis can be summarized as follows:

- Introduction of a new framework for combining vision and action in the exploration of curved objects. In particular, this framework exploits the use of continuous, contour-driven viewpoint control both to enhance the shape recovery capabilities of an active vision system and to simplify its computations.

- Formulation of exploration as a process of constraining the deformation of the occluding contour of curved objects. It is shown that this formulation allows the use of results from the local and global differential geometry of surfaces to formalize the connection between vision and action in the exploration process.

- Demonstration that continuous, contour-driven viewpoint control leads to local geometrical analysis, local processing in the image, and simple, locally-controlled motions even when global shape recovery tasks are performed. In particular, it is argued that (1) analysis of global shape recovery tasks becomes mathematically tractable, and (2) *tangential viewpoint control* becomes a key elementary motion in the exploration of curved objects. This motion controls viewpoint on the tangent plane of an automatically selected surface point, requires only tangent computations in the neighborhood of the point's projection, and is the cornerstone of all exploration strategies developed in the thesis.

- A theoretical investigation of how viewpoint control and vision processing can be combined to develop strategies with provable properties for exploring curved objects with complex surface geometry.

- Development of an active vision strategy for detecting an object's occluding contour. It is shown that viewpoint control enables detection without extrinsic calibration of the camera and without *a priori* identification of surface markings.

- Development of an active vision strategy for recovering surface curvature from the occluding contour. It is shown that viewpoint control enables local shape recovery without measurement of camera velocities or accelerations, and without *a priori* identification of surface markings.

- Development of an active vision strategy that controls viewpoint to globally reconstruct an object's surface from the occluding contour. No existing approaches guarantee the outcome of this global exploration process for curved objects of complex surface geometry (e.g., non-convex, self-occluding objects), using either cameras or laser range-finders.

1.4 Thesis Outline

The remainder of this thesis is organized conceptually into two parts. The first part, consisting of Chapters 2 and 3, further develops our general approach to visual object exploration and provides necessary background. The approach is then applied in Chapters 4, 5, and 6 to three shape recovery tasks requiring the extraction of progressively richer information about the shape of an unknown object. In each of these three chapters, a distinct viewpoint control strategy is developed through a theoretical investigation of geometric relationships between continuous viewpoint control, the geometry of curved surfaces, and their constantly evolving projection. Initial experiments with simulated and real objects serve as "proofs of principle" and practical illustrations of our theoretical arguments.

Chapter 2 places the work in this thesis within the broad context of shape recovery, presents more details of the approach, and discusses its relation to previous approaches in computer vision that employ image-driven viewpoint control. The key sections of this chapter are Section 2.2, which states the main design criteria of the exploration strategies developed in subsequent chapters, Section 2.3 which motivates the use of the occluding contour for shape recovery, and Section 2.4, which introduces two geometric notions, *special viewpoints* and the *exploration frontier*, which form the basis of all exploration strategies in subsequent chapters.

Chapter 3 introduces *tangential viewpoint control*, which is the basic mechanism used in this thesis to control viewpoint. It involves (1) moving on the tangent plane of a selected point projecting to the occluding contour, and (2) tracking the point's projection as viewpoint changes on that plane. The chapter introduces necessary geometrical background, describes how this motion is used in the thesis to constrain the contour's deformation, and presents an approach for implementing the viewpoint's planar motion and the point tracking operation.

Chapter 4 studies the *occluding contour detection task.* The task involves distinguishing the image curves corresponding to the occluding contour from those that are projections of markings and discontinuities on an object's surface. Detection is based on the *non-stationarity* property of the curves projecting to the occluding contour: When viewpoint is changed, these curves "slide," rigidly or non-rigidly, over the surface. The chapter shows that by moving to a special viewpoint defined by the object's shape and the initial viewpoint, these curves can be distinguished from *stationary* surface curves, i.e., markings and discontinuities. The viewpoint is reached by tangential viewpoint control. Initial experimental results with simulated and real data are presented to demonstrate the effectiveness of the approach.

Chapter 5 studies the *local curvature estimation task.* The task involves computing the curvature of the surface at a selected point projecting to the occluding contour. The exploration strategy developed for this purpose is based on a relation between the geometries of an object's surface and of its occluding contour: If the object is viewed from a special viewpoint, corresponding to the principal direction of the surface at a point projecting to the occluding contour, surface curvature at the point becomes trivial to compute from the contour's shape. The chapter shows that such a viewpoint can be reached by tangential viewpoint control. Initial experimental results illustrating the performance of the method with simulated and real data are presented.

Chapter 6 studies the *global surface reconstruction task*. The task involves building a three-dimensional model of an object's entire surface, or as much of the surface as possible, from images of the occluding contour. The chapter develops an exploration strategy for continuously controlling viewpoint that allows this task to be provably achieved for generic objects of arbitrary shape. The strategy is formulated as a process of constraining the evolution of the exploration frontier (defined in Chapter 2), and involves a repeated application of tangential viewpoint control steps. Simulation results are presented to illustrate the developed strategy and the chapter's theoretical arguments.

Chapter 7 summarizes the main contributions of the thesis and its main limitations, and discusses directions for future work.

Chapter 2

A Framework for Visual Exploration of Surface Geometry

Our goal is to investigate the effectiveness and generality of the continuous, contour-driven viewpoint control framework for recovering shape properties of an unknown curved object. We do this by (1) classifying shape recovery tasks into four broad categories, (2) defining a set of design criteria that we must take into consideration when exploring an unknown object to recover its shape properties, and (3) developing a framework that is applied to the exploration of curved objects, uses the occluding contour, controls viewpoint in a task-driven and image-driven manner, and can be used to perform exploration tasks in each of the four categories of shape recovery tasks while conforming to the stated design criteria.

This chapter discusses the above three elements of the approach. Section 2.1 presents a classification of shape recovery tasks and Section 2.2 presents the design criteria we use. These two sections are not meant to be all-encompassing of the work in computer vision; rather, their purpose is to set the context within which we view our work and our contributions. Section 2.3 defines the occluding contour and motivates its use for exploration. Section 2.4 introduces the notions of a *special viewpoint* and of the *exploration frontier*, the two geometric notions underlying all strategies we develop in subsequent chapters. Sections 2.5 and 2.6 briefly review existing work that exploits viewpoint control, relating it to the discussion in Sections 2.1 and 2.4.

2.1 A Classification of 3D Shape Recovery Tasks

In this thesis we argue that visual object exploration is not an end in itself but must be studied in the context of the requirements of particular types of tasks [6]. The reason is purely computational; given that any vision system has limited resources (e.g., CPU cycles, means for interacting with the environment), these resources have to be applied in the most effective and efficient manner to solve the task at hand [14, 15]. For example, building a three-dimensional model of an object in order to grasp it, which in many cases can be performed simply by analyzing the object's silhouette [26], may not only be unnecessary, but can introduce numerical instabilities caused by inverting the projection process [6]. We study exploration by considering four classes of shape recovery tasks that depend on the acquisition of different types of information about the geometry of an object. Geometric information can be classified with respect to two broad categories: *local vs. global*, and *quantitative vs. qualitative*.

Local shape properties characterize an object's surface in the neighborhood of a point (e.g., convexity/concavity, curvature, etc.)[24, 38]. Global shape properties characterize an object's entire surface (e.g., the configuration of the parabolic curves on an object [136], the configuration of surface orientation discontinuities [79], or a description of an object's entire shape [33, 135]). Visual information is spatially local; the need to interact with the environment in a well-defined manner is therefore particularly critical for performing tasks that require recovery of an object's global shape properties.

Qualitative shape properties depend on affine or projective coordinate descriptions of an object's shape [118], or are defined in terms of bounds or constraints [16, 42, 185] (e.g., surface convexity/concavity, depth ordering of points). Quantitative shape properties depend on Euclidean coordinate descriptions of an object's shape (e.g., surface curvature at a point, distance from the camera, equation of a parabolic curve). Quantitative shape recovery is much more demanding since coordinate information is lost during the projection process.

These two categories lead to the following classification of shape recovery tasks:

- **Local qualitative shape recovery tasks**

  These tasks involve determining surface convexity/concavity at one or more points on the object, identifying the boundaries of visible object regions, computing the affine or projecting coordinates of a collection of points, or identifying the visible surface orientation discontinuities. Example applications include grasp planning [26] and recognition [97].

- **Local quantitative shape recovery tasks**

  These tasks involve extracting a parameterization for local surface patches describing neighborhoods of points on the object's surface, computing the fundamental forms which completely determine the curvature of the surface in such patches [38], or computing the Euclidean coordinates of a set of surface points. Example applications include surface inspection [122], solid modeling and reverse engineering [150], as well as grasp planning [104].

- **Global quantitative shape recovery tasks**

  These tasks involve extracting surface descriptions that describe quantitatively either the entire surface, or regions that are completely determined by global surface properties (e.g., all regions that can be made visible and all non-concave regions). Applications are solid modeling, reverse engineering, and model-building for inspection, grasping, motion planning, and recognition.

- **Global qualitative shape recovery tasks**

  These tasks involve the extraction of affine, projective, or coordinate-free representations for (1) describing regions that are completely determined by global surface properties (e.g., the nesting configuration of convex, concave, and hyperbolic regions on an object's surface [39, 90, 120]), or (2) performing visual search [187] over a globally-defined surface region (e.g., over all potentially-visible regions). Example tasks include qualitative shape modeling, searching for markings on an object's surface (e.g., manufacturer's identification, or bar code), searching for graspable object regions, and object recognition.

There has been a considerable amount of work on controlling viewpoint in environments that have been completely modeled in advance [48, 49, 53, 68, 69, 81, 86, 107, 113, 143, 158, 159, 164, 188]. However, as discussed in Sections 2.5 and 2.6, work on controlling viewpoint to perform shape recovery tasks in the above categories has been limited. We study visual exploration by studying specific instances of the above shape recovery tasks; the tasks of occluding contour detection, surface curvature estimation, and global surface reconstruction studied in this thesis are representative instances of the first three categories of exploration tasks, respectively.

2.2 Key issues

The classification in the previous section suggests that depending on the context in which the exploration of an unknown object takes place, different requirements and expectations will be placed upon the vision system's resources. In particular, three key issues are:

1. **What parameters of the vision system need to be measured?**
2. **How efficient is the exploration process?**
3. **Under what conditions about the environment's objects are the exploration strategies successful?**

The projected geometry of an unknown object is highly dependent on the parameters defining the state of the vision system (e.g., 3D position and orientation, which constitute the camera's extrinsic parameters, intrinsic camera parameters such as focal length and pixel aspect ratio, and motion parameters such as velocity and acceleration). Clearly, the more state parameters known, the easier the task of "inverting" this projection process, and the richer the geometrical information about an unknown object we can visually extract. On the other hand, assumed knowledge of state parameters such as camera position and velocity affects the practical utility of an exploration strategy in three ways. First, such information may be unavailable or hard to obtain (e.g., determining the 3D position requires extrinsic camera calibration [163]). Second, the vision system's parameters may change (e.g., during viewpoint control) making repeated calibrations necessary. Third, errors in state parameter measurements can introduce errors in the extracted shape information and, consequently, negatively affect the outcome of the exploration process. It therefore becomes important to try and keep state measurements to a minimum, requiring only those that can be measured as easily and as robustly as possible. In this thesis we show that image-driven viewpoint control can have an impact on the required measurements, allowing us to "trade off" measurement requirements with simple controlled motions.

The use of viewpoint control also raises an efficiency issue unique to active vision systems. Unlike traditional, passive vision techniques where the parameters of the vision system are not under the direct control of the vision system, image-driven and task-driven viewpoint control *requires* the interaction of the vision system with the environment. This is manifested in two ways. The process of controlling viewpoint in a task- and image-dependent manner can impose a computational burden on the overall exploration process; to avoid this burden, which is not present in passive approaches, requires keeping the viewpoint control computations to a minimum. Furthermore, viewpoint control is a limited resource of an active vision system [72, 142]; this makes it necessary to know how much motion is performed during the exploration process. An important contribution of this thesis is to show that the use of simple and efficient computations for exploring curved objects is not only possible, but also enables us to guarantee the outcome of the exploration process (e.g., completeness of a constructed object model).

A question posed in almost every approach in computer vision concerns assumptions about the geometry of the objects in the environment. We focus on the exploration of curved objects and, primarily, objects with few or no discontinuities. While a clearly limited class, curved objects are being used in manufacturing [128] and can also describe the surface of many natural objects (e.g., rocks, flowers, fruit, etc.). While we impose no restrictions on the class of curved objects for two of the three exploration strategies we develop, the global surface reconstruction task is considered in the context of exploring generic objects. Generic objects are bounded by a smooth surface and can have an almost arbitrary shape: It only takes an infinitesimal deformation to make any smooth object generic. The two key restrictions imposed by the generic object assumption is that generic objects cannot contain flat regions or discontinuities. While these restrictions are clearly important, generic objects do allow us to study the visual exploration problem for objects of nearly unrestricted shape; the problems of exploring an object's non-convex regions and of dealing with self-occlusions must be directly addressed to provably explore such objects.

2.3 The Occluding Contour

The absence of identifiable features that persist across viewpoints (e.g., markings, discontinuities) renders traditional stereo [115] and shape-from-motion [17, 109] techniques inadequate for recovering the shape of curved objects with few or no markings on them. Two general approaches to this problem have been suggested in the literature: (1) Inferring shape directly from image intensities and their variation, and (2) using the occluding contour. Unfortunately, intensity-based approaches such as shape-from-shading [78, 126], shape-from-specularities [27, 93, 171], and shape-from-texture [4, 8],

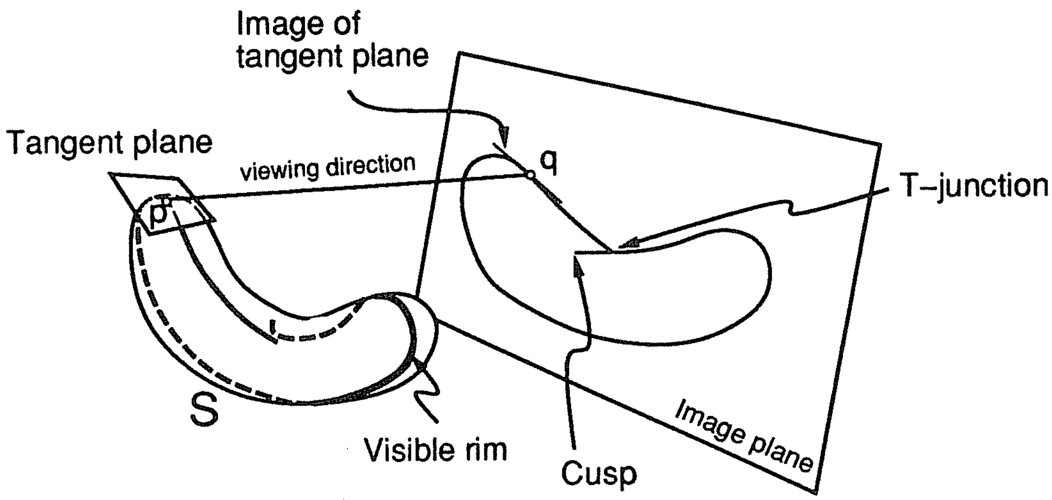


Figure 7: The visible rim and the occluding contour of a bean-shaped surface (adapted from [36]). The occluding contour consists of a single curve whose endpoints are a T-junction and a cusp.

while potentially very useful, rely on assumptions about the reflectance properties of the viewed surface or about its texture distribution, both of which may not be known.

We use the occluding contour to guide the visual exploration process and to extract the shape of the object being explored. The *occluding contour*[1] is the projection of the one-dimensional set of visible points at which the surface "turns away" from the viewer. Formally, the occluding contour is defined as the projection of the *visible rim* [90],[2] the set of visible surface points whose tangent plane contains the viewpoint (Figure 7). For almost every curved surface and almost every viewpoint (in a measure-theoretic sense), the occluding contour is a collection of open and closed curves whose endpoints terminate at *cusps* and *T-junctions* [91].

The main reason for using the occluding contour for the exploration of curved objects is that there is a strong connection between the shape of a surface, the shapes of the visible rim and the occluding contour, and the way that their shapes change when viewpoint changes. Furthermore, the occluding contour is a source of both qualitative and quantitative information about the local and global shape of curved objects, and, along with shadows [147], discontinuities [18, 114] and markings [43, 153], is

[1] Also known as the *contour* [36], the *apparent contour* [42, 44], the *profile* [62], and the *extremal boundary* [170].
[2] Also known as the *contour generator* [42], the *critical set* of the projection mapping [36], and the *fold* [91, 181].

the only shape information that we can hope to extract from an image when physical surface properties are unknown. From a computational point of view, the occluding contour is just a collection of edges, which can be efficiently tracked across frames [28].

The connection between the shape of an object and its occluding contour makes it applicable to qualitative, quantitative, local, as well as global shape recovery tasks:

- **Local, qualitative shape recovery**

  Several studies have shown that the geometry of the occluding contour severely constrains the underlying surface geometry even from a single viewpoint [18, 108, 114, 121, 136]. For example, the sign of the Gaussian curvature at the visible rim (i.e., whether the surface at the rim is convex or hyperbolic) is completely determined by the occluding contour [88]. However, even though a single image of the occluding contour constrains surface geometry, the problem of distinguishing the occluding contour itself from markings and discontinuities cannot be resolved from a single viewpoint [170, 193]. In general, unless viewpoint can be controlled in an object-dependent manner, this problem becomes equivalent to recovering surface curvature at the visible rim, a quantitative task requiring camera calibration and viewpoint motion information [44].

- **Local, quantitative shape recovery**

  A slight change in viewpoint will affect the geometry (i.e., curvature) and possibly the topology of the visible rim, and hence the occluding contour. Moreover, the set of rim points changes and therefore new constraining information about the surface shape becomes available. This observation has been used to recover a complete description of local surface shape by forcing viewpoint to undergo a predefined motion [44, 62]. Unfortunately, when no surface markings can be identified *a priori*, predefined camera motion leads to approaches that require measurement of velocities and accelerations.

- **Global, quantitative shape recovery**

  In global quantitative shape recovery tasks, there is a fundamental need for viewpoint control when the occluding contour is used: When the only information available is the occluding contour at a single viewpoint or at a spatially localized set of viewpoints, we can only hope to extract local information about an object's geometry, i.e., for points in the neighborhood of the visible rim. Even though approaches have recently appeared for piecing-together local surface descriptions derived from the occluding contour [82, 146, 192], the problem of controlling viewpoint in an object-dependent manner to recover an object's global shape has not been considered. In this thesis we show how, by appropriately controlling viewpoint, global shape recovery tasks can also be solved.

- **Global, qualitative shape recovery**

  Object-dependent viewpoint control is necessary for performing global qualitative shape recovery tasks since not all points on an object's surface can be seen from a single viewpoint [45]. The occluding contour is an important piece of information in such tasks because it is the projection of the boundary of an object's visible regions, and it can be used to induce the visibility of occluded regions [117].

Despite the above advantages of the occluding contour, any shape recovery approach based on the occluding contour has to deal with a number of complications:

- **Occluding contour detection:** In real-world situations the surface of an object may contain surface markings, shadows, discontinuities, as well as specularities. An efficient and reliable method for distinguishing the occluding contour from these other surface curves, if they exist, is therefore necessary (Chapter 4).

- **Occluding contour tracking:** Unlike surface markings and discontinuities, the visible rim is a collection of curves that deform and change connectivity as viewpoint changes. In order to track the occluding contour, these changes must be taken into account. Current curve tracking algorithms cannot handle such general image curve deformations and connectivity changes. A promising direction for future work on this problem is briefly discussed in Section 4.8.

- **Effects of global surface geometry:** Changes in the topology and the geometry of the visible rim and the occluding contour are important when dealing with global shape recovery tasks because such "events" determine which parts of the surface are explored. These changes depend on the global shape of the object's surface, and provably-correct global shape recovery cannot be achieved without taking them into account (Chapter 6).

- **Recovering the shape of concavities and flat regions:** A fundamental limitation of the occluding contour is that points in concave surface regions can never project to the occluding contour, and that the occluding contour becomes degenerate when flat areas are viewed "edge-on". Consequently, the shape of an object's concavities and the extent of flat regions cannot be recovered quantitatively, although their shapes can be constrained [192]. When cameras are used for quantitative shape recovery the alternative is to use information such as shading, shadows, discontinuities and markings to recover information about such regions.

2.4 Continuous, Contour-Driven Viewpoint Control

Chapter 1 motivated a framework in which exploration of an unknown object is formulated as a process of constraining the deformation of the object's occluding contour or, equivalently, the motion of the visible rim over the surface. Clearly, object exploration is task-dependent, and not every exploration process enables such a formulation. However, this thesis shows that we can solve a number of shape recovery tasks by exploiting two general manifestations of this dynamic structuring process:

1. **Moving to a special viewpoint**
2. **Controlling the motion of the exploration frontier**

A key result in this thesis is that even though the shape recovery tasks we consider have different objectives, being both local and global, and even though they rely on different principles for formulating exploration as a dynamic structuring process, the viewpoint control mechanisms in all of these tasks always control viewpoint in one of two simple ways: Either by moving on the tangent plane of the surface at a visible rim point automatically selected during the exploration process, or by moving on a normal plane at such a point. As a consequence, viewpoint is always *locally-controlled* even though the task itself may require *global* shape recovery. Thus viewpoint is always controlled with respect to a single point on the object's surface, and depends on image-computable quantities that are localized in the image.

Moving to a special viewpoint. Special viewpoints are viewpoints that have a well-defined geometric relationship with the local or global shape of a curved object (e.g., a viewpoint along the principal direction at a surface point, or a viewpoint on an object's support plane). Very little attention has been paid in the past to the existence of such viewpoints (Section 2.5); special viewpoints have been predominantly considered as unwanted degeneracies (e.g., a viewpoint corresponding to a side view of a cube) that occur only by "accident" and complicate visual processes such as object recognition [110, 182]. However, because viewpoint is highly constrained, these viewpoints provide qualitative and quantitative geometric information about the surface that cannot be easily obtained from arbitrary viewpoints.

Two of the shape recovery tasks considered in this thesis, namely occluding contour detection (Chapter 4) and local curvature estimation (Chapter 5), are solved by using special viewpoints. To reach these viewpoints we ask how we can generate a continuous path that leads us to the desired special viewpoint. Because these special viewpoints can be defined in terms of the shape of the occluding contour itself, reaching them involves dynamically changing viewpoint until the contour's shape "evolves" into one that satisfies the conditions characterizing a special viewpoint.

Controlling the motion of the exploration frontier. In order to perform tasks involving exploration of an object's global shape such as global surface reconstruction a vision system must be able to (1) represent the set of points on an object's surface that have already been explored, and (2) characterize precisely how a specific viewpoint change will affect this set. The *exploration frontier* is the boundary of this set. In general, the structure of the exploration frontier changes unpredictably if we move to a distant viewpoint. However, we can make such tasks tractable by formulating exploration as a process of controlling viewpoint in a continuous fashion in order to "slide" the exploration frontier over the entire surface, or over as much of the surface as possible (Figure 8).

In this thesis we control the evolution of the exploration frontier for the purpose of global surface reconstruction (Chapter 6). We ask how to generate a continuous path so that the visible rim slides over the entire surface, even though the the connectivity of the visible rim may change, and even though its motion depends on the object's local and global shape. A basic result is that because we control viewpoint in a continuous and highly-constrained way, we can perform a local analysis of this task

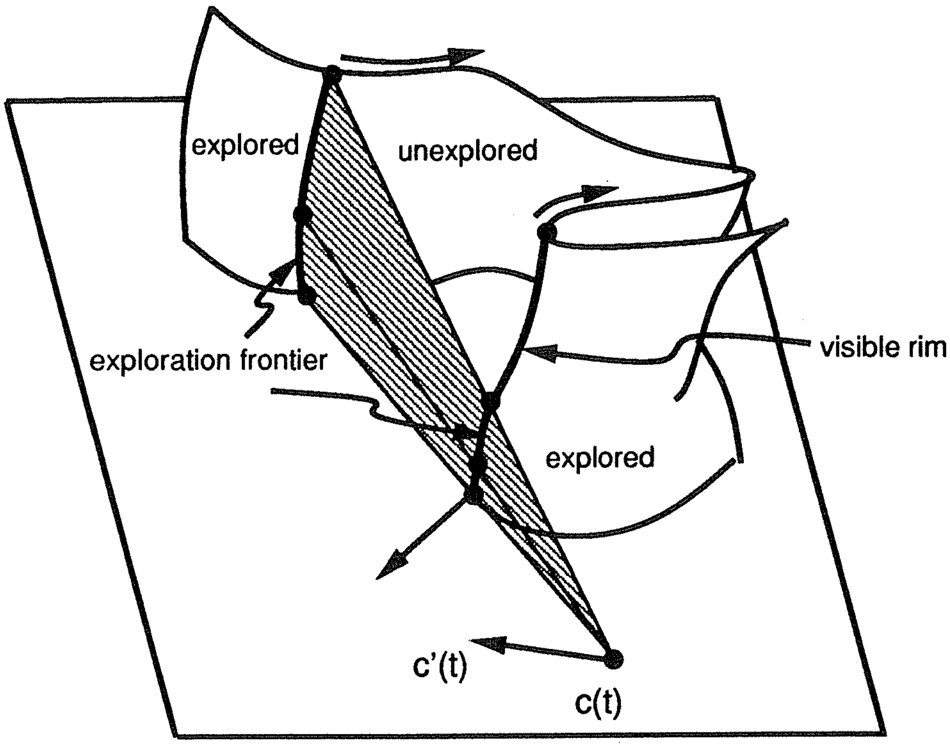


Figure 8: Controlling the motion of the exploration frontier to make visible all points on the surface; the explored points are the points that have already been made visible. In this example, parts of the exploration frontier coincide with the visible rim. Under an instantaneous viewpoint change, both the visible rim and the exploration frontier will slide over the surface as shown in the figure.

while at the same time guaranteeing global reconstruction of generic objects of arbitrary shape. In effect, continuous, contour-driven viewpoint control gives us the ability to provide global guarantees by reducing the global reconstruction process to a local one at those discrete positions along the viewpoint's path where the rim's topology changes.

2.5 Relation to Previous Approaches Using Image-Driven Viewpoint Control

Very little attention has been given in computer vision to the use of image-driven viewpoint control for either exploring an object or for recognizing it. Furthermore, with the exception of the work by Blake and his group, the strategies developed either assume polyhedral objects, or assume the existence of markings and discontinuities.

The existence of special viewpoints has been one of the main motivations for exploiting viewpoint control. Following the projection-structuring framework, Wilkes and Tsotsos [183, 184] suggested a strategy for reaching viewpoints where the projected lengths of two fully-visible creases on an Origami object are simultaneously maximized; these viewpoints were used to simplify recognition of such objects and are reached through a sequence of differential motions. The strategy was later generalized by Madsen and Christensen [112]. In an analogous approach, Yoshimi and Allen [189] used a strategy for aligning the optical axis with a hole in an object (assumed to be a point in the image) in order to simplify peg-in-hole insertion. Grosso and Ballard [70], motivated by general observations about the re-orientation behaviors of biological organisms (e.g., forcing the optical axis to be perpendicular to a book before reading it), proposed a control scheme for changing the orientation of the optical axis about arbitrary vectors in space.

Image-driven viewpoint control has also been used for increasing robustness in local quantitative exploration tasks. Smith and Papanikolopoulos [149] incorporated viewpoint control into a gradient-based, visual servoing control scheme in order to obtain a robust estimate for the depth of points on a textured object. Hervé and Aloimonos [75] proposed the use of small, controlled movements to increase the robustness of shape-from-shading and shape-from-texture computations. However, both strategies relied on the assumption that either markings can be identified on the object or that surface reflectance and motion information is available.

Image-driven viewpoint control in environments containing curved objects was studied by Blake *et al.* in the context of grasping curved objects or moving around them [25, 29, 31, 51, 160]. They formulated viewpoint control as a local, gradient-based search for the optimal differential movement. A predefined camera motion before each optimal movement was used to recover the shape of the surface in the neighborhood of the visible rim, thus enabling a local curvature-based optimization process. The developed strategies were local; no guarantees were given about their success or failure for general non-convex objects.

Image-driven viewpoint control for exploring curved objects has thus been very limited. This thesis is an attempt to study viewpoint control as a continuous, image-driven, appearance-structuring process, and to use it for simplifying the tasks of both local and global, qualitative and quantitative shape recovery for curved objects.

2.6 Relation to Previous Approaches Using Range-Driven Viewpoint Control

The use of visual information is only one way to explore unknown objects. The development of tactile sensors [2, 3, 20, 151] and sensors for extracting physical properties of objects is still a topic of active research [37, 59, 87]. Laser range-finders [23] are the most commonly used alternatives to cameras for exploring unknown objects.

Laser range-finders are particularly appropriate for extracting local, quantitative information about an object's shape since they estimate depth, a dimension lost during the projection process. Such sensors are usually composed of a laser beam emitter device and a device for receiving the beams reflected from an object's surface. State-of-the-art range finders can provide accuracies for three-dimensional measurements that far surpass those obtained from images [22, 74]. However, these sensors have a number of limitations that still make cameras appealing for exploring unknown objects. Range-finding systems have a fixed field of view whose extent is dependent on the desired accuracy. On the other hand, the accuracy of a range-finder depends on the optical characteristics of the sensed object's surface (e.g., reflectance, color, etc.), as well as on environmental conditions such as ambient light and temperature; even though some effects such as sensitivity to light absorption characteristics can be reduced using high-powered lasers, such lasers can lead to eye safety hazards.

Since range-finders simplify the problem of extracting local, three-dimensional coordinate information, a considerable amount of effort has been spent on building three-dimensional models of unknown objects, although work on grasping has been reported as well [104]. One of the key problems is occlusion [104, 117]; since coordinate information can only be extracted for points that are visible both to the beam emitter and to the receiver, viewpoint control is needed to get the shape of occluded regions. Initial studies of this problem appeared almost a decade ago [45]. Since then, viewpoint has been controlled in a predefined manner [1], or controlled using the search-based viewpoint control framework.

The optimality criteria guiding the search process either minimize the uncertainty of the current noisy measurements [72, 177–180], or maximize estimates of the area reconstructible at the new viewpoint [117].

Unfortunately, the problem of using range-finders for fast and reliable global reconstruction of curved objects arbitrary shape is still unsolved. The question of how global surface geometry (e.g., self-occlusions) affects the correctness of existing exploration strategies has not been considered. Currently, no system exists which can provide guarantees about which parts of a complicated object (e.g., with holes, dents, and protrusions) will be reconstructed.

The usefulness of trying to optimize the number of viewpoint changes is also not clear. Although the developed optimization criteria attempt to minimize this number, they do not necessarily minimize the length of the path traced by going to each of those viewpoints. Wixson [186], after studying the viewpoint selection problem in simulated two-dimensional polygonal environments, has recently argued that the strategy of moving to a random viewpoint has similar performance characteristics to strategies that attempt to "intelligently" decide on moving to a distant viewpoint.

The results in this thesis are useful in the context of range-based object exploration in two ways. First, our study of the global reconstruction task is directly relevant to the study of global reconstruction using range-finders. The initially-occluded regions on an object can only be made visible by forcing the exploration frontier to slide over them; the strategies in this thesis are designed precisely for this purpose. Second, the availability of a range-finder should not lead to a computational burden and to viewpoint control strategies that lack correctness guarantees; we believe that both cameras and range-finders have their place in the exploration of an unknown object, and can complement each other's limitations.

The key question in combining vision and range sensing for exploration is how to use a range-finder in the most effective manner; one way to achieve this is to consider range-finders as tools for (1) refining shape information that has been extracted with simpler and more efficient means, such as cameras, and (2) overcoming the limitations of simpler sensors, e.g., the inability to explore concavities and flat areas on an object's surface using only occluding contour information obtained by a camera.

2.7 Summary

This chapter laid the conceptual foundations of our work, motivating our study of the general problem of visual object exploration, the main issues addressed in our viewpoint control strategies, and the use of the occluding contour for exploration.

Two key notions were introduced: The existence of *special viewpoints*, and the motion of the *exploration frontier*. We argue that these notions can be extremely valuable in the exploration of unknown objects, can be exploited by controlling viewpoint in a continuous, contour-driven fashion, and can lead to simple, locally-controlled motions even when performing global shape recovery tasks. *Tangential viewpoint control*, which is the main locally-controlled motion we use to reach special viewpoints and to structure the motion of the exploration frontier, is described in the next chapter.

Chapter 3

Tangential Viewpoint Control

The continuous, contour-driven viewpoint control framework requires addressing two competing practical issues: On one hand, it requires that per-frame computations during the viewpoint's motion be efficient. On the other, it requires that a continuous stream of geometrical information is available to guide viewpoint which, in principle, can be computationally-expensive to extract. Viewpoint control mechanisms that balance these two issues are therefore desirable.

We claim that *tangential viewpoint control* is such a mechanism. In particular, we claim that tangential viewpoint control requires simple and local computations in the image while at the same time, leads to provably-correct exploration strategies for both local and global shape recovery tasks. Tangential viewpoint control can be defined as follows: "Given a visible rim point p projecting to the occluding contour, circumnavigate p by moving on its tangent plane until a specified condition is satisfied." Among the conditions used in this thesis are maximization of the occluding contour's curvature at the projection of the point p (Chapter 5), and occlusion of p by points closer to the camera (Chapter 6). Tangential viewpoint control involves computing p's tangent plane or its intersection in the image, changing viewpoint in a constrained way (i.e., on a single plane), tracking p's projection across frames, and detecting its occlusion. All viewpoint control strategies developed in this thesis can be described as repeated selections of a point p and applications of this highly-constrained motion.

This chapter studies tangential viewpoint control and its geometry. Sections 3.1 and 3.2 provide necessary geometrical background. Section 3.3 presents the key ideas of this chapter; it shows that tangential viewpoint control can be used as an "appearance structuring tool" to manipulate the relationships between the shape of the surface, the shape of the occluding contour, and the way in which this shape changes under continuous viewpoint control. Sections 3.4 and 3.5 then consider the practical issues of restricting the viewpoint's motion to a plane, and tracking the projection of the surface point defining the tangential motion. These two sections are not required for the material in subsequent chapters. Section 3.6 summarizes the chapter.

3.1 Projection Model

Projection models relate the shape of objects to the shape of their image. Several projection models have been used in the computer vision literature, the most notable among these being the *perspective* and *orthographic* projection models [5]. These models differ in two fundamental ways. First, the notions of a "viewpoint" and of a "viewpoint change" are defined differently. Second, the projection of the same physical point on an object's surface is different for each of them. The first difference implies that the effects of a tangential motion depend on the projection model. The second difference impacts the exploration process computationally because different methods are needed to invert the projection process. Both differences must be taken into account in the design of viewpoint control strategies for exploration.

The model most accurately describing the projection process in CCD cameras is the perspective projection model. Given the camera's pose (position and orientation) and its focal length, this model defines the projection of a 3D point p to be the point of intersection of the image plane with the line through p and the camera's focal point (Figure 9(a)).

Of the three strategies developed in this thesis, two are applicable to both projection models and one of them relies on the orthographic projection model (Chapter 4). In this model the points on an object's surface are projected along parallel rays that are perpendicular to the image plane (Figure 9(b)). These rays define the camera's *viewing direction.* For the sake of uniformity our results are all presented using the orthographic projection model.

It is clear from the geometry of the two projection models that the projection of a given 3D point depends both on the position of the camera and on its orientation: In the case of perspective projection any change in the camera's pose along any of the three spatial and three orientation dimensions will

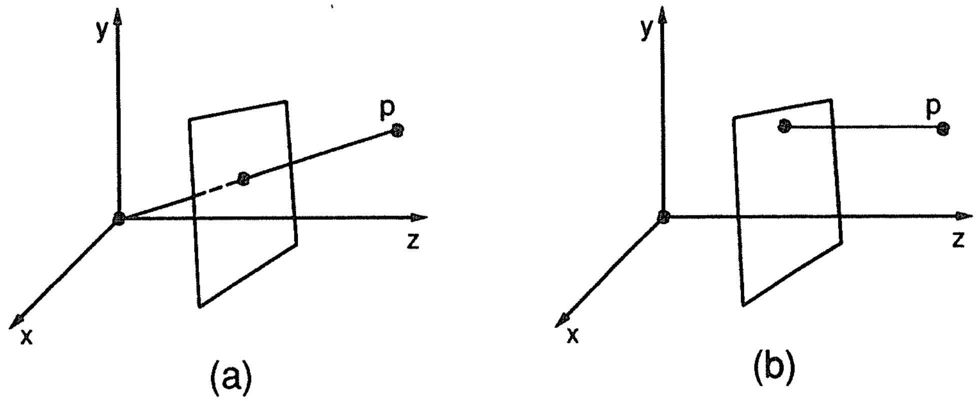


Figure 9: Projection models. (a) Perspective projection. (b) Orthographic projection.

change the point's projection; in orthographic projection, the parallelism of the projection rays implies that motions perpendicular to the image plane do not affect the projection process.

While in general, camera motion under orthographic projection has five degrees of freedom, the position and orientation degrees of freedom are coupled in all the strategies we describe. In particular, (1) camera motions are always performed with respect to a reference point on the object being explored [13, 70], (2) the camera's orientation is always such that the reference point's projection is at the center of the image, and (3) no rotations about the viewing direction are performed. Any positional movement of the camera is therefore combined with a *fixation*, i.e., a concurrent change in viewing direction that centers the reference point in the image. The reference point is always the point defining a tangential viewpoint control motion.

A consequence of the orthographic projection model and of the tight coupling of the camera's positional and orientational degrees of freedom is that for a given reference point, there is a 1-1 correspondence between camera positions and viewing directions. We use the term *viewpoint* to refer to both. For a given reference point, viewpoints can be thought of as points on a *viewing sphere* of infinite radius surrounding the object [36] (Figure 10).

Unfortunately, orthographic projection cannot be used in all circumstances as an accurate model of the projection process. First, it cannot be used in exploration problems that require motion with three positional degrees of freedom (e.g., getting closer to an object in order to see through an opening) since

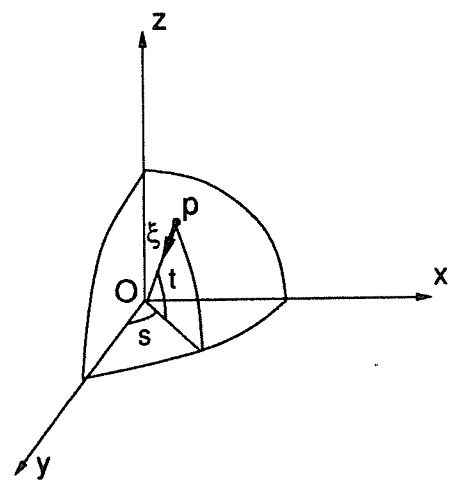


Figure 10: Viewing direction ξ is represented as the point $(\cos t \cos s, \cos t \sin s, \sin t)$ with $s \in [0, 2\pi), t \in [0, \pi)$.

the camera has only two positional degrees of freedom under orthography.

Second, and most important, orthographic projection is valid only when the distance of the camera to the viewed object remains large compared to the camera's focal length [161]. This assumption can be quite unreasonable when the viewing parameters are not known or are not controllable. In an active context, however, this assumption is not as restrictive because the camera-to-object distance does not have to be arbitrary; this distance can be increased, and the validity of the ray parallelism property can be actively verified, to ensure the accuracy of the orthographic projection model.

3.2 Local Surface Geometry

The effects on an object's projection of moving on the tangent plane of a visible rim point p are completely determined by the local and global surface geometry. Below we introduce some basic notions from the local geometry of surfaces that are required for this study.

Suppose S is a smooth, oriented surface in $\Re^3$, viewed under orthographic projection along a viewing direction ξ. Let x be a parameterization of S and $p = \mathrm{x}(u, v)$ be a point on S. The partial derivatives $\mathrm{x}_u(p), \mathrm{x}_v(p)$ of x with respect to u and v define $T_p(S)$, the plane tangent to S at p. The visible rim of S

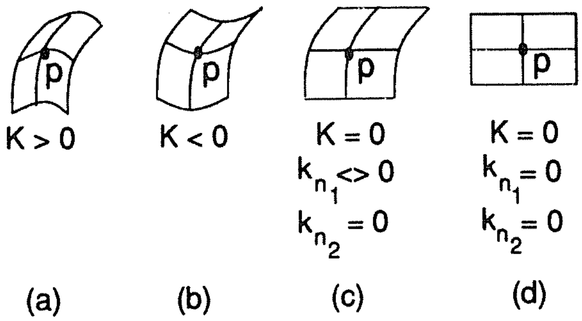


Figure 11: Classification of the surface point p based on K: (a) p is elliptic, (b) p is hyperbolic, (c) p is parabolic, and (d) p is planar.

is the set of those visible points p for which $T_p(S)$ contains a line parallel to ξ.

In order to move on $T_p(S)$ we need to be able to compute it, or at least know its intersection with the image. Barrow and Tannenbaum [18] showed that $T_p(S)$ can be determined using a very simple relationship that holds between the tangent plane at p and the shape of the occluding contour at p's projection: The tangent plane at p is the plane defined by the viewing direction ξ and the tangent to the occluding contour at p (Figure 7). Therefore, its intersection with the image is along the tangent to the occluding contour at p's projection.

Motion on the tangent plane of a visible rim point p produces qualitatively different effects depending on the way in which this tangent plane intersects the surface in the vicinity of p. Specifically, let $N(p) : S \to \mathbf{S}^2$ be the Gauss map of S, assigning a unit normal vector $N(p)$ in the direction of the vector product $\mathbf{x}_u \wedge \mathbf{x}_v$ at every point $p \in S$. The normal section of S along a direction ξ in $T_p(S)$ is the plane curve produced by intersecting S with the plane of ξ and $N(p)$. The second fundamental form, $\mathbf{II}_p(\xi)$, gives an expression for the curvature of this curve at p [38, 90]. $\mathbf{II}_p$ has a single maximum and minimum, k_{n_1} and k_{n_2}, along two orthogonal directions, e_1 and e_2, respectively. These directions are called the *principal directions* at p. The surface in the neighborhood of p can then be described qualitatively by looking at the sign of their product $K = k_{n_1}k_{n_2}$, the Gaussian curvature of S at p (Figure 11).

This qualitative surface classification determines the local structure of the intersection of the surface with p's tangent plane [38, 89]. The strategies we develop in this thesis control viewpoint only on the

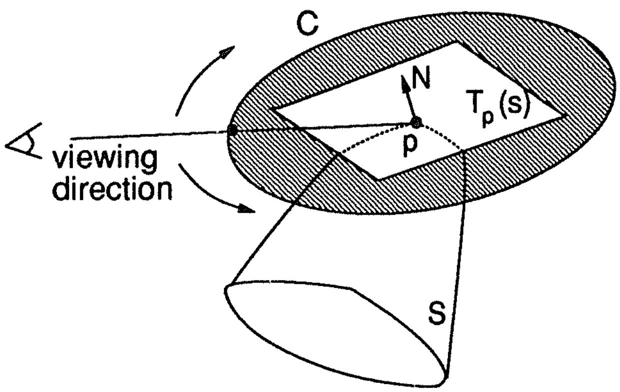


Figure 12: Moving on the tangent plane of an elliptic point p. Viewing directions on $T_p(S)$ correspond to points on a circle C of infinite radius lying on $T_p(S)$ and centered at p.

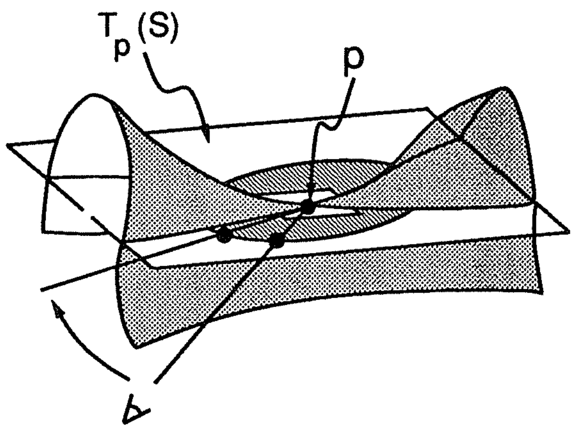


Figure 13: Moving on the tangent plane of a hyperbolic point p.

tangent plane of elliptic and hyperbolic points. We therefore restrict our analysis to these two cases (Figures 12 and 13). For elliptic and hyperbolic points, the local structure of $S \cap T_p(S)$ is shown in Figure 14.

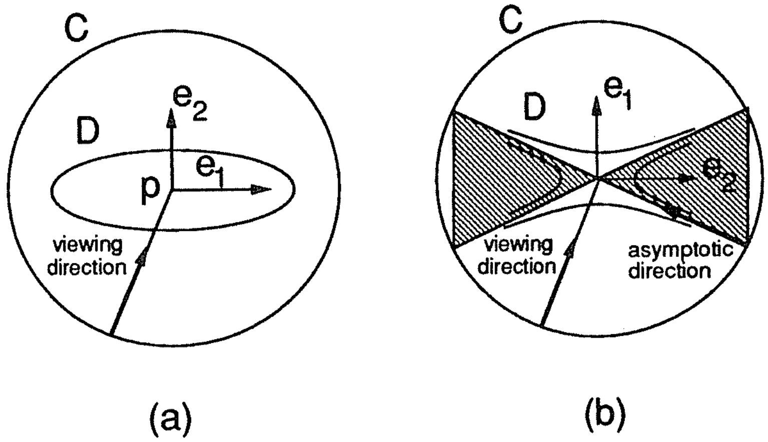


Figure 14: Structure of $T_p(S)$ in the neighborhood of p. Top views of the tangent plane are shown for the cases where p is (a) elliptic, and (b) hyperbolic. The axes represent the principal directions, and the origin corresponds to the point of contact, p, with the surface. C represents the circle of viewing directions on the tangent plane. (a) As viewpoint changes on $T_p(S)$, p will not be occluded by points in its neighborhood. The intersection of the surface with planes close to p and parallel to $T_p(S)$ is described by Dupin's indicatrix [38], which is a first-order approximation for the curve of intersection, and is an ellipse in this case. (b) When p is hyperbolic, p can become occluded by points in its neighborhood (shaded areas). The visibility transition will occur when the viewing direction is along an *asymptotic direction*, along which the surface is locally flat. Dupin's indicatrix is a hyperbola, with asymptotes coincident to p's asymptotic directions.

3.3 Tangential Motion as an Appearance Structuring Tool

One of the basic characteristics of the occluding contour is that it depends both on viewpoint and on the shape of the viewed object. Motion on the tangent plane of a visible rim point becomes a key motion in the exploration of an unknown object because it can be used to dynamically structure the shape of the occluding contour and the motion of the visible rim in a way that factors out some of their viewpoint-dependent properties; what remains is *viewpoint-independent* information about the local and global shape of the object being explored. More specifically, tangential motion allows us to compensate for three viewpoint-dependent properties of the occluding contour and visible rim:

1. The one-dimensional set of points comprising the visible rim will change under any infinitesimal change in viewpoint.

2. The shape of the occluding contour (i.e., tangents and curvatures) may change under an infinitesimal change in viewpoint.

3. Under an infinitesimal viewpoint change, the motion of the visible rim over the surface and its connectivity depend on (1) the local and global shape of the surface, (2) the initial viewpoint, and (3) the direction of the infinitesimal motion.

Below we present the main ideas behind how the dependency on viewpoint of each one of these properties can be factored out through tangential viewpoint control. These ideas are expanded upon in Chapters 4-6, which put the ideas into the context of specific exploration tasks.

Forcing visible rim point stationarity. Suppose p is a point on the visible rim at a given viewpoint. p will not remain on the visible rim if that viewpoint is perturbed in an arbitrary fashion. However, if we start changing viewpoint on p's tangent plane, p will remain on the visible rim for as long as it does not become occluded by points that are closer to the camera [82, 134, 138] (Figures 12,13). This follows directly from the definition of the visible rim.

Consequently, even though in general the visible rim can be thought of as sliding over the surface when viewpoint changes continuously, we can force the visible rim to remain "stationary" at specific points by controlling viewpoint on their tangent plane. We show in Chapter 4 that by appropriately choosing these points we can detect the occluding contour without the need for extrinsic camera calibration or viewpoint motion measurements.

Factoring out the dependence of the occluding contour's shape on viewpoint. In general, when a surface point p is viewed from different viewpoints on its tangent plane, the local shape of the occluding contour (i.e., its curvature at p's projection) will be different. However, the manner in which the occluding contour's shape changes around p's projection as viewpoint changes on p's tangent plane is completely determined by the local shape of the surface at p. This allows us to reach special viewpoints on p's tangent plane at which the contour's shape depends only of the local shape of the surface. This is the basis of our shape recovery approach in Chapter 5.

Factoring out dependence of the visible rim's motion on viewpoint We show in Chapter 6 that under small viewpoint adjustments, the changes in the connectivity of the visible rim near a point p and the visible rim's motion near that point are completely determined by the relationship between the initial viewpoint and the endpoints of the arcs of viewpoints from which p is visible from its tangent plane (Figure 15). Furthermore, there exists a direct relationship between the global shape of a surface and the distance between the endpoints of these arcs. We show that we can force the visible rim's motion to depend only on the global shape of the surface by keeping the viewpoint's relationship to the endpoints of these arcs fixed (e.g., always at the middle of the arcs) before forcing the visible rim to slide over points p on the exploration frontier. Our main result in Chapter 6 is that this is sufficient to structure the visible rim's motion in a way that guarantees global reconstruction of arbitrarily-shaped generic objects.

3.4 Enforcing Planar Viewpoint Control

Tangential viewpoint control requires viewpoint to be controlled in a very constrained way, i.e., by moving on a plane corresponding to the tangent plane at a selected visible rim point and on a collection of normal planes. Given that robotic arms and pan-tilt units have constraints of their own on how their end-effectors can be moved, the issue of how these "hard" constraints on the viewpoint's motion can be performed with reasonable accuracy and speed becomes important. We briefly describe below one approach for implementing such constrained motions that (1) exploits the natural constraints imposed on the motion of orienting devices such as turntables and pan-tilt units, and (2) changes viewpoint by controlling the orientation of the object.

Suppose that we have placed an object on a horizontal turntable and that both the viewing direction and the image rows are horizontal. Furthermore, suppose that the image's center row is tangent to the object's occluding contour at the projection of a point p (Figure 16(a)). Then, because the viewing direction and the image's center row define a horizontal plane, any rotation of the turntable will force the viewing direction to move strictly parallel to p's tangent plane. In order to enforce planar viewpoint control on the tangent plane of an arbitrary visible rim point we develop a strategy that enforces this special geometry between the object, the turntable, and the camera (Figure 16). Ideally, this can be achieved by finding a transformation that makes the point's tangent plane horizontal, and applying that

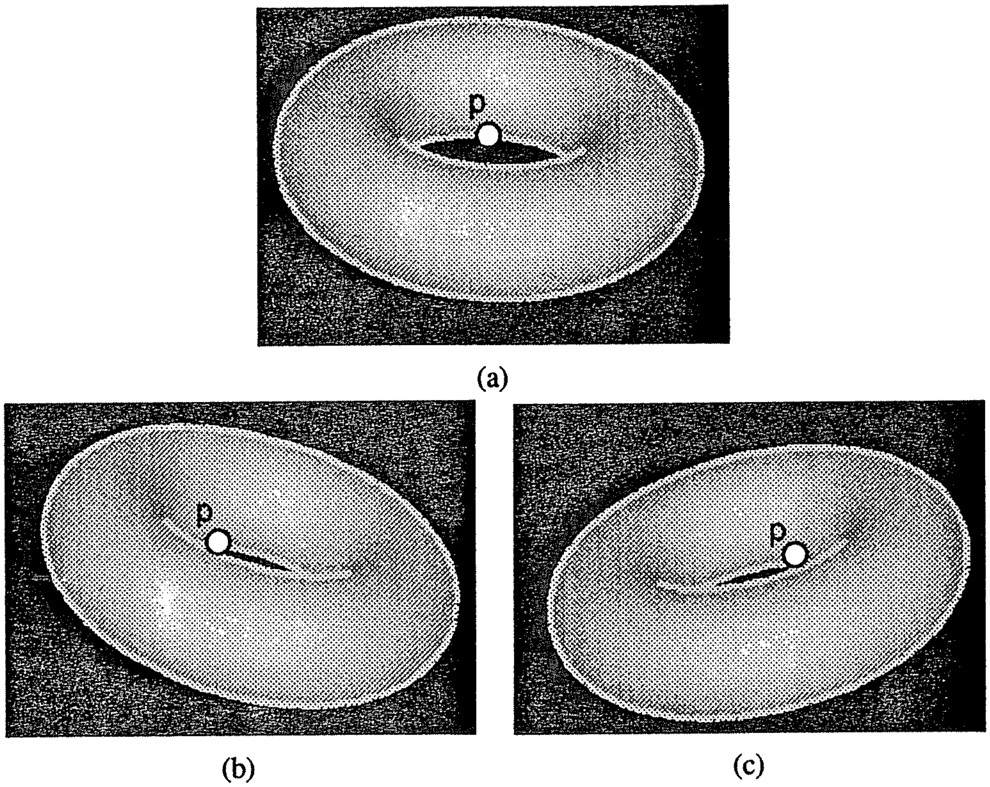


Figure 15: Views of a torus from $T_p(S)$, which is horizontal and perpendicular to the plane of the page. (a) View from an arbitrary position on $T_p(S)$; (b) view from the position where p first becomes occluded while moving to the left on $T_p(S)$; (c) view from the position where p first becomes occluded while moving to the right. The distance between the viewpoints corresponding to (b) and (c) and their distance from the viewpoint corresponding to (a) determine how quickly the visible rim curve containing p will disappear under a vertical downward viewpoint adjustment starting from (a)

transformation to the object so that the original viewpoint (i.e., the position of the camera relative to the object) remains unchanged by this transformation.

In theory, object re-orientation amounts to a rotation about an axis parallel to the viewing direction. In practice, object re-orientation is limited by the configuration and the degrees of freedom of the devices used to re-orient the object [103]. The questions we answer are: Given $T_p(S)$, (1) how can we re-orient the object so that $T_p(S)$ becomes as close to horizontal as possible,[1] and (2) how can we

[1]The ability to make $T_p(S)$ horizontal will depend on the joint limits, accuracy, and resolution of the device used to re-orient the object. We do not treat the problem of joint limits here. On the other hand, the accuracy and resolution of

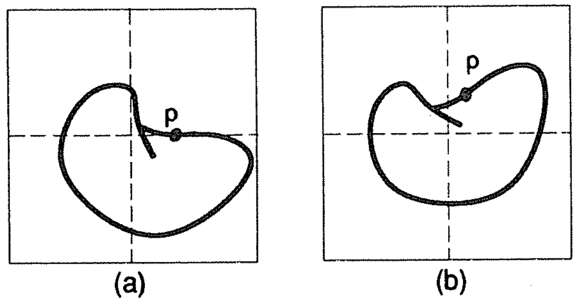


Figure 16: (a) Geometry of viewpoint control within $T_p(S)$. The object orientation and camera position are such that $T_p(S)$ is horizontal. (b) To allow viewpoint control on $T_p(S)$ for an arbitrary visible rim point p, we have to re-orient the object to obtain a view identical to the one in (a) (up to a translation in the image plane).

ensure that the viewpoint remains unaffected by this process? Object re-orientation can be achieved with a rotational stage mounted on pan-tilt unit (Figure 17). Fortunately, by adding a simple step to the object re-orientation process we can also guarantee that the initial viewpoint is preserved up to a change in distance from p (which does not affect the ability to move on $T_p(S)$).

More specifically, we base our strategy for object re-orientation on the relationship between the surface normal at the selected point and the three axes of rotation of the RRR unit shown in Figure 17. The strategy, outlined below, can be used to implement viewpoint control on any plane tangent to a visible rim point p. It requires calibration of the hand-eye system, i.e., knowledge of the angles between the three rotational axes of the RRR unit and any plane defined by the viewing direction and a line in the image. Figure 18 shows the geometry of the object re-orientation process and Figure 19 shows views, which were obtained by manually controlling the orientation of a RRR unit and the camera's position, of an object undergoing this re-orientation process.

Object Re-Orientation Strategy

the orienting device determines the distance between the selected point p and the closest point, p', on the object whose tangent plane is horizontal when the object re-orientation process is completed. As long as p's tangent plane is close to being horizontal after this process, the distance between p and p' will be small. This distance is determined up to first order by the principal curvatures of the surface at p.

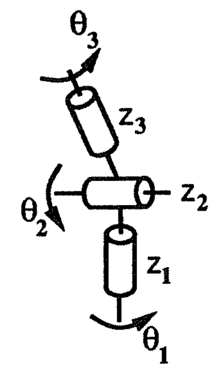


Figure 17: Object re-orientation can be achieved using an orienting device with three consecutive rotational joints (RRR unit). The first joint axis $\mathbf{z}_1$ corresponds to a "pan" axis, and joint axis $\mathbf{z}_2$ corresponds to a "tilt" axis. $\mathbf{z}_1$ and $\mathbf{z}_2$ are perpendicular, as are $\mathbf{z}_2$ and $\mathbf{z}_3$.

Step 1: Compute the rotation θ_3 about $\mathbf{z}_3$ that forces the surface normal, $N(p)$, to become perpendicular to $\mathbf{z}_2$.

Step 2: Compute the rotation θ_2 about $\mathbf{z}_2$ that forces $N(p)$ to become collinear with $\mathbf{z}_1$.

Step 3: Rotate the object first by θ_3 about $\mathbf{z}_3$ and then by θ_2 about $\mathbf{z}_2$ (Figure 19(c),(d)).

Step 4: Let θ_1 be the rotation about $\mathbf{z}_1$ induced by the rotations about $\mathbf{z}_2$ and $\mathbf{z}_3$. Rotate the object by $-\theta_1$ about $\mathbf{z}_1$ (Figure 19(e)).

The first step provides us with all the information we need to compute the two rotations making $T_p(S)$ horizontal and can be performed using a camera-centered coordinate frame. These rotations are then applied in Step 4. Step 5 ensures that the original relationship between the camera position and point p is preserved up to a change in distance from p: Since rotations about axes $\mathbf{z}_2$ and $\mathbf{z}_3$ in general cause rotations in $T_p(S)$, Step 5 corrects for any such rotations. With this strategy, accurate motion along arbitrary planes can be performed.

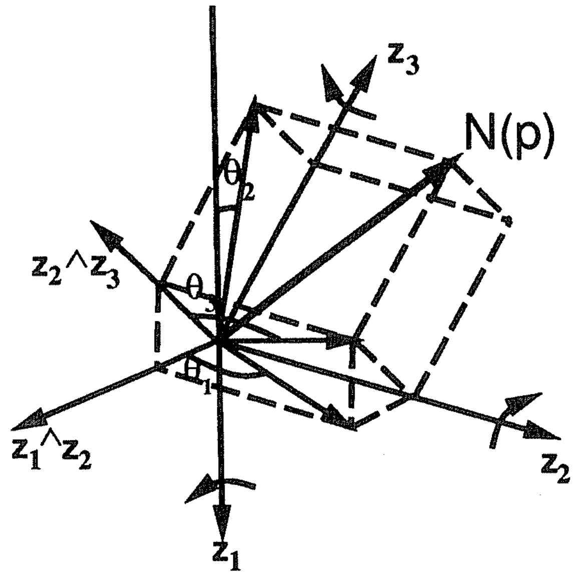


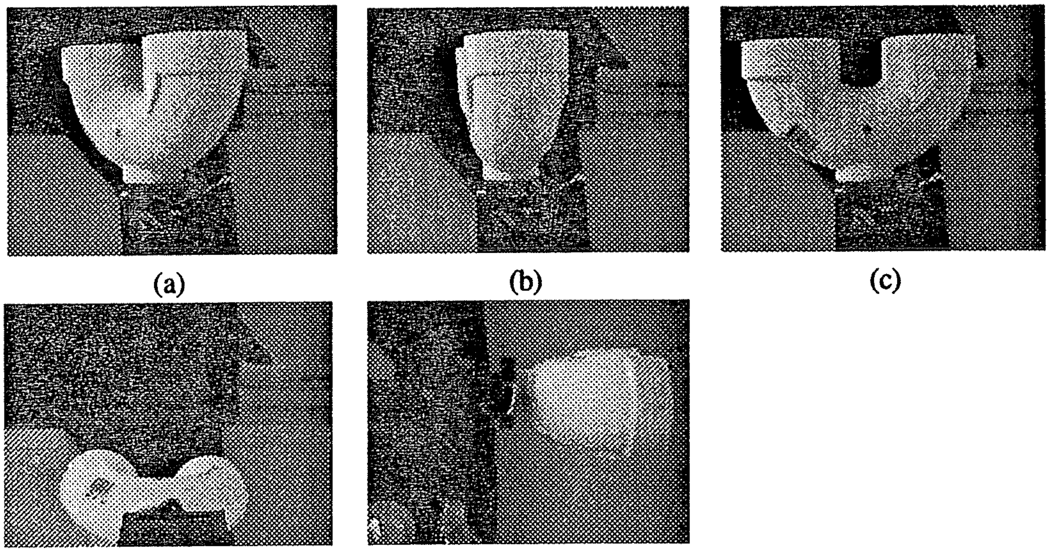
Figure 18: Geometry of object re-orientation. Rotation by θ_3 about $\mathbf{z}_3$ forces $N(p)$ to lie in the plane of $\mathbf{z}_1, \mathbf{z}_3$. Rotation by θ_2 about $\mathbf{z}_2$ forces $N(p)$ to become collinear with $\mathbf{z}_1$. The rotation about $\mathbf{z}_3$ induces a rotation about $\mathbf{z}_1$ which must be corrected in order to ensure that object re-orientation does not affect the direction along which p is viewed on $T_p(S)$.

3.5 Tracking Points During Tangential Viewpoint Control

One of the useful properties of tangential viewpoint control is that this motion can be performed using local and efficient processing in the image. In particular, the exploration strategies we develop in the next chapters require only four computational steps in order to control the viewpoint's motion:

- Computing the occluding contour tangent at the projection of a selected elliptic or hyperbolic visible rim point p
- Tracking p's projection as viewpoint changes on $T_p(S)$
- Detecting when p becomes occluded during motion on $T_p(S)$
- Detecting when a point tangent to $T_p(S)$ becomes visible during motion on $T_p(S)$

Here we discuss a simple implementation of these steps which allowed us to control viewpoint on the tangent plane of selected elliptic or hyperbolic visible rim points.



(a) (b) (c)

(d) (e)

Figure 19: (a) View of a pipe. (b) View of the pipe from a position where the dot on the pipe's surface in (a) lies on the visible rim. (c) Rotating about axis $\mathbf{z}_3$. (d) Rotating about the horizontal axis $\mathbf{z}_2$. The tangent plane at the circular mark is now horizontal. (e) Rotation about the vertical axis $\mathbf{z}_1$ ensures that the point will be viewed along the same viewing direction on its tangent plane.

Clearly, apart from the first step, the remaining steps are impossible to perform on a curved object with no markings or discontinuities when viewpoint is controlled in an arbitrary fashion. However, they become particularly easy to solve in our case precisely because we control viewpoint on the tangent plane of the point being tracked.

First consider the problem of tracking a visible rim point p while changing viewpoint on $T_p(S)$. Suppose for the sake of specificity that $T_p(S)$, the viewing direction, and the image rows are all horizontal. The occluding contour at p will then be tangent to a row in the image (Figure 16(a)). When viewpoint is controlled on $T_p(S)$, p projects to the occluding contour as long as it remains visible. So, to track p we need to know which occluding contour point corresponds to p when the viewpoint is changed. This is given to us by the geometry of Figure 16(a): Since the viewpoint remains on $T_p(S)$, the occluding contour at p's projection must always be tangent to same image row. This leads to the following tracking procedure:

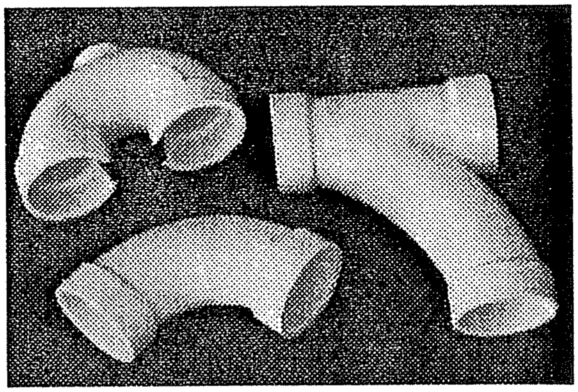


Figure 20: Objects used in the experiments.

Tangent Point Tracking Procedure: Let l be the row tangent to p's projection. To track p, match p's projection in the previous frame to the point in the current frame that is tangent to l and is closest to p's previous projection.

In practice, to perform tracking we use five steps: (1) Center a small window W on row l and on the column of p's previous projection, (2) apply a Canny edge detector to W, (3) discard all edge pixels whose detected orientation differs by more than a threshold from l's orientation, (4) perform hysteresis thresholding on the remaining edge pixels, and (5) track the edge pixel closest to p's previous projection.

Occlusion of the tracked point is detected when tracking fails. To detect when a point tangent to $T_p(S)$ becomes visible after p's occlusion, we process the window W in a manner identical to the way we track p, looking for edge pixels that are close to row l. Since processing is only performed within W, however, we need to know where to position this window along l. One approach would be to estimate where p would project if it were visible. However, to do this we either need to know p's 3D coordinates, or we need to track other rigidly-moving points [95, 118]. Instead, we used a simpler approach: When p becomes occluded we simply shift W along l by the window width at each successive frame. This reduces the effective frame rate for detecting p's reappearance, which we compensate for by reducing the speed by which the viewpoint is moved on $T_p(S)$ during this disocclusion detection process. The benefit is that neither 3D information nor tracking of additional points is needed.

Figure 20 shows the objects we used to test the above point tracking and occlusion/disocclusion detection approach. The point tracking computations allowed us to maintain tracking while rotating these objects with speeds of 40 degrees/second using no dedicated hardware. Disocclusion detection was achieved with rotations up to 10 degrees/second.

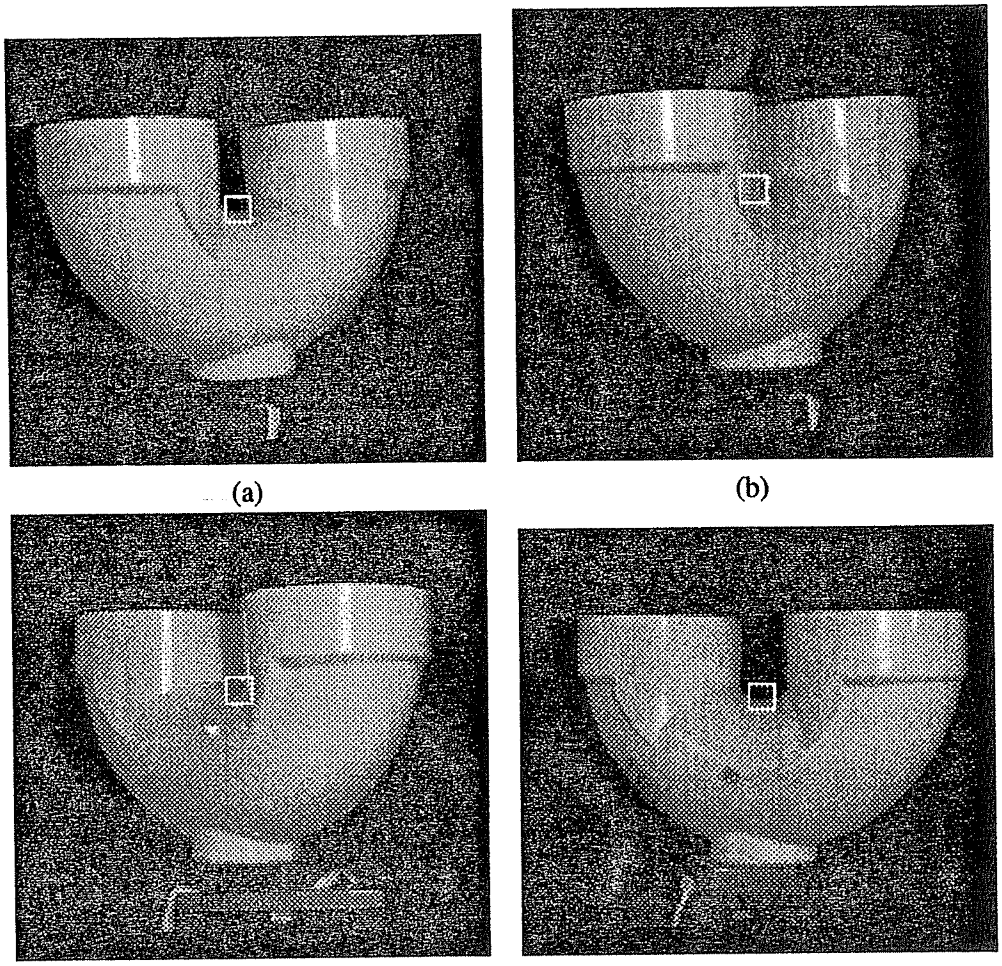
What makes these computations efficient is the size of the window W: In our particular experiments, with an image resolution of 640x512 pixels, the tracked point did not deviate by more than 5-6 pixels from p's initial row, allowing window sizes of less than 20x20 pixels (see Figures 21-24). We also found that the tracking process is relatively insensitive to the choice of thresholds, which were left unchanged in all our experiments. Edge pixels whose orientation differed by more than $\pi/20$ radians from l's orientation were rejected to encourage good localization of tangency points and few false positives. Edge detection was performed with $\sigma = 4$, and hysteresis thresholding used low thresholds to make the tracker resistant to shadows and changes in shading, which are common when dealing with complicated objects.

In our limited experiments, the disocclusion detection test proved very effective in accurately detecting when the tracked point became disoccluded (Figures 21, 24). Furthermore, to improve the robustness to noise in the disocclusion detection computations, given the low thresholds we use for edge detection, we required that tracking be maintained for at least 4-5 frames on edge points in W before point disocclusion was signalled. For the objects we considered, this approach succeeded in detecting point disocclusion without generating false positives.

3.6 Summary

This chapter introduced tangential viewpoint control as a basic tool for structuring the shape of the occluding contour for exploration. The geometry of tangential viewpoint control was presented and practical issues of realizing such controls were discussed. While from a geometrical point of view tangential viewpoint control is simple, it raises two important practical questions that need to be addressed further.

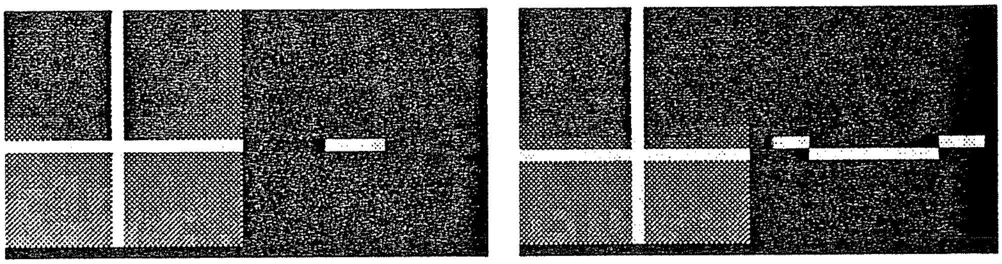
Constrained viewpoint control requires the means to relate projected surface geometry to viewpoint motion. This problem becomes particularly hard for curved objects with few or no markings and



(a) (b) (c) (d)

Figure 21: Point tracking while moving on its tangent plane. (a) Initial view of the object. The window W is centered on the point p to be tracked. $T_p(S)$ is horizontal. (b) View of the object when p's occlusion is detected during counter-clockwise object rotation. (c) View of the object when p's occlusion is detected during clockwise object rotation. (d) View of the object when disocclusion is detected for a point tangent to $T_p(S)$. The object is viewed from a direction nearly opposite the one in (a).

discontinuities since no fixed reference points will generally be found on their surface. The simple approach taken in this thesis is to use hand-eye calibration as a way of relating image tangents to planar motions in space. The problem, however, deserves more attention. In particular, it may be possible to verify the quality of this calibration by checking whether the occluding contour remains tangent to the

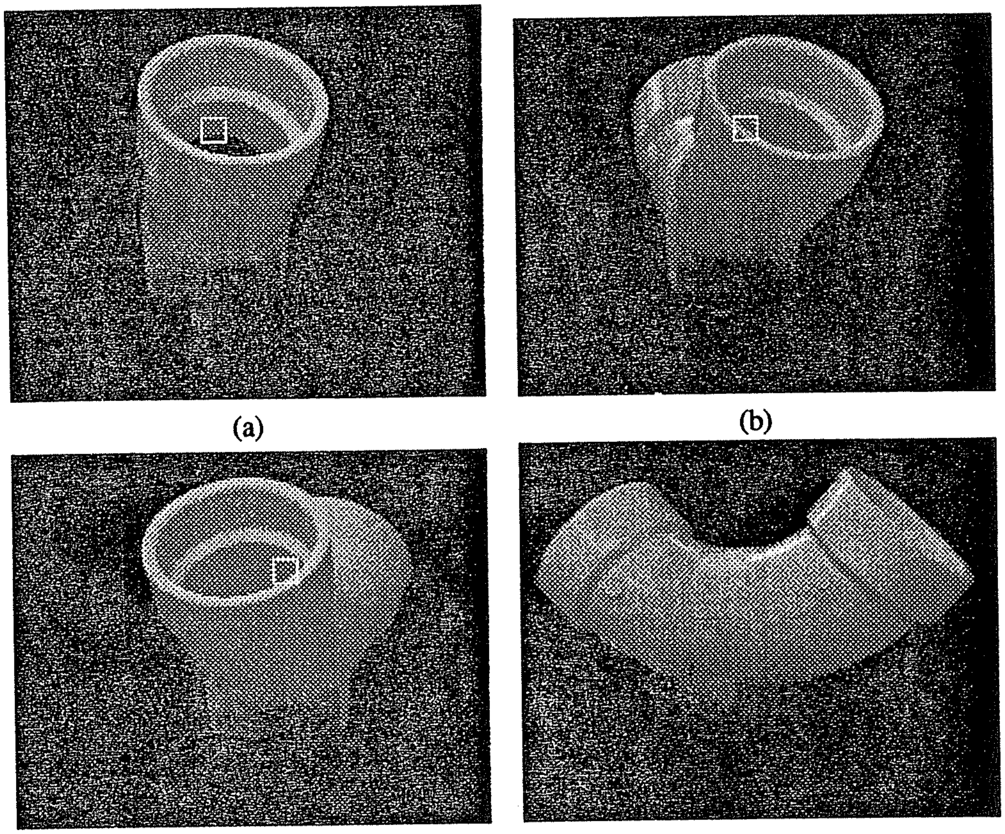


(a) (b)

Figure 22: (a) *Left:* Magnification of the window in Figure 21(a). The cross is centered on the point to be tracked. *Right:* Output of window processing after Canny edge detection and orientation thresholding, and before hysteresis thresholding. (b) Same as in (a) for the window in Figure 21(d).

motion plane at the point being tracked.

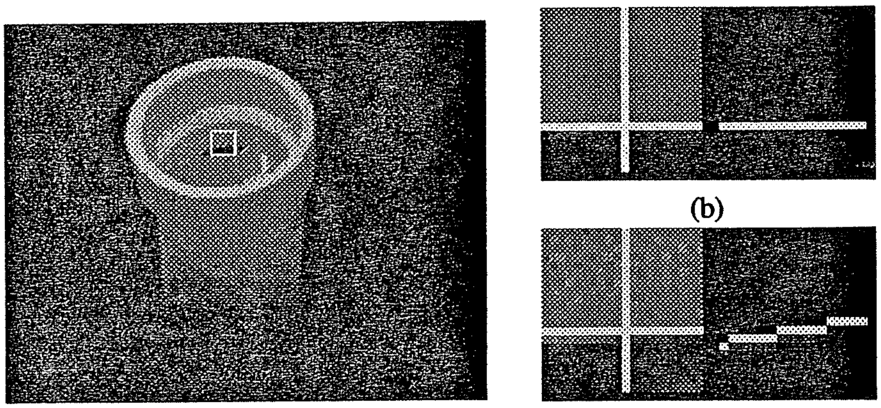
The problem of tracking a surface point while moving on its tangent plane also deserves a more detailed analysis (e.g., along the lines of [52]). Even though our simple tracking approach proved fast and effective for our purposes, additional efficiency gains (as well as the ability to deal with larger image motions) can also be obtained using a Kalman-filter approach to perform edge tracking [28].



(a) (b)

(c) (d)

Figure 23: Application of the system to a different object. (a) Initial view of the object. The window is centered on the point p to be tracked, which lies on the upper interior surface of the pipe. $T_p(S)$ is horizontal. (b) View of the object when p's occlusion is detected during counter-clockwise object rotation. (c) View of the object when p's occlusion is detected during clockwise object rotation. (d) Another view of the object during counter-clockwise object rotation. This view was obtained after p's occlusion was detected in (b) but before any point disocclusions on $T_p(S)$ were detected.



(a)

(b)

(c)

Figure 24: Detecting point disocclusion. (a) View of the object in Figure 23 when disocclusion is detected for a point tangent to the horizontal plane corresponding to $T_p(S)$ in Figure 23. This view corresponds to viewing the pipe's interior from a direction opposite the one in Figure 23(a). (b) *Left:* Magnification of the window in Figure 23(a). The cross is centered on the point to be tracked. *Right:* Output of window processing after Canny edge detection and orientation thresholding, and before magnitude thresholding. (c) Same as in (b) for the window in (a).

Chapter 4

Occluding Contour Detection

The occluding contour plays a key, dual role in our continuous contour-driven viewpoint control framework. Its continuously-evolving shape guides the viewpoint control process during the exploration of an unknown object. Furthermore, it is the source of information we use about the unknown geometry of an environment's objects.

In this chapter we present an approach for detecting the occluding contour curves in an image. This detection process is a necessary initial step in the exploration of curved objects since, in practice, an object's surface may contain markings or discontinuities which also project to curves in the image. Occluding contour detection can be formulated as follows: "Given a collection of image curves, determine which curves belong to the occluding contour and determine their *sidedness*, i.e., on which side of the occluding contour the surface closest to the camera lies." Apart from detecting the contour, this also allows us to get a segmentation of an image in terms of the visible surfaces, to find grasping points [29], and to determine the sign of the Gaussian curvature of the surface at the visible rim [34, 88].

The rest of the chapter is organized as follows. The next section presents the basic ideas of the approach and compares it to previous work on occluding contour detection. Section 4.2 introduces the geometry of the problem. Section 4.3 studies how the projections of the visible rim and of markings and discontinuities deform when viewpoint is changed in a continuous fashion. The section also motivates the use of a prediction-verification scheme for detecting the contour after moving to an appropriate special viewpoint. Section 4.4 then uses this result to develop a viewpoint control strategy for detecting the occluding contour, under the assumption that the image of markings and discontinuities at arbitrary viewpoints can be predicted. Section 4.5 presents the main result of this chapter, which gives us a way to form these predictions through tangential viewpoint control by using elements from the theory of affine-invariant representations. Section 4.6 discusses how we can account for errors in the image data, and Section 4.7 presents results from initial experiments. The chapter concludes with a discussion in Section 4.8.

4.1 Active Occluding Contour Detection

Under continuous viewpoint control the visible rim possesses two properties that distinguish it from markings on an object's surface:

- The curves comprising the visible rim "slide" rigidly or non-rigidly over the surface, possibly changing their connectivity.
- The deformation of the visible rim's projection, the occluding contour, uniquely determines the shape (i.e., curvature) of the parts of the surface over which the visible rim slides [44, 62].

Most previous work on identifying the occluding contour of curved objects focused on the second property: Because occluding contour deformations uniquely determine shape, one can check whether the deformations of an image curve determine the shape of a curved surface. Since the deformation of the projection of a surface marking corresponds to a surface of infinite curvature coinciding with the marking itself [42, 44, 169, 170], this test is sufficient for identifying the occluding contour curves in a sequence of images. Unfortunately, the test involves comparing the speeds and accelerations of the curves in the image sequence to the speed and acceleration of the viewpoint's motion, and, hence, requires accurate measurement of these quantities (or an accurately calibrated stereo system [170]). This makes the occluding contour process sensitive to errors, and is undesirable when the occluding contour is used to extract qualitative information about an object's shape, e.g., for model indexing or for grasping, where accurate measurements of the viewpoint's motion are not necessary. Furthermore, the robustness of the surface reconstruction process itself can be greatly improved by *a priori* distinguishing between the occluding contour and the surface markings [44], and, as will be shown in Chapter 5, specialized viewpoint control strategies can be used to simplify local curvature estimation once the occluding contour has been identified.

This chapter shows that when viewpoint is controllable the occluding contour can be identified *directly*, i.e., without first computing surface shape (distance and curvature). This is achieved by exploiting the first property of the visible rim: Unlike the visible rim curves, surface markings are *stationary*, i.e., their position on the surface is fixed and viewpoint-independent.[1] Instead of attempting to accurately measure the speeds and accelerations of the visible rim curves over the surface, we utilize the stationarity property of surface markings; by determining which curves in the image are projections of non-stationary surface curves, we factor out the need for recovering metric properties about the surface or the viewpoint's motion.

Our approach is closely related to that of Zisserman *et al.* [193] who also suggested a direct approach for detecting the non-stationarity of the occluding contour, although without controlling the viewpoint's motion. Unfortunately, the ability to distinguish the occluding contour from markings is highly dependent on the viewpoint's motion and the discrimination performance can be poor when viewpoint is not under the direct control of the vision system. Furthermore, object-independent motions cannot be used, in general, to detect the sidedness of the contour, and some surface markings need to be identified in the image *a priori* to bootstrap the occluding contour detection process.

Continuous, image-driven viewpoint control can solve these problems. In particular, no markings need to be identified *a priori* and viewpoint can be controlled so that discrimination power depends only on global properties of the objects being viewed. The only requirements of this approach are that (1) a correspondence can be established during the viewpoint control process for the image curves whose identity is sought (i.e., "surface marking" or "occluding contour"), and (2) at least four image curve points with parallel tangents can be identified at the initial viewpoint.

The method is based on one simple observation: Suppose a curve on the visible rim coincides with a surface marking. Because the position of the visible rim curve depends on viewpoint while the position of the surface marking does not, the coincidence relationship between the projections of the two curves will not be preserved when the viewpoint changes. So, if we are able to compute how a visible surface

[1] When the position of the light source and an object is fixed, image curves corresponding to shadows are also stationary. We do not distinguish between such image curves and the projections of surface markings in this chapter.

curve would project at other viewpoints assuming it is stationary, we can simply change viewpoint and compare its projection with the one predicted under the stationarity assumption. When the predicted projection of a curve does not coincide with the actual one, the curve must be on the visible rim.

The crucial issue one must address to exploit this observation is how to predict the projection of a surface curve under the stationarity assumption. Previous work on predicting novel views of a three-dimensional object either assumed the availability of an object model [145], the existence of a collection of "model" views in which the projections of either the surface markings or the visible rim curves have been identified [167], or the existence of a small number of easily-identifiable point features on the surface of an unknown object that could be matched across frames [148]. Here we present a detailed geometrical analysis of this prediction problem that (1) shows how to distinguish stationary from non-stationary surface curves using an active vision system and discusses under what conditions this discrimination can be achieved, (2) shows how this ability can be used to detect the occluding contour as well as the contour's sidedness when no surface markings have been identified *a priori*, and (3) takes into account errors in image measurements.

The basic assumption used in all previous approaches for identifying the occluding contour and for predicting novel views of an object was that viewpoint motion was object-independent. This means that the motion of the viewpoint between any two views of the object is not related in any way to the geometry of the object. This is a reasonable approach, however, only when the vision system cannot control viewpoint. When the viewpoint's motion can be controlled, the choice of viewpoint(s) does not have to be arbitrary.

The significance of our method lies in the use of continuous, image-driven viewpoint control to achieve and maintain specific geometric relationships with the viewed surface in order to distinguish the occluding contour from the projections of stationary surface curves. We show how we can use an active vision system to move to special viewpoints in which the image of a stationary surface curve can be accurately predicted, and then use this predictive power in order to classify the curves in the image either as occluding contour curves, or as projections of stationary surface curves. Throughout this chapter we assume that each image has been processed to extract a collection of curves.

4.2 Viewing Geometry

The shape and position of the visible rim depends on the shape of a surface S and the viewpoint. This is the fundamental difference between visible rim curves and surface markings; the surface position of the curves corresponding to such markings is independent of the viewpoint. We qualitatively characterize the difference between these two types of curves with the notion of *non-stationarity*:

Definition 4.1 *A surface curve is* stationary *if its position on the surface does not change when viewpoint changes. It is* non-stationary *if its position is viewpoint-dependent.*

Our goal is to exploit the non-stationarity of the visible rim to identify the occluding contour. This operation requires a correspondence between points on the surface at different viewpoints, and between points in images. We use the epipolar plane correspondences [44, 62, 170] for this purpose.

In particular, suppose the viewing direction changes according to $\xi(t)$ on a motion plane with normal N_E. This motion defines a family of planes parallel to the motion plane called the *epipolar planes*. If $B(t_0)$ an image curve at time t_0, the epipolar plane correspondence matches a point $q \in B(t_0)$ to the intersection of $B(t_0 + \delta t)$ with the epipolar plane through q. This correspondence induces a correspondence between points belonging to the surface curve $\beta(t)$ projecting to $B(t)$.

Figures 25(a,b) make explicit the non-stationarity property characterizing the visible rim curves: When the tangent of an occluding contour curve at a point q does not belong to an epipolar plane, we can think of the point $q(t)$ corresponding to q at time t as being the projection of a point $p(t)$ that "moves" on the intersection of the surface with the epipolar plane through q. We call such points $p(t)$ *non-stationary*.

The key property of the epipolar correspondences used in our approach is derived from the geometry of Figure 25(c): Even though, in general, we cannot determine which points in the image are projections of stationary points, as discussed in Chapter 3 we can *force* the stationarity of specific points by appropriately controlling viewpoint; viewpoint simply needs to move in a plane parallel to the tangent plane at those points.

Tangential Motion Property: If a point q on an image curve has its tangent parallel to the motion plane, and epipolar plane correspondences have been established for q across

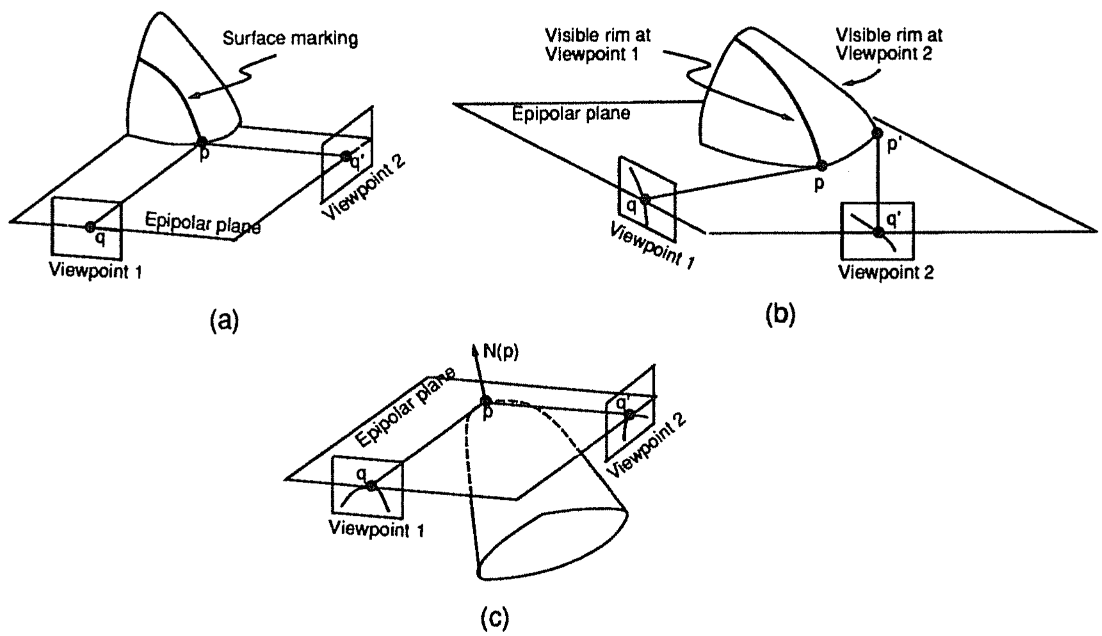


Figure 25: Point correspondences induced by the epipolar geometry. Point q' is the point corresponding to q. (a) When q is the projection of a point p on a surface marking at Viewpoint 1, q' is the projection of p at Viewpoint 2. (b) When p is on the visible rim and the tangent of the image curve at q does not belong to an epipolar plane, $T_p(S)$ is not an epipolar plane and q' is the projection of p', not p. (c) When the tangent of the image curve at q belongs to an epipolar plane, $T_p(S)$ is an epipolar plane and both q and q' are projections of p (Chapter 3).

frames, all points corresponding to q must be projections of the *same* surface point. This property does not depend on whether or not the image curve containing q is the projection of a surface marking.

In the following, if p is a surface point projecting to q at the initial viewpoint, we denote by $p(t)$ and $q(t)$ the points corresponding to p and q, respectively, at viewpoint $\xi(t)$.

4.2.1 Affine-Invariant Representations

A basic step in our method for determining the non-stationarity of the visible rim points is that of *re-projection* [19, 148, 167]: Given the projections of a collection of 3D points along a sequence of viewing directions, compute the projection of those points along a viewing direction that is not contained in the sequence. Affine-invariant representations are important because they allow us to re-project points without knowing the viewpoint motion parameters and without recovering any metric properties of the corresponding surface points. The related theory was introduced and applied in computer vision only very recently [21, 54, 58, 95, 118, 173, 175], and was first applied to the occluding contour detection problem in [193].

Let $p_1, \ldots, p_n \in \Re^3, n \geq 4$, be a collection of points at least four of which are not coplanar. An *affine-invariant representation* of those points is a representation that does not change if the same non-singular linear transformation (e.g., translation, rotation, scaling) is applied to all the points. The representation consists of three components: The *origin*, p_o, which is one of the points $p_i, i = 1, \ldots, n$; the *affine basis points*, $p_{b_1}, p_{b_2}, p_{b_3}$ which are three points from the collection that are not coplanar with the origin; and the *affine coordinates* of the points p_i, which are the coordinates of $p_i - p_o$ with respect to the affine basis vectors $\mathbf{b}_j = p_{b_j} - p_o, j = 1, 2, 3$. To re-project, we use the following two properties of affine-invariant representations:

Property 4.1 *When the projection of the origin and basis points is known along a viewing direction $\xi(t)$, we can compute the orthographic projection of a point p_i from its affine coordinates.*

Property 4.2 *The affine coordinates of $p_1, \ldots, p_n$ can be computed when their projection along at least two viewing directions is known.*

More specifically, the affine basis vectors $\mathbf{b}_j, j = 1, 2, 3$ constitute a basis for $\Re^3$. If $B = [\ \mathbf{b}_1\ \ \mathbf{b}_2\ \ \mathbf{b}_3\ ]$ is the invertible 3x3 matrix describing the affine basis, the relation between a point p and its affine coordinates can be expressed in matrix form as

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} - p_o = \mathbf{B} \begin{bmatrix} \alpha_p^1 \\ \alpha_p^2 \\ \alpha_p^3 \end{bmatrix} \tag{1}$$

where $[\ x\ \ y\ \ z\ ]^T$ and $[\ \alpha_p^1\ \ \alpha_p^2\ \ \alpha_p^3\ ]^T$ are the Euclidean and affine coordinates of p, respectively.

Let $[\ u_p^k\ \ v_p^k\ ]^T$ be the projection of p along viewing direction $\xi_k,\ k = 1, 2$, and let $[\ u_{\mathbf{b}_j}^k\ \ v_{\mathbf{b}_j}^k\ ]^T$ be the projection of the basis vector $\mathbf{b}_j,\ j = 1, 2, 3$. Computation of the affine coordinates of p is achieved using the formula:

$$(2) \qquad \begin{bmatrix} u_p^k \\ v_p^k \end{bmatrix} = \begin{bmatrix} u_{\mathbf{b}_1}^k & u_{\mathbf{b}_2}^k & u_{\mathbf{b}_3}^k \\ v_{\mathbf{b}_1}^k & v_{\mathbf{b}_2}^k & v_{\mathbf{b}_3}^k \end{bmatrix} \begin{bmatrix} \alpha_p^1 \\ \alpha_p^2 \\ \alpha_p^3 \end{bmatrix} + \begin{bmatrix} u_{p_o}^k \\ v_{p_o}^k \end{bmatrix}, k = 1, 2$$

This formula gives rise to two linear equations for each viewing direction. Three of these equations can be used to solve for the affine coordinates of p. When the affine coordinates of p and the projection of $\mathbf{b}_j$ are known, re-projection is achieved through a simple matrix multiplication using Eq. 2.

When the affine coordinates of all points $p_1, \ldots, p_n$ are sought, and when the projection of those points from m viewing directions is available, the affine coordinates of those points can be found by solving an overdetermined system of linear equations:

$$(3) \qquad \begin{bmatrix} u_{p_1}^1 & \cdots & u_{p_n}^1 \\ \vdots & & \vdots \\ u_{p_1}^m & \cdots & u_{p_n}^m \\ v_{p_1}^1 & \cdots & v_{p_n}^1 \\ \vdots & & \vdots \\ v_{p_1}^m & \cdots & v_{p_n}^m \end{bmatrix} = \begin{bmatrix} u_{\mathbf{b}_1}^0 & u_{\mathbf{b}_2}^0 & u_{\mathbf{b}_3}^0 \\ \vdots & \vdots & \vdots \\ u_{\mathbf{b}_1}^m & u_{\mathbf{b}_2}^m & u_{\mathbf{b}_3}^m \\ v_{\mathbf{b}_1}^0 & v_{\mathbf{b}_2}^0 & v_{\mathbf{b}_3}^0 \\ \vdots & \vdots & \vdots \\ v_{\mathbf{b}_1}^m & v_{\mathbf{b}_2}^m & v_{\mathbf{b}_3}^m \end{bmatrix} \begin{bmatrix} \alpha_{p_1}^1 & \cdots & \alpha_{p_n}^1 \\ \alpha_{p_1}^2 & \cdots & \alpha_{p_n}^2 \\ \alpha_{p_1}^3 & \cdots & \alpha_{p_n}^3 \end{bmatrix}$$

These derivations were used by Koenderink [95], Weinshall [173], and Weinshall and Tomasi [174] to compute the affine representation of a collection of points.

4.3 Detecting Point Non-Stationarity

The previous section showed that a fundamental property distinguishing the visible rim from surface markings is *non-stationarity*: Except for a finite (and, in practice, small) collection of points that depends on the viewpoint's motion plane, the points on the visible rim are non-stationary. Hence, we can identify the occluding contour curves by controlling viewpoint to determine which image curves contain projections of non-stationary points.

Our approach is based on a theorem that allows us to distinguish between the projection of stationary and non-stationary points by exploiting a few simple properties of smooth surfaces and their occluding contours. We only consider the case where the surface is initially viewed from a "general viewpoint," i.e., a viewpoint for which an arbitrarily small perturbation does not produce topological changes of the visible rim and the occluding contour. We also restrict our analysis to the problem of determining the non-stationarity property for points that are not endpoints of a curve in the image.

Suppose viewing direction changes on a single plane according to a smooth function $\xi(t)$, let $p(0)$ be the surface point projecting to $q(0)$, and let E be the epipolar plane through $p(0)$. Theorem 4.1 characterizes the distance between $q(t)$ and the projection, $\hat{q}(t)$, of $p(0)$ along $\xi(t)$ (Figure 26). See Appendix A for proofs.

Theorem 4.1 *Points $p(t)$, $q(t)$, and $\hat{q}(t)$ have the following properties:*

(1) Suppose that the curves in $S \cap E$ are parameterized so that their curvature is positive when their normal is toward the surface interior. Then, $p(t)$ is contained in the maximal, connected, convex subset of $S \cap E$ that contains $p(0)$.

(2) Let λ be the smooth curve corresponding to the set in (1). The distance between $q(t)$ and $\hat{q}(t)$ is given by

$$\|q(t) - \hat{q}(t)\| = \begin{cases} 0 & \text{if } p(0) \text{ is stationary,} \\ |\,[p(t) - p(0)] \cdot n(p(t))| & \text{if } p(0) \text{ is non-stationary,} \end{cases} \tag{4}$$

where $n(p(t))$ is the normal[2] of curve λ at $p(t)$.

(3) If $p(0)$ is non-stationary, $\|q(t) - \hat{q}(t)\|$ is zero along at most three viewing directions on E, with $\xi(0)$ being one of them.

Theorem 4.1 motivates the use of a prediction-verification scheme for determining whether or not a point $p(0)$ is stationary: If we are able to compute $p(0)$'s projection along any viewing direction on the motion plane, we can assume that $p(0)$ is stationary and then check the validity of that assumption. The stationarity of $p(0)$ can be verified by moving to a new viewing direction $\xi(t)$ and then verifying that $\hat{q}(t)$, i.e., the *predicted* position of $q(t)$ under the stationarity assumption, coincides with $q(t)$ in the new

[2] In order to distinguish curve normals from surface and plane normals we use lowercase n for the former and uppercase N for the latter.

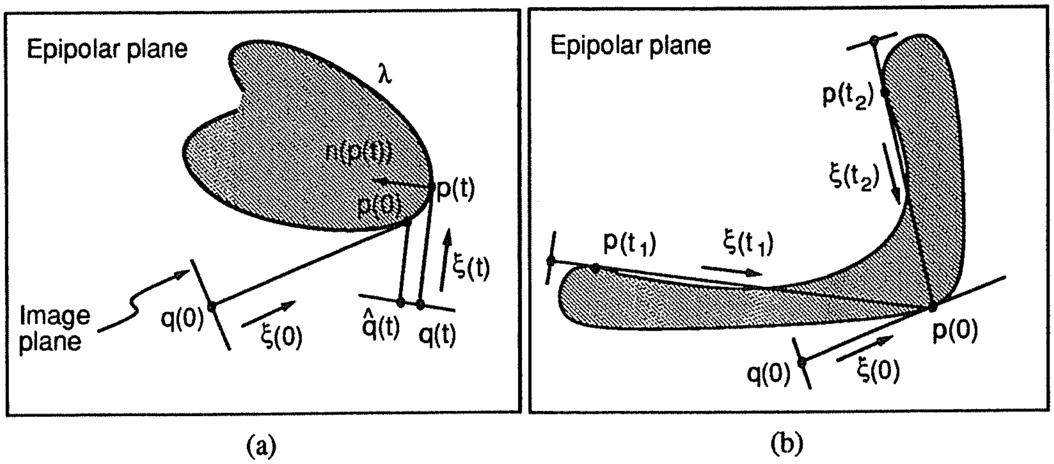


Figure 26: Distinguishing stationary from non-stationary points. (a) A "top" view of the epipolar plane is shown. The shaded area corresponds to the surface interior. λ is the open curve corresponding to the set of Theorem 4.1(1). (b) A case where three viewing directions, $\xi(0), \xi(t_1)$ and $\xi(t_2)$ force points $q(t)$ and $\hat{q}(t)$ to coincide according to Theorem 4.1(3).

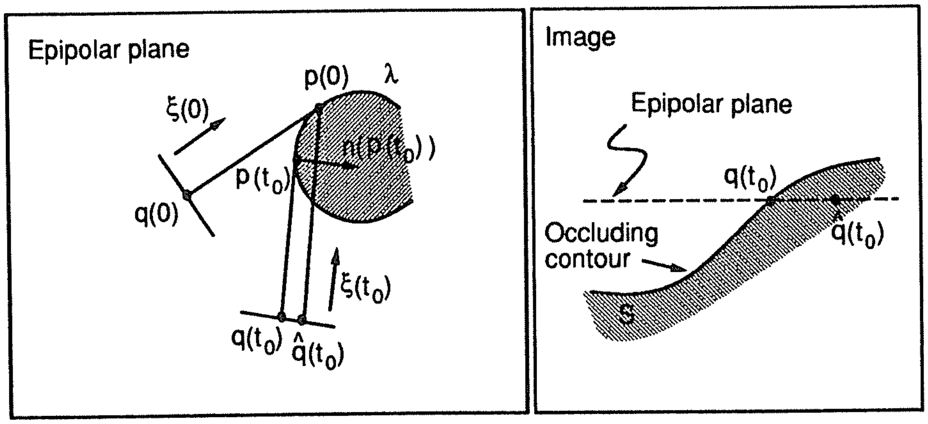
image. If $q(t)$ and $\hat{q}(t)$ do not coincide, $p(0)$ must be a non-stationary point. Intuitively, if not equal to zero, the distance $\|q(t) - \hat{q}(t)\|$ measures the "degree" of non-stationarity of $p(0)$.

The following corollary to Theorem 4.1 goes a step further, showing that the position of $\hat{q}(t)$ can serve as a qualitative indicator of the sidedness of the occluding contour (Figure 27):

Corollary 4.1 *If $p(0)$ is non-stationary and $\xi(t_0)$ is such that $\|q(t) - \hat{q}(t)\| \neq 0$ for $0 < t \leq t_0$, the surface must lie on the side of $q(t_0)$ containing $\hat{q}(t_0)$.*

Theorem 4.1 tells us that even an arbitrarily small change in viewing direction on E will force $\|q(t) - \hat{q}(t)\|$ to become non-zero. Even though, in theory, this should be sufficient for identifying the non-stationary points in an image, in practice we must allow for errors in image measurements and for the inability to measure arbitrarily small distances in an image. The next corollary to Theorem 4.1 characterizes the effectiveness of this prediction-verification scheme by specifying how large $\|q(t) - \hat{q}(t)\|$ can be (Figure 28):

Corollary 4.2 *The function $f(t) = \|q(t) - \hat{q}(t)\|$ is maximized when $[p(t) - p(0)] \cdot \xi(t) = 0$, i.e., when*



(a) (b)

Figure 27: Determining the sidedness of the contour. (a) Since λ is convex, the vector $\hat{q}(t) - q(t)$ points in the direction of λ's normal, $N(p(t))$, at $p(t)$. The corresponding image is shown in (b). The epipolar plane is always viewed "edge-on", and hence it projects to a line in the image.

$p(t)$ is at a local maximum distance from $p(0)$. (1) If at least one such point exists on λ then there is a point $p(t^*)$ corresponding to a maximum of f, such that

$$f(t^*) \geq \frac{|\sin \phi|}{\kappa_{max}} \tag{5}$$

where κ_{max} is the maximum absolute principal curvature of the surface, and ϕ is the minimum angle between E's normal and the surface normal along λ. (2) If $f(t)$ is only maximized at the endpoints of λ,

$$f(t^*) \geq \min\{D(p_{mid}, l_1), D(p_{mid}, l_2)\} \tag{6}$$

where p_1, p_2 are the two endpoints of λ; l_1, l_2 are the tangent lines at p_1, p_2, respectively; p_{mid} is the point on λ that is equidistant from p_1 and p_2; and $D(p, l)$ is the distance from point p to line l.

Theorem 4.1 and its corollaries demonstrate that by computing $p(0)$'s projection along a new viewing direction on the epipolar plane we can identify the occluding contour and determine its sidedness. Furthermore, by appropriately choosing this viewpoint we can maximize the contour's detectability, making it depend on global surface properties. We discuss the implications of the above results in the next section, where we describe a strategy that continuously controls viewpoint to detect the contour.

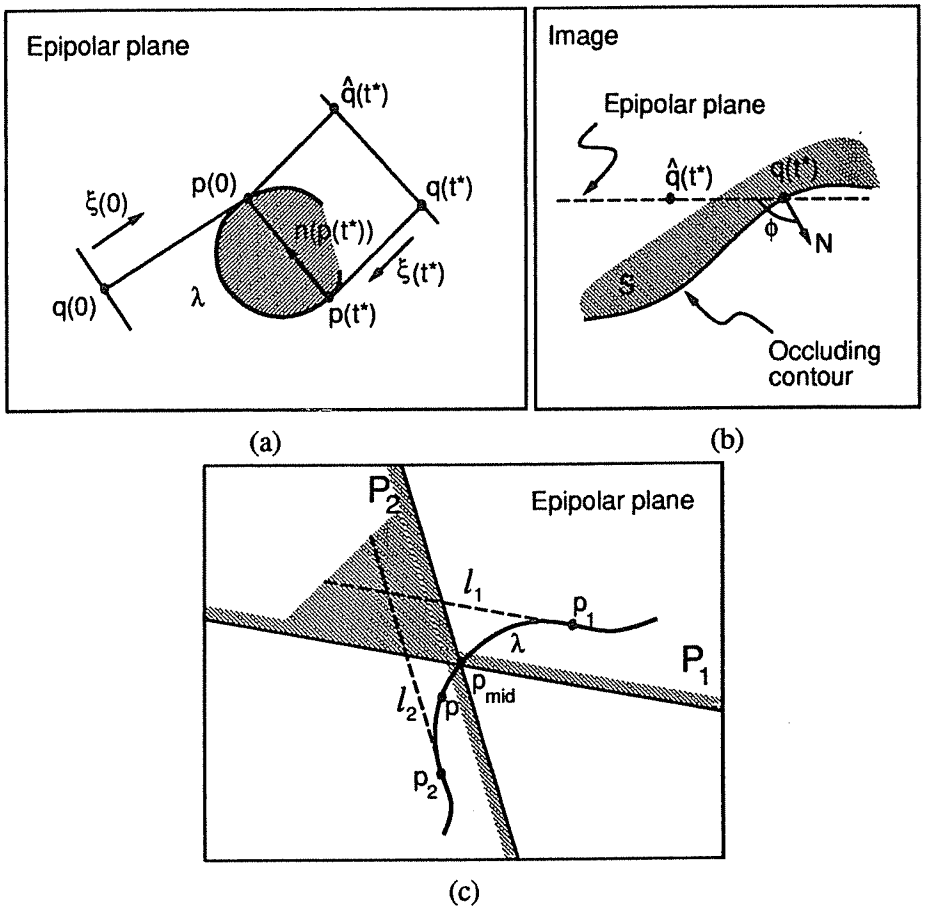


Figure 28: Bounding the distance between $q(t)$ and $\hat{q}(t)$. (a) A viewing direction maximizing the distance between $q(t)$ and $\hat{q}(t)$. When the viewing direction is along $\xi(t^*)$, the image plane is parallel to the normal of λ at $p(t^*)$. (b) The image of the surface corresponding to (a). N is the projection of the surface normal in the image plane and is along the normal of the occluding contour at $q(t)$. (c) The shaded area of the epipolar plane corresponds to the intersection of the two half-planes, P_1, P_2, of the points p satisfying the inequalities $D(p, l_1) \leq D(p_{mid}, l_1)$ and $D(p, l_2) \leq D(p_{mid}, l_2)$, respectively.

4.4 Active Occluding Contour Detection

The goal of our approach is to control viewpoint in an image-driven fashion so that the identity (visible rim or surface marking) of the surface curves projecting to the image can be determined. We do this by assuming the curves projecting to the image are stationary, selecting a motion plane, and then controlling viewpoint on that plane to verify the stationarity assumption. This process involves verifying that the actual projection of the surface curves coincides with that predicted under the assumption they are stationary. To completely specify our occluding contour detection strategy we therefore need to answer three questions: (1) How to select the motion plane, (2) how to control viewing direction on that plane, and (3) how to predict the projection of the selected curves from viewpoints on that plane under the stationarity assumption? Below we first outline the main ideas, and in Sections 4.5 and 4.6 present more details.

4.4.1 An Active Strategy for Detecting the Occluding Contour

Suppose that the viewpoint's motion plane has been selected and that a method is available for predicting the projection of a collection of surface curves $\beta_1, \ldots, \beta_n$ from viewpoints on the selected motion plane. We first consider how to control the viewing direction to determine the non-stationarity of a single point $p(0)$ on the visible rim at the initial viewpoint.

Let $q(0)$ be $p(0)$'s projection. Theorem 4.1 tells us that there are viewing directions on the selected motion plane from which the non-stationarity of $p(0)$ can be established by verifying that $p(t)$'s projection does not coincide with that of $p(0)$. Since the distance between the two projections can be arbitrarily small depending on the viewpoint, we use a strategy to reach the special viewpoints for which this distance is maximized. Corollary 4.2 shows that in most cases, the maximum distance can be characterized by only two parameters, one of which depends only on the intrinsic surface geometry.

Let E be the selected motion plane. Viewpoints on E can be thought a points on a unit circle C on E (Figure 29). As the viewing direction changes on E, the corresponding point moves on C. Our goal is to smoothly move this point on C until the viewpoint maximizing the distance between $q(t)$ and the projection, $\hat{q}(t)$, of $p(0)$ is reached. There are only two possibilities for moving on the unit circle, either clockwise or counter-clockwise. Clearly, we can move on C as long as correspondences for $q(0)$

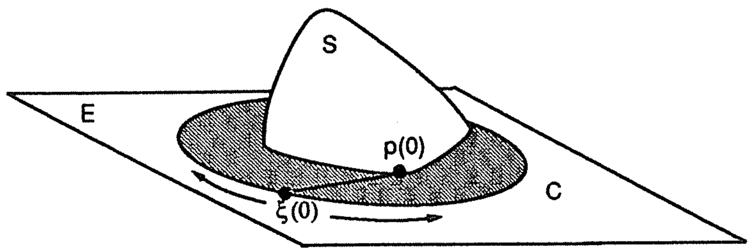


Figure 29: Changing viewing directions on E. Viewing directions correspond to points on the unit circle C contained in E.

can be established, i.e., as long as the curve λ containing $p(0)$ is convex, and as long as $p(t)$ is not occluded. Corollary 4.2 says that because the distance between $q(t)$ and $\hat{q}(t)$ can have local maxima, the motion must be performed both in the clockwise and counter-clockwise directions if a global distance maximum is sought.

When the goal is to identify the type of each curve in a collection of curves in an image, the above process can be applied simultaneously to all points on the selected curves. These considerations lead to the following strategy for identifying the occluding contour curves:

Active Occluding Contour Detection Strategy

Step 1: Let $B_1, \ldots, B_n$ be the image curves whose identities are sought.

Step 2: Select the motion plane E.

Step 3: Perform a small clockwise viewing direction change.

Step 4: For every point $q(0)$ on $B_1(0), \ldots, B_n(0)$ for which a correspondence $q(t)$ can be established along the current viewing direction,

a. compute $\hat{q}(t)$,

b. compute the distance $\delta(q(t)) = \|q(t) - \hat{q}(t)\|$.

Step 5: Repeat Steps 3-5 until either the initial viewing direction is reached, or no correspondences can be established for curves $B_1(0), \ldots, B_n(0)$.

Step 6: If the initial viewing direction is not reached, repeat Steps 4-5 while changing viewing direction in a counter-clockwise fashion.

Step 7: Given the computed distances $\delta(q(t)), 0 \leq t \leq T$, for each point $q(0)$ on the curves $B_1, \ldots, B_n$, label these curves as either type surface markings or type occluding contour.

In the next section we show how we can use tangential viewpoint control to construct an affine invariant representation and perform Step 4. Section 4.6 then discusses the curve classification process in Step 7, which takes into account noise in image measurements.

4.5 A Prediction Mechanism for Detecting Non-Stationarity

The occluding contour detection strategy described in the previous section made an important assumption: Given a point $p(0)$ on the surface, we can compute $p(0)$'s projection along arbitrary viewing directions on its motion plane. In general, this computation cannot be performed unless the three-dimensional coordinates of that point are known. Here we show that by exploiting the geometry of tangential motion developed in Chapter 3 and the Tangential Motion Property (Section 4.2) to select the viewpoint's motion plane, we can formulate this computation as a re-projection; this allows the assumption to be slightly relaxed without affecting the correctness of the strategy given in Section 4.4.

Suppose we want to determine whether or not a point $p(t_1)$ projecting to $q(t_1)$ is non-stationary. Theorem 4.1 says that if we select a plane through $p(t_1)$ that is not tangent to $p(t_1)$ and change our viewing direction on that plane to $\xi(t)$, we can determine that $p(t_1)$ is non-stationary by verifying that the predicted projection, $\hat{q}(t)$, of $p(t)$ along $\xi(t)$ does not coincide with the measured one, i.e., $q(t)$. To exploit this theorem without having to explicitly compute $\hat{q}(t)$ we simply note that the exact distance between $\hat{q}(t)$ and $q(t)$ is not important, as long as we can determine that it is non-zero. We use the following result, which extends Theorem 4.1:

Theorem 4.2 *Let $p(t_1)$ be a surface point and let $q(t_1)$ be its projection along $\xi(t_1)$. Let $q(t_2)$ be the point corresponding to $q(t_1)$ along a viewing direction $\xi(t_2)$ on the motion plane, E. If (1) the projections, $r_1(t_1), \ldots, r_4(t_1)$, of four stationary and non-coplanar surface points can be identified in the image along $\xi(t_1)$, and (2) $[\ \alpha_1 \quad \alpha_2 \quad \alpha_3\ ]^T$ is the affine coordinate vector computed from Eq. 2* *with* $[\ u_p^k \quad v_p^k\ ]^T = q(t_k), k = 1, 2,$ *and* $[\ u_{b_j}^k \quad v_{b_j}^k\ ]^T = r_j(t_k) - r_1(t_k), j = 2, 3, 4,$ *then the point* $\tilde{q}(t)$ *defined by*

$$\tilde{q}(t) = [\ r_2(t) - r_1(t) \quad r_3(t) - r_1(t) \quad r_4(t) - r_1(t)\ ] \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} \tag{7}$$

has the following two properties:

1. *Let* λ *be defined as in Theorem 4.1, and let* $\tilde{p}$ *be the 3D point with affine coordinates* $[\ \alpha_1 \quad \alpha_2 \quad \alpha_3\ ]^T$. *The distance between* $q(t)$ *and* $\tilde{q}(t)$ *is given by*

$$\|q(t) - \tilde{q}(t)\| = \begin{cases} 0 & \text{if } p(t_1) \text{ is stationary,} \\ |\,[p(t) - \tilde{p}] \cdot n(p(t))| & \text{if } p(t_1) \text{ is non-stationary,} \end{cases} \tag{8}$$

   where $n(p(t))$ *is the normal of curve* λ *at* $p(t)$.

2. *If* $p(t_1)$ *is non-stationary,* $\|q(t) - \tilde{q}(t)\|$ *is zero along at most four viewing directions on* E.

Theorem 4.2 implies that if we can identify the projections of four non-coplanar stationary points in the image, we can replace the computation of $\hat{q}(t)$ in Step 4a of the Active Occluding Contour Detection Strategy by computing $\tilde{q}(t)$ without compromising the correctness of the strategy. The first property of $\tilde{q}(t)$ ensures that by computing $\tilde{q}(t)$ instead of $\hat{q}(t)$ we do not introduce any "false positives" in our occluding contour detection process. The second property shows that the distance $\|q(t) - \tilde{q}(t)\|$ behaves almost identically to the "true" distance $\|q(t) - \hat{q}(t)\|$, allowing us to use the Active Occluding Contour Detection Strategy to identify the occluding contour. The following corollary, which is similar to Corollaries 4.1 and 4.2, makes this explicit:

Corollary 4.3 *(1) The function* $\tilde{f}(t) = \|q(t) - \tilde{q}(t)\|$ *is maximized when* $[p(t) - \tilde{p}] \cdot \xi(t) = 0$, *i.e., when* $p(t)$ *is a local maximum distance from* $\tilde{p}$. *If at least one such point exists on* λ *and that point is not one of its endpoints, then there is a point* $p(t^*)$ *corresponding to a maximum of* $\tilde{f}$, *such that*

$$\tilde{f}(t^*) \geq \frac{|\sin\phi|}{\kappa_{max}} \tag{9}$$

where κ_{max} is the maximum absolute principal curvature of the surface, and ϕ is the minimum angle between E's normal and the surface normal along λ. (2) Let $\xi(t^*)$ be a viewing direction maximizing $\tilde{f}$. If $p(0)$ is non-stationary, the surface must lie on the side of $q(t^*)$ containing $\tilde{q}(t)$.

The proof of Corollary 4.3 is similar to that of Corollaries 4.1 and 4.2.

Corollary 4.3 shows that the only major impact of computing $\tilde{q}(t)$ instead of $\hat{q}(t)$ is in the determination of the contour's sidedness. Intuitively, the contour's sidedness can be correctly determined only along viewing directions at or close to the global maximum of $\|q(t) - \tilde{q}(t)\|$.

To use Theorem 4.2 and Corollary 4.3 we must be able to identify the projections of four stationary points in the image. In general, when the viewpoint's motion cannot be controlled this is impossible to achieve. However, when we can select the viewpoint's motion plane, the Tangential Motion Property can be exploited: If four points in the first image can be identified whose tangents are parallel, the motion plane defined by those tangents guarantees the stationarity of those points in the orthographic projection model. This solves the problem of predicting the projection, $\hat{q}(t)$, of $p(t)$ in Step 4a of the Active Occluding Contour Detection Strategy under the assumption that $p(0)$ is stationary.

Theorem 4.2 shows that at least three views, $\xi(t_1), \xi(t_2), \xi(t_3)$, are required to determine that a point $p(t_1)$ projecting to $q(t_1)$ belongs to the visible rim: $\xi(t_1), \xi(t_2)$ are necessary for computing $\tilde{q}(t_3)$ along $\xi(t_3)$, and $\xi(t_3)$ is necessary for computing the distance $\|q(t_3) - \tilde{q}(t_3)\|$. The need for three views was also noted by Vaillant and Faugeras [170], where the occluding contour was identified by first computing the shape of the surface in the vicinity of the contour using trinocular stereo.

4.6 Measurement Errors

The final step in the Active Occluding Contour Detection Strategy is the classification of image curves based on whether or not they contain projections of non-stationary points. This requires determining if a specific quantity, namely $\|q(t) - \tilde{q}(t)\|$, is zero or not. In practice, due to errors in image measurements, this quantity may not be zero even for projections of stationary points. It is therefore necessary to account for such errors in the classification process. We accomplish this in three ways: (1) We use many views to determine the non-stationarity of each point $p(0)$ projecting to the initial image, (2) we estimate the probability that $p(0)$ is non-stationary under the assumption the image measurement errors have a zero-mean Gaussian distribution, and (3) to classify a curve, we estimate the above probability for all points projecting to that curve. We briefly outline this classification process below.

To incorporate multiple frames in the classification of a point $p(0)$ projecting to $q(0)$, we compute the affine coordinates of $\tilde{q}$ as the least-squares solution to the overdetermined linear system given by Eq. 3 [67, 174]. More specifically, if W is the $2K \times 3$ matrix of image coordinates of the affine basis at K viewpoints, and q is the vector collecting $q(t)$'s coordinates at those viewpoints, we compute the affine coordinate vector a of $\tilde{p}$ by

$$\mathrm{a} = \left(W^T W\right)^{-1} W^T \mathrm{q}. \tag{10}$$

The projection of $\tilde{p}$ at the K viewpoints is then given by

$$\tilde{\mathrm{q}} = W \left(W^T W\right)^{-1} W^T \mathrm{q}, \tag{11}$$

where $\tilde{\mathrm{q}}$ is the vector collecting the image coordinates of $\tilde{p}$.

To classify point $p(0)$ as non-stationary, we compute the *squared Mahalanobis distance*, $\|\mathrm{q} - \tilde{\mathrm{q}}\|^2$, of the residual $\mathrm{q} - \tilde{\mathrm{q}}$ from $\mathbf{0}$, assume that this distance is corrupted by noise in the image measurements, and then estimate the probability that it is non-zero [190]. If this probability is high, $p(0)$ and $q(0)$ are classified as being on the visible rim and the occluding contour, respectively.

To simplify the error analysis we only allow for errors in the measurement of q.[3] We also assume the errors in each element of q are independent and have a Gaussian distribution with zero mean and variance σ. Under these assumptions, and when $p(0)$ is stationary, the distance $(1/\sigma^2)\|\mathrm{q} - \tilde{\mathrm{q}}\|^2$ follows a χ^2 distribution with three degrees of freedom.[4]

To perform point and curve classification we now have the following criteria:

Point Classification Criterion: Classify $q(0)$ as an occluding contour point if

$$\Pr\left[\frac{1}{\sigma^2}\|\mathrm{q} - \tilde{\mathrm{q}}\|^2 > \delta \mid p(0) \text{ is stationary}\right] < m \tag{12}$$

where m is an *a priori* defined constant, δ is the computed value of $(1/\sigma^2)\|\mathrm{q} - \tilde{\mathrm{q}}\|^2$, and the probability is computed using a χ^2 distribution table.

Curve Classification Criterion: Classify curve B as an occluding contour curve if more than $M\%$ of the points on B are classified as being on the occluding contour.

---

[3] However, a more sophisticated error analysis that takes into account errors in W is also possible (e.g., see [170]).

[4] This follows from the non-coplanarity of the basis points, which ensures that the rank of W is three [50, 190].

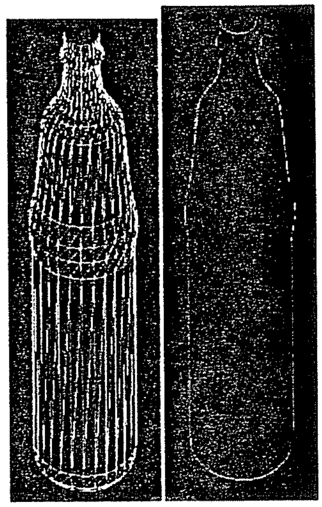


Figure 30: Polyhedral model of a bottle and its visible rim.

4.7 Experimental Results

To demonstrate the effectiveness of our occluding contour detection method we have performed preliminary experiments using both simulated and real data. We implemented a system that (1) detects and tracks image curves across frames, (2) detects and tracks the curve points whose tangents are parallel to the viewpoint's motion plane, (3) selects an affine basis from those points, and (4) classifies the curves using our Active Occluding Contour Detection Strategy.

Figure 30 shows the model used in our simulations, a polyhedral representation of a bottle. Four points were randomly selected from the polyhedron's vertices to be the affine basis points. The object was rotated a total of π radians about a vertical axis through the center of the bottle, and the projection of the affine basis points was computed for each of the frames. The generated images and the projections of the basis points were the only inputs to the system. The affine coordinate computations were performed using the first two frames of the sequence. This simulation corresponds to the best-case scenario for our strategy: Due to the model's symmetry, the occluding contour remains unchanged throughout the model's rotation, while the projection of a stationary surface curve varies with viewpoint. The simulation results are shown in Figure 31. They show that the difference between the actual and predicted positions

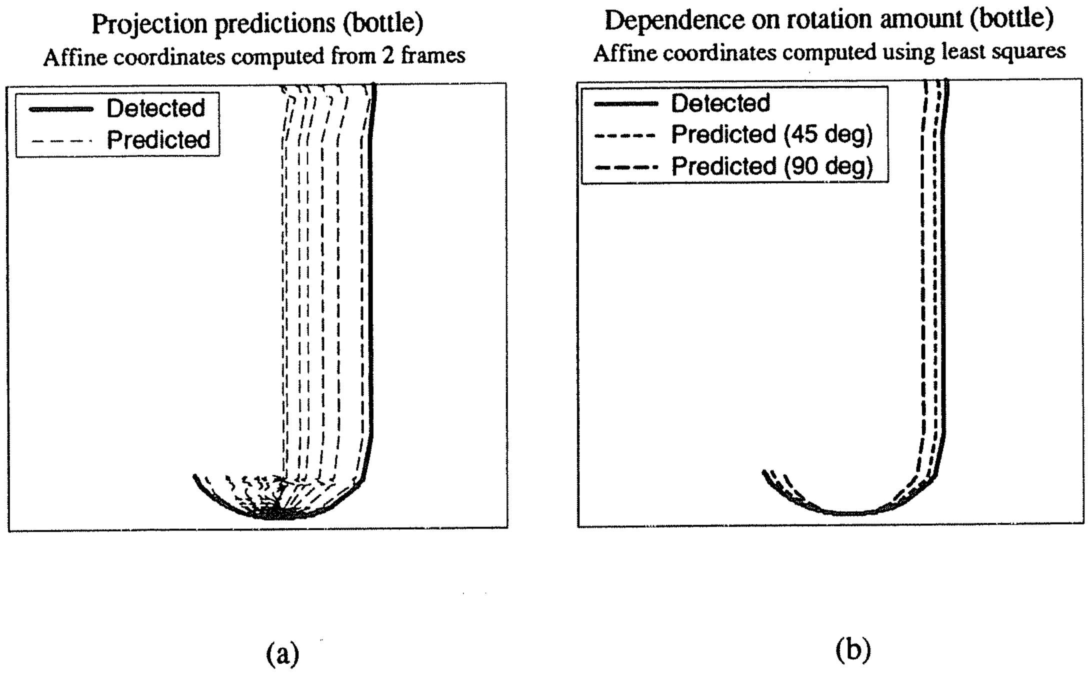


Figure 31: (a) Predictions for 10 frames overlayed with the detected curve which does not move throughout the rotation. (b) Curve predictions for the first frame when 10 frames (rotation of $\pi/4$ radians) and 20 frames (rotation of $\pi/2$ radians) are used for the affine computations (Eq. 3). respectively. Note that the distance between the predicted and actual position of the curves is smaller when least-squares is used due to its smoothing effect. The distance also becomes smaller as the surface normal approaches the axis of rotation, as stated by Corollary 4.3.

of the image curves can be dramatic, in this case almost equal to the radius of the bottle. Note also that the predicted position of the curve is on the side of the contour where the surface lies, correctly indicating the sidedness of the contour.

Figures 32-36 show the results of applying our strategy to a real scene. The image sequence consisted of five frames, showing rotation of about $\pi/2$ radians. No information about the object's motion (apart from the direction of rotation) or the camera parameters was used. Viewing direction was changed in a horizontal plane perpendicular to the plane of the page. The system detected points with horizontal tangents and tracked them across the frames. The affine basis points were selected by minimizing the condition number of the affine basis matrix [174]. The affine computations and the point and curve classifications were performed as described in Section 4.6. We used $\sigma = 1$ and classified a curve as

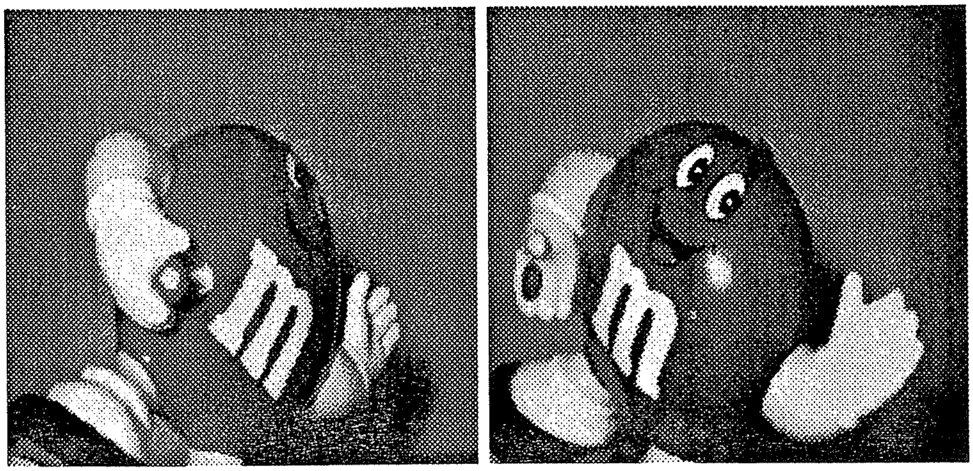
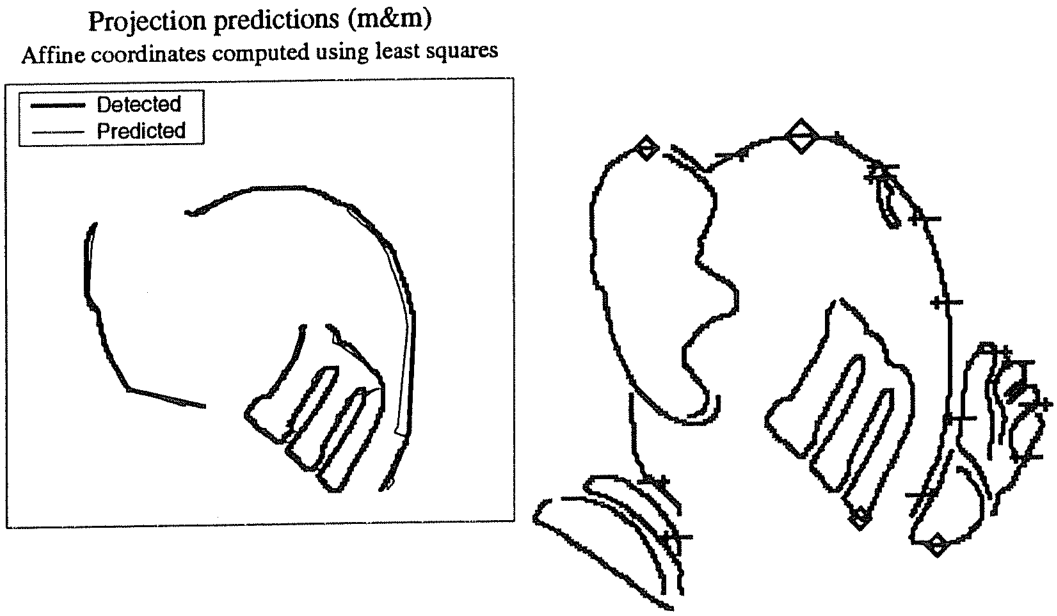


Figure 32: Two images of a rotating toy.

an occluding contour curve if more than 80% of its points had a probability greater than 90% of being non-stationary.

The results of the classification process for three of the image curves can be evaluated using the graphs in Figures 34-36. The first graph shows the variation of $||q - \tilde{q}||^2$ for the curve points with respect to their position on the curve. This quantity must be close to zero for stationary points and is the basic information used to determine a point's identity. Because we employed a very simple method for establishing point correspondences across frames, i.e., independently for each point on the curve, the second graph evaluates tracking performance. The graph plots the variance of the inter-frame matching distance for every point on the curve with respect to its position on the curve. Sharp peaks indicate tracking errors for the associated points. The third graph shows the input to the final stage of the curve classification process. It is a histogram of the number of points on the curve that have a given probability of being non-stationary. This information is used by the Curve Classification Criterion (Section 4.6) to determine the identity of the image curve. The mass under the histogram should be concentrated near 100% for non-stationary curves.

The "m" curve on the surface and the upper occluding contour curve (top of the head) were classified correctly, and the sidedness of the upper occluding contour curve was correctly determined.



(a) (b)

Figure 33: (a) The measured and predicted curves for three of the curves detected in the first frame. (b) Results of the classification process. Horizontal bars denote the occluding contour curves. The cross is on the side of the surface. The large diamond corresponds to the origin of the affine coordinate frame. The three small diamonds correspond to the points defining the three affine basis vectors.

The approache's success is also indicated by the dramatic difference in their associated probability histograms. However, due to the small amount of rotation and the high surface curvature near the visible rim, the distance between the predicted and actual curves for both occluding contour curves in Figure 33(a) was not as great as in the case of the bottle. This resulted in a misclassification of the left curve corresponding to the right arm of the object. In addition, a considerable number of points were misclassified in the "m" curve. This was due to tracking errors, indicated by the strong correlation between errors in $\|q - \tilde{q}\|^2$ and sharp peaks in the variance of the inter-frame point matching distance (e.g., around point position 250). We do not currently use this correlation information to classify points. In the case of the upper occluding contour curve, a number of points were assigned low probability of being non-stationary. This is not an error; these points are near the top of the head, where the surface normal is close to that of the motion plane.

A number of observations can be made from these experiments. First, a curve may not contain points of only one type. For example, the upper occluding contour curve contains points on its bottom end that are not occluding contour points. However, a sharp drop in the prediction error could be used to decide where to segment such curves. Second, the results show that the approach is more effective with large motions. In effect, the approach factors out the need for measuring differential properties of the viewed surface by observing large-scale effects of viewpoint control on the surface's occluding contour. Of course, large motions require reliable curve tracking across many frames, as well as consistent tracking of points with tangents parallel to the observer's tangent plane. Tracking of such points will not, in general, persist for large viewing direction changes; however, if at least four tangency points are visible during the observer's motion (not necessarily the same ones throughout), one can use an approach similar to that of Tomasi and Kanade [162] to handle occlusions by changing the affine basis points as they become occluded. Further extensions can be made to improve tracking performance and reduce errors in the computation of $||\mathbf{q} - \tilde{\mathbf{q}}||$ by tracking curves rather than points.

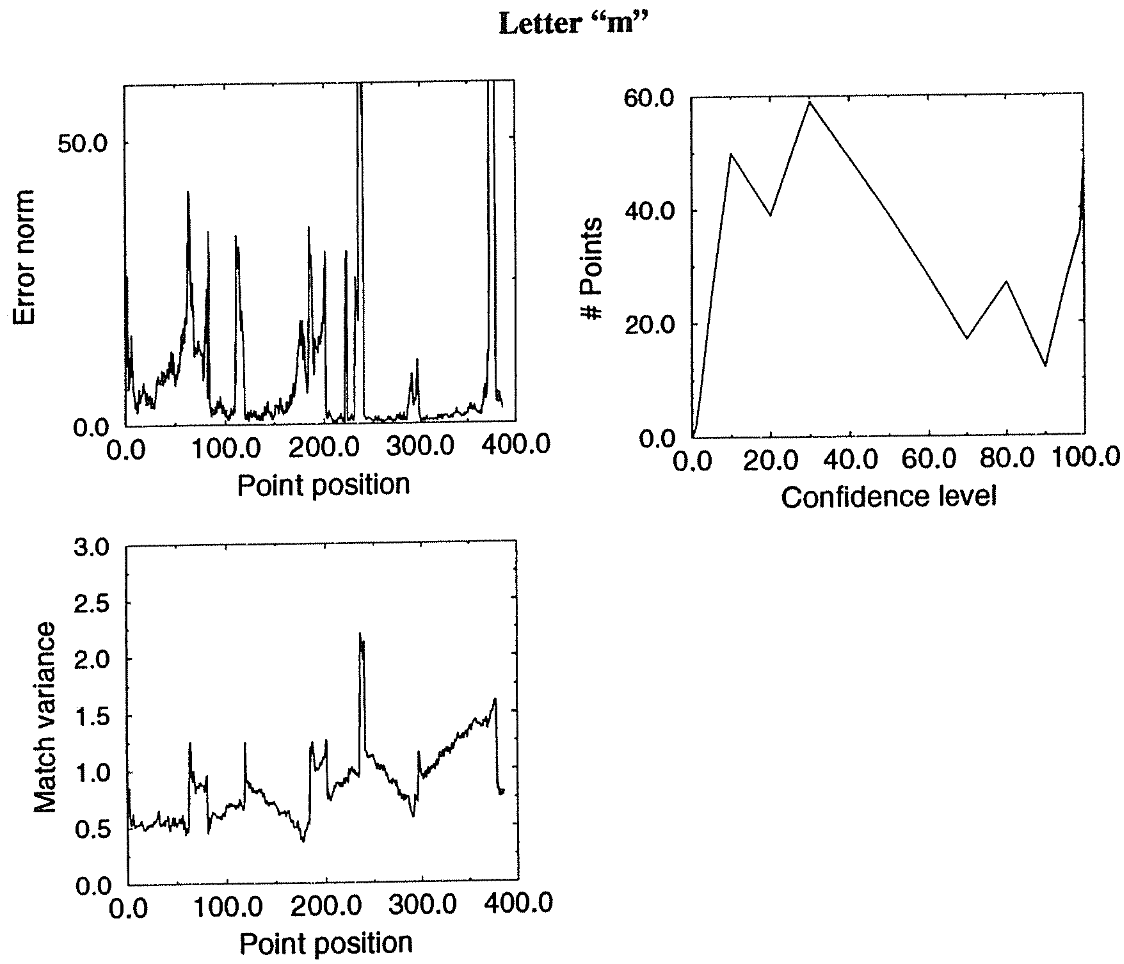


Figure 34: *Left top:* Plot of the squared Mahalanobis distance between predictions and measurements for every point on the curve corresponding to the letter "m." *Left bottom:* The variance of the distance in image positions between matched points in consecutive frames. *Right top:* Histogram of the number of points on each curve having a given probability of being non-stationary.

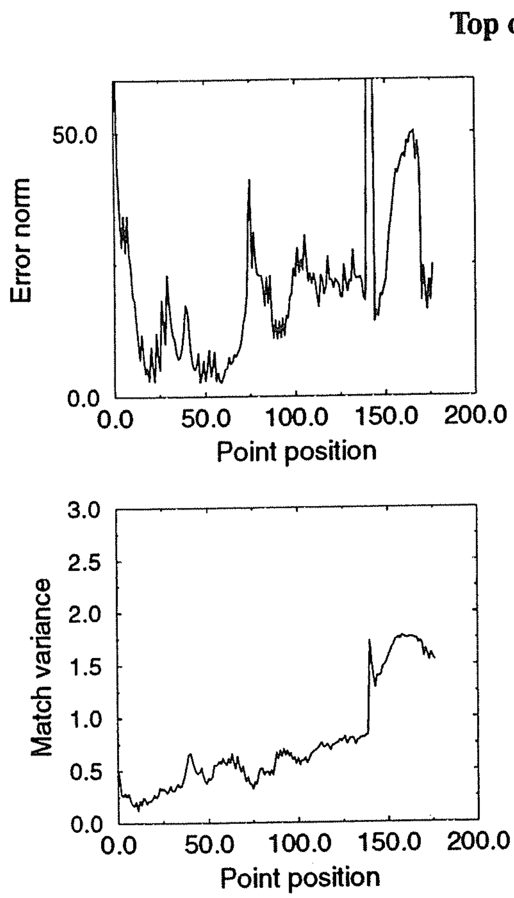


Figure 35: *Left top:* Plot of the squared Mahalanobis distance between predictions and measurements for every point on the curve corresponding to the toy's head. *Left bottom:* The variance of the distance in image positions between matched points in consecutive frames. *Right top:* Histogram of the number of points on each curve having a given probability of being non-stationary.

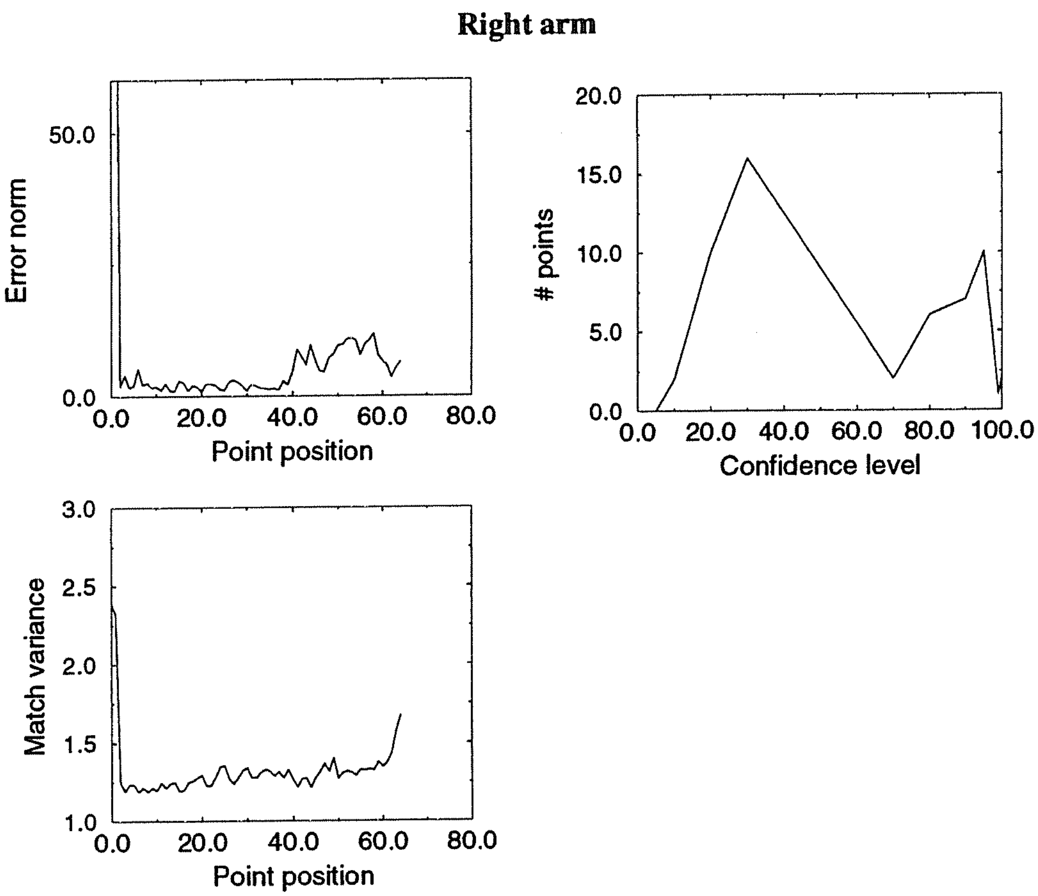


Figure 36: *Left top:* Plot of the squared Mahalanobis distance between predictions and measurements for every point on the curve corresponding to the toy's right arm. *Left bottom:* The variance of the distance in image positions between matched points in consecutive frames. *Right top:* Histogram of the number of points on each curve having a given probability of being non-stationary.

4.8 Summary

We have demonstrated that an active vision system can follow an image-driven viewpoint control strategy to identify the occluding contour edges in an image and determine their sidedness. The approach is based purely on image measurements, does not recover 3D shape or any metric scene properties, and, hence, factors out the need for differential measurements of viewpoint motion. Results from an initial implementation of this occluding contour detection process are encouraging. At the heart of the approach lies (1) the ability to control viewpoint on the tangent plane of selected points on an object's surface, which allows us to detect the contour using an invariant-based analysis, and (2) the ability to reach special viewpoints on this plane which maximize the contour's detectability.

As a side-effect, our method constructs an affine representation for a collection of fixed 3D points, on or close to the surface, whose projection can be predicted when viewpoint changes (Theorem 4.2). We believe that this representation is a very important one, and that it leads to entirely new ways of extracting information about the shape of an unknown curved object. The reason is that this fixed set of points can be thought of as a collection of "imaginary" markings, "visible" throughout the viewpoint's motion, that can serve as fixed reference points during the exploration process when no such points exist on the surface itself.

One potential application of this concept is in *parallax* measurements [109, 141]. Cipolla and Blake [44] have shown that shape estimates based on parallax measurements (i.e,. measurements relating changes between the shapes of the occluding contour and of nearby markings) are insensitive to viewpoint acceleration and rotational velocity. One aspect of our future work will be to study how shape representations for curved, featureless objects can be recovered robustly and with few or no motion measurements using such "imaginary" markings. Furthermore, because these parallax measurements are available in each frame the shape information they convey (e.g., surface curvature) can be used to improve tracking performance of the occluding contour curves by more accurately predicting the position of the occluding contour in subsequent frames, and predicting its potential connectivity changes.

Finally, even though our approach allows the contour to be distinguished from *fixed* surface curves, it cannot be used to distinguish it from other viewpoint-dependent curves, such as those resulting from specularities. Detection of such curves is an open issue, although some promising work on specularity detection has been reported recently [41, 106].

Chapter 5

Recovering Local Surface Shape

Current approaches for recovering surface curvature from the occluding contour detect the contour as a side-effect of the recovery process. The continuous, contour-driven viewpoint control framework allows an alternative approach: By considering the tasks of contour detection and curvature estimation sequentially, we can design viewpoint control strategies that are specialized to each of the two tasks. This independent, sequential treatment of the tasks simplifies the solution of both.

The previous chapter studied occluding contour detection; this chapter considers the local curvature estimation task. The specific task we address is the following: "Given a point p projecting to the occluding contour, compute the principal curvatures and principal directions of the surface at p." This quantitative exploration task allows the surface in the neighborhood of a single visible rim point to be approximated using a quadratic polynomial.

The rest of the chapter is organized as follows. Section 5.1 describes the basic ideas of our curvature estimation method and its relationship to previous work. Section 5.2 reviews basic terminology. Section 5.3 discusses the relationships we exploit between viewpoint and the geometries of the occluding contour and the surface. This section presents the major result enabling us to actively recover surface curvature from the occluding contour. Section 5.4 uses this result to describe the main shape recovery step of our approach. Our results are then extended in the following two sections. Section 5.5 shows how the approach can be used for global surface reconstruction of surfaces of revolution, and Section 5.6 considers how we can automatically select points on the surface in order to extend the single-point recovery process. Section 5.7 then presents experimental results on synthetic and real images to illustrate our geometrical arguments. Section 5.8 summarizes the main contributions of the chapter and discusses limitations and possible extensions of the approach.

5.1 Active Shape Recovery

There has been considerable interest in recovering information about the structure of an environment from sequences of images, assuming a camera in motion (e.g., work on optical flow [78] and shape-from-motion [162]). One common feature of approaches dealing with curved objects is the use of known motion (i.e., velocities and accelerations) in order to recover quantitative surface properties such as curvature [44, 82, 98, 146, 157, 170, 191, 192]; shape recovery without such motion information has been restricted to local, qualitative tasks involving the processing of one [88, 108, 114, 136] or more [193] images.

Key to the extraction of quantitative shape information from the occluding contour is the contour's dependency on viewpoint. Differential viewpoint changes affect the geometry (i.e., curvature) of the rim, and hence the occluding contour. Moreover, the set of rim points changes and therefore new constraining information about the surface shape becomes available. Giblin and Weiss [62] showed that if we know how the geometry of the occluding contour changes with viewpoint, we can derive a parameterization of the surface and determine its shape. The issue here is how to accurately measure such changes in the rim's geometry with small viewpoint changes. For example, we must be able to measure the velocity and acceleration of surface points entering and leaving the rim [44, 170], a problem that, in general, requires first- and second-order image differentiation operations, and knowledge of viewpoint velocities and accelerations, and hence is sensitive to noise. While parallax measurements can eliminate dependence on rotational velocities and on acceleration, the existence of *a priori* identified surface markings close to the visible rim is required [44].

This chapter applies the continuous, contour-driven viewpoint control framework to recover quantitative surface shape information without requiring differential motion measurements. We show that shape for a selected point on the visible rim can be recovered from the occluding contour of two views. The only requirements are that (1) the surface point projects to the occluding contour in both views, and (2) the viewpoint for one of the views has a special relationship with the surface geometry at the selected point. The idea of the approach is to employ tangential viewpoint control in order to achieve such a well-defined geometric relationship with respect to a 3D shape. We show that this relationship is characterized by specific image-computable quantities and enables a simple maximization-based analysis [99] analogous to the one described in the previous chapter.

The basic assumption used by all previous approaches was that the viewpoint and its motion are object-independent. Our approach uses continuous, contour-driven viewpoint control to obtain a view that allows exact shape information to be recovered from the occluding contour. The main step of the approach is based on a relation between the geometries of a surface in a scene and its occluding contour: If the viewpoint is along a principal direction for a selected surface point whose projection is on the contour, the corresponding principal curvature at the point can be recovered. Hence, even though in general surface curvature estimation from the occluding contour of a single view is an under-constrained problem, for any given point there do exist viewpoints that make this recovery problem well-defined. If the viewpoint can move to one of those special viewpoints, the ambiguities caused by the projection process can be resolved. We show that we can in fact deterministically move to these special viewpoints by simply maximizing or minimizing a geometric quantity of the occluding contour (curvature at a point) while changing viewpoint in a constrained way (through tangential viewpoint control). Furthermore, we show that we can recover the shape of the surface at the selected point (i.e., both principal curvatures) from the occluding contour of one additional view for which the selected point projects onto the contour. Thus viewpoint is moved to one of the special viewpoints in order to make shape recovery a well-defined problem.

The significance of the method lies in the use of continuous, contour-driven viewpoint control to achieve and maintain purely geometrical relations between a surface and its occluding contour in order to recover surface shape. Hence, there is no need to perform any velocity or acceleration measurements in the vicinity of the visible rim, a process requiring point-to-point correspondences in the images and precise knowledge of viewpoint motion. Furthermore, since there is a well-defined procedure to reach the desired viewpoint, the viewpoint control strategy does not need to perform a complicated search in order to find it.

Even though our approach is limited to the recovery of surface shape in the vicinity of a single point on the visible rim, we show that there is an important special case for surfaces of revolution, for which we can recover the shape of the entire surface. In this case the viewpoint can be actively "aligned" with the viewed surface in order to find a viewpoint giving complete surface information (i.e., one perpendicular to the surface's axis of rotation).

We also present an extension to the above approach that recovers the shape of points in the vicinity of the rim. After the shape of a selected rim point is recovered, the viewpoint is changed in order to bring a new surface point onto the rim and to recover its shape. Since our basic shape recovery step involves aligning the viewpoint with one of the principal directions at the new point, it is important for this visual alignment process to require only small viewpoint adjustments. We show that if (1) the new point selected is in the normal plane of the previously selected point, and (2) the new point is sufficiently close to the previously selected point, these adjustments will in fact be small and their extent will depend entirely on the intrinsic properties of the surface. This is a major difference from approaches using "passive" motion, where the points selected for reconstruction cannot be controlled.

5.2 Local Surface Geometry

Let S be a smooth, oriented surface in $\Re^3$, viewed under orthographic projection along a viewing direction ξ. The shape of the occluding contour depends on S and the viewing direction. Our goal is to use this contour information to recover a description for the parameterization $\mathbf{x}$ at points of S in the vicinity of the corresponding rim points.

The shape of the surface around p can be completely described by the principal curvatures k_{n_1}, k_{n_2}, and principal directions e_1, e_2, at p (Section 3.2). In particular, we can use the principal curvatures at p to compute the curvature of the normal sections along any direction on $T_p(S)$ using Euler's formula:

$$k_n(\phi) = k_{n_1} \cos^2 \phi + k_{n_2} \sin^2 \phi \tag{13}$$

where ϕ is the angle between the direction on $T_p(S)$ and e_1. Hence, we can recover the local shape of S at a point p completely from the principal curvatures of S at p.

Our goal is to recover the principal directions and principal curvatures at selected points on S. We focus on the general case where p is not an umbilic point, i.e. a point where any pair of orthogonal directions on $T_p(S)$ is a pair of principal directions; recovering the local shape of the surface at umbilic points is then straightforward. In the vicinity of non-umbilic points there exists a special parameterization $\mathbf{x}(u, v)$ of S such that the tangents to the curves $\mathbf{x}(u, v_0)$ and $\mathbf{x}(u_0, v)$ (u_0, v_0 constant) are along the principal directions. These curves are called *lines of curvature* and their properties are intrinsically related to the underlying surface. Therefore they serve as a natural basis for describing a surface [34, 153]. In the rest of the paper $\mathbf{x}$ will refer to such a parameterization.

The geometry of a point on the occluding contour and the information we can derive from it depends on whether the point is a projection of an elliptic, hyperbolic or parabolic point. This qualitative classification is therefore especially important in order to evaluate the results of our approach.

5.3 Local Surface Geometry from Occluding Contour

The problem of recovering surface geometry from the occluding contour has been mainly studied under the assumption that the viewing direction is object-independent. For example, there are surfaces for which their rim is planar when viewed from a particular set of directions [116]. The assumption that the viewpoint is arbitrary immediately excludes such viewing directions from consideration since the rim is not always planar [90]. Unfortunately we can only derive a limited amount of information from the occluding contour when this assumption is in effect.

Let p be a point on the visible rim of S when viewed from direction ξ and let q be its projection on the image plane. There are three main results describing what can be recovered from the shape of the occluding contour under orthographic projection and from an arbitrary viewing direction:

- We can recover the surface normal and the tangent plane at p from ξ and the tangent to the occluding contour at q (Section 3.2).
- Let k_o be the curvature of the occluding contour at q. Then k_o and the Gaussian curvature K of S at p have the same sign [34, 88].
- If k_n is the normal curvature of S at p along ξ,[1] then $K = k_n k_o$ [34, 62, 88].

[1] The quantity k_n is also referred to as the *radial curvature* of S at p [170].

Similar results hold for perspective projection where the plane of projection is not positioned at infinity, and for the case where $k_o = 0$ [34]. Because K is defined as the product of two curvatures on the surface (i.e., k_{n_1}, k_{n_2}), these results suggest that if we know k_o then we only need to measure one curvature on the surface instead of two. In fact k_n and k_o determine the second fundamental form at p. This was the main idea behind the surface reconstruction approach of Cipolla and Blake [44].

The above results are important but they also imply that if we have no additional information about the shape of the viewed surface, the information provided by the occluding contour is primarily qualitative. However, when the viewing direction can be actively controlled, we can exploit the existence of directions that allow the derivation of complete information about the surface. We show this by presenting three simple corollaries to a result of Blaschke [90]. Blaschke's result is analogous to Euler's formula and relates the curvature of the occluding contour with the principal curvatures of S at the rim:

Theorem 5.1 (Blaschke) *Let ϕ be the angle between ξ and the principal direction e_1 at p. Then,*

$$k_o^{-1}(\phi) = k_{n_1}^{-1} \sin^2 \phi + k_{n_2}^{-1} \cos^2 \phi \tag{14}$$

Corollary 5.1 *If ξ is along e_1, then $k_o = k_{n_2}$.*[2]

Corollary 5.2 *Let ξ, ξ' be two distinct viewing directions in $T_p(S)$ from which p is visible, and let k_o, k_o' be the curvatures of the occluding contour at the corresponding projections of p. If (1) $K \neq 0$ at p, (2) $\xi = e_1$, and (3) the angle between ξ and ξ' is known, then we can compute k_{n_1}, e_2, and K at p.*

Corollary 5.3 *Let p be a point on the visible rim of S with $K \neq 0$. Let $\phi \in [-\pi, pi)$ be the angle between $\xi \in T_p(S)$ and e_1. (1) If p is elliptic and non-umbilic, the function $k_o(\xi)$ takes its minimum and maximum values only when ξ coincides with one of the principal directions. (2) If p is hyperbolic, $k_o(\xi)$ is well-defined only when $|\phi| < \arctan\sqrt{(k_{n_1}/ - k_{n_2})}$ for $|\phi| < \pi/2$, or $\pi - |\phi| < \arctan\sqrt{(k_{n_1}/ - k_{n_2})}$ for $|\phi| \geq \pi/2$. For these directions, $k_o(\xi)$ takes its maximum value when ξ coincides with e_1 and it has no minimum value. (3) If p is umbilic, $k_o(\xi)$ is constant.*

See Appendix B for proofs. Corollary 5.1 suggests that the principal directions at p form a special set of directions providing explicit information about surface geometry in the vicinity of p. Now assume

[2] This is also mentioned in [90].

that we are viewing a point p from a particular viewing direction and can measure the curvature of the occluding contour at p's projection. If somehow we can adjust our viewing direction to coincide with a principal direction at p and know what this adjustment is, Corollary 5.2 shows that we can derive the second fundamental form of S at p. This solves the shape recovery problem for p. The most important result is given by Corollary 5.3. It shows that the problem of finding the principal directions at a point can be treated as a simple maximization (or minimization) problem. We describe the implications of this result in the next section and show how it can be used to find the principal directions at p.

5.4 Recovering the Local Geometry of a Surface Point

The basic step of our surface reconstruction approach is to select a point on the occluding contour and recover the local surface geometry for its corresponding rim point. We do not address the point selection problem directly. The reason for this is that we cannot decide *a priori* which point on the occluding contour will prove the most useful. This will depend on the context in which the approach is used. However, there are specific types of points for which our reconstruction method may not work. Therefore, our task will be to select a point on the rim for which we can ensure that our approach is effective. Below we first outline the main ideas, and in Section 5.6 present more details.

5.4.1 The Active Reconstruction Approach

Suppose we have selected a point p on the rim of surface S. For simplicity we will assume that p is at the origin. We first consider the case where p is non-umbilic. Corollary 5.3 says that if p is a hyperbolic point or a non-umbilic elliptic point, there are only two viewing directions in $T_p(S)$ for which k_o obtains a local maximum value and two directions for which k_o obtains a local minimum. Our goal is to find one of these directions since they correspond to e_1 and e_2. We discuss the problem of finding e_2; e_1 is treated similarly.

Both the initial viewing direction and the direction e_2 belong to the tangent plane of the selected point. In order to align the viewing direction with e_2, we execute a tangential viewpoint control step until e_2 is reached (see Figure 37). To do this we must answer two questions: (1) Which direction should the viewpoint move on the tangent plane, and (2) how can we detect when the viewing direction

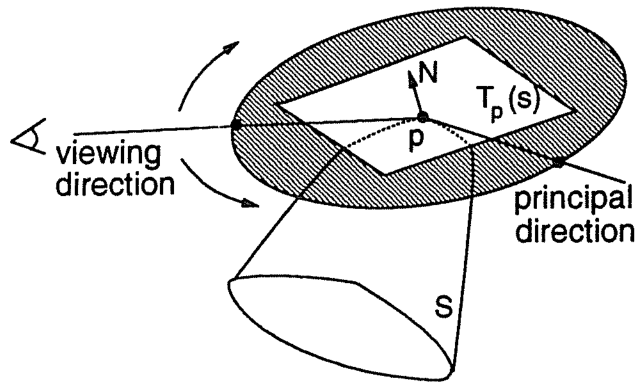


Figure 37: Aligning the viewing direction with a principal direction on $T_p(S)$.

is equal to e_2 ?

We only have two possibilities for moving on the unit circle representing viewpoints on $T_p(S)$, either clockwise or counterclockwise. Obviously, we prefer the minimal motion solution in which the desired extremum is attained with the smallest possible change in viewing direction. In particular, if we move in the direction of increasing k_o, the first extremum we reach is a maximum. It easily follows from the local geometry of elliptic and hyperbolic points that this strategy will in fact produce the smallest viewing direction change (Figure 38). On the other hand, parabolic points do not have this property.

The second question, detecting when the viewing direction is equal to e_2, is partly answered by Corollary 5.3. It says that we can detect this event by detecting a local maximum of k_o. However, in order to detect this local maximum, p must be visible; k_o cannot be measured otherwise. The visibility of p is affected by the local surface geometry at p as well as by the global geometry of S. Ignoring for a moment the case where p is occluded by some distant point on S, we arrive at the following two conclusions: (1) If p is elliptic, we can align the viewing direction with either e_1 or e_2. Furthermore, the maximum possible direction change before the alignment takes place is $\pi/2$ (Figure 38(a)). (2) If p is hyperbolic, we can align the viewing direction only with e_1. The maximum possible direction change in this case is determined by the point's asymptotes (Figure 38(b)).

The problem of recovering the local surface geometry at an umbilic point on the visible rim is even simpler. Corollary 5.1 suggests that if we know p is umbilic, we can recover the local surface shape

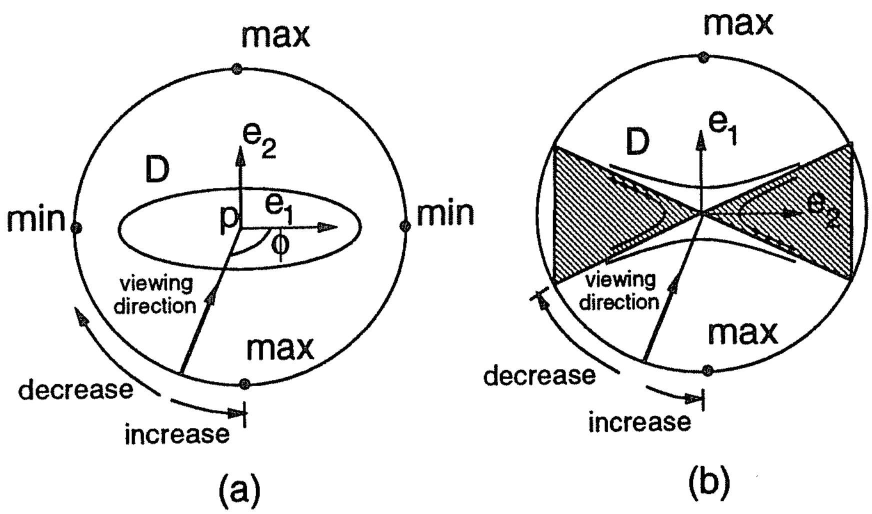


Figure 38: Finding the principal directions. Top views of the tangent plane are shown (refer to Figure 14). The viewing direction makes an angle ϕ with the first principal direction. (a) p is an elliptic point. Clockwise change in viewing direction decreases k_o. The viewing direction can change by at most $\pi/2$ before a local minimum or a local maximum is reached. (b) p is hyperbolic. The only achievable extremum is a local maximum, obtained in this case by a counterclockwise rotation. Shaded areas, delimited by the asymptotes of the point, represent the directions where p is occluded. The maximum viewing direction change before an extremum is found in this case decreases to the angle between e_1 and the asymptotic directions. (Note that the axis labels have been reversed.)

at p by simply measuring the curvature of the occluding contour at p's projection. It therefore suffices to find a way of detecting that p is umbilic. Corollary 5.3 shows that this can be done by determining whether k_o remains constant as the viewing direction moves on $T_p(S)$.

These results suggest a simple algorithm to align the viewing direction with e_2 and recover the local surface shape at p:

Active Local Shape Recovery Strategy

Step 1: Perform a small change of viewing direction on $T_p(S)$ and measure the difference between the previous and current value of k_o. If it increases, continue to change the viewing direction in the same way so that e_2 will be reached first. If it decreases, move the viewing direction in the opposite way. If it remains constant, stop moving; k_{n_1} and k_{n_2} are both equal to k_o (i.e., p is umbilic).

Step 2: Continue moving in the same direction until k_o reaches a maximum. This viewpoint corresponds to e_2 and therefore the viewpoint's motion can be stopped and the current value of k_o used as an estimate of k_{n_1}.

Step 3: Measure the total change of viewing direction between the initial and final directions. Corollary 5.2 says that this angle along with k_{n_1} and the initial value of k_o can be used to determine k_{n_2}.

The above algorithm assumes that we can measure relative changes in viewing direction. For elliptic points this requirement can be relaxed at the expense of additional motion: We can recover the principal curvatures at the selected point by moving to the viewpoints corresponding to the maximum *and* the minimum value of k_o. Hence, in this case shape recovery can be achieved without relying on any quantitative measurements involving the viewpoint's motion; we must simply be able to control viewpoint around the selected point by moving clockwise and counterclockwise on its tangent plane.

5.4.2 Selecting Surface Points for Reconstruction

Any viewpoint motion minimizing or maximizing k_o must take into account the effects of global surface geometry: Irrespective of its local structure, p may become occluded by distant points on S. The following proposition shows that (a) there are at least some points on the visible rim of S that cannot be occluded by S if the viewing direction is changed as described above, and (b) these points are easily detected on the occluding contour (Figure 39).

Proposition 5.1 *(1) Let p be a visible, elliptic point on the rim of a smooth surface S when viewed from direction ξ under orthographic projection. Let q be the projection of p in the image plane and let l be the tangent to the occluding contour at q. Then, p is visible from every direction on $T_p(S)$ iff l does not intersect the occluding contour and is not tangent to it at any point other than q.*

(2) Let C be the occluding contour of S when viewed from direction ξ. Then there is at least one point on S projected in C that is visible from every direction in $T_p(S)$.

See Appendix B for a proof. Figure 40 shows the results of applying Proposition 5.1 to the occluding contour of a candlestick. The proposition implies that the only points ensuring the correctness of the algorithm are elliptic. However, this is a *necessary* requirement for the presence of occlusion but not a

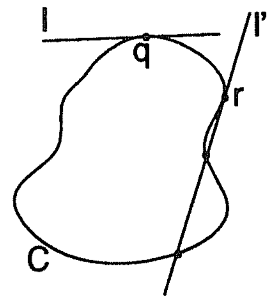


Figure 39: Determining the complete visibility of rim points. Since the tangent at q does not intersect the occluding contour, C, elsewhere, q corresponds to a rim point visible from every direction in its tangent plane.

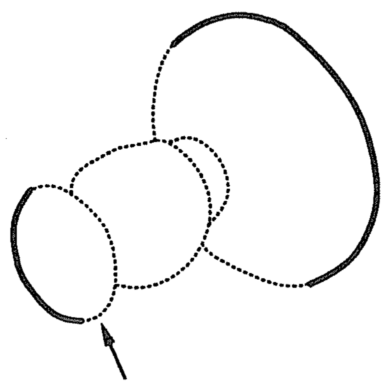


Figure 40: Selecting points for surface recovery. Solid lines on the occluding contour of a candlestick show the points that cannot become occluded while changing viewing direction in their tangent planes. The arrow indicates the point having the greatest (absolute) curvature of all acceptable points.

sufficient one. This means that there are cases where the geometry of hyperbolic points can be recovered with our approach. In fact, shape recovery for hyperbolic points requires less motion on average since the extent of the visibility of these points is limited by their asymptotic directions.

5.5 Surfaces of Revolution

In the last section we presented an algorithm for recovering the shape of a single point on the surface rim. However, there are surfaces for which the local shape of a single rim point reveals global properties of the surface. Surfaces of revolution present an ideal example of such surfaces. The properties and appearance of surfaces of revolution and their generalizations have been studied extensively [34, 38, 77, 116, 131, 137, 153, 168]. Here, we focus on the specific relation between their global structure and the local shape of points on the rim.

Surfaces of revolution are formed by rotating a planar curve around a straight axis that does not meet the curve. Therefore, we can completely describe a surface of revolution by the axis and the generating curve. Approaches for recovering the axis of a surface of revolution have been mainly geared towards detecting symmetries in their occluding contour or outline [116], or utilizing their viewpoint-invariant properties [77, 131, 132]. The problem with detecting symmetries in the occluding contour is that the existence of such symmetries depends on viewpoint. On the other hand, the identification, detection and utilization of viewpoint-invariant properties is a non-trivial task. For example, in [132] the axis was recovered using a Hough transform-based technique. However, such a technique largely depends on the number of rim points actually detected. In addition, the axis is severely foreshortened for near-"top" views of a surface of revolution (i.e., when the viewing direction is almost parallel to the axis of rotation), limiting the applicability of methods relying on a single, arbitrary view to precisely recover the axis. Our active approach neatly lends itself to these problems in order to make them easier to handle. The idea is that if the viewing direction can be aligned with a principal direction of a rim point, then shape and symmetry analysis of the occluding contour becomes especially simple. This is because one of the principal directions corresponds to a "side" view of the surface (i.e., a view for which the viewing direction is perpendicular to the axis of rotation). If the generating curve of the surface can be written in the form $y = f(x)$, then the recovery of the axis of rotation allows us to recover the generating curve directly from a side view. Even further, the surface rim from such a view is guaranteed to be completely visible. In the rest of this section we focus on surfaces of revolution whose generating curve has this property.

Consider a point p on the rim of a surface of revolution when viewed from an arbitrary direction

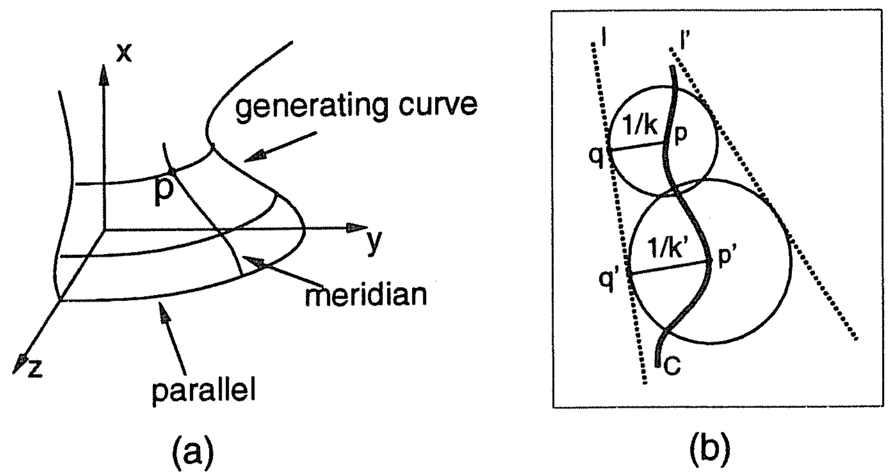


Figure 41: (a) A surface of revolution. The x-axis is the axis of rotation. (b) Constraining the axis of rotation from the principal curvatures of two rim points from a side view. Curve C is a segment of the occluding contour corresponding to a side view. Since the view is a side view, C belongs to a generating curve of the surface. Let the principal curvature corresponding to the parallels at p and p' be k and k', respectively. The distance of p and p' from the projection of the axis of revolution, l, is $1/k$ and $1/k'$, respectively. Let q and q' be the projections of the points of intersection of the axis with the planes of the parallels at p and p', respectively. The axis must be perpendicular to the lines along pq and $p'q'$. Therefore the axis must be tangent to the circles of radius $1/k$ and $1/k'$, centered at p and p' respectively. There are at most two such lines that do not intersect the generating curve C. The direction of the axis is the normal of the plane defined by the viewing direction and the line through pq.

(Figure 41(a)). The two principal directions at p correspond to the tangents to the parallel and the meridian passing through p. Since the parallel is a planar curve, if the visual ray is tangent to the parallel at p it is contained in the plane of the parallel. Hence, it is perpendicular to the axis of rotation and the view corresponds to a side view of the surface. The occluding contour from such a side view is symmetric. Therefore, the axis of rotation (as well as the generating curve) can be recovered by simply using existing symmetry-seeking approaches (e.g., [77]) which are well-defined for such a viewpoint. However, the direction and position of the axis can also be constrained by recovering the principal curvatures corresponding to the parallels for two points on the rim of a side view (Figure 41(b)). This approach is similar to one used by Richetin *et al.* [137] where the geometry of the occluding contour at two parabolic points was used to hypothesize the pose for surfaces that are straight homogeneous generalized cylinders.

We must choose between moving towards the principal direction of minimum curvature or moving towards the one of maximum curvature. Although the curvature extremum corresponding to a side view for a selected point is not known *a priori*, this choice is easy if the visible rim contains hyperbolic points. Recall that the only principal direction from which these points are visible is the direction of maximum curvature and that if the generating curve of the surface can be written in the form $y = f(x)$, then these points must be visible from a side view of the surface. Hence, we can select one hyperbolic point and align the viewing direction with the principal direction of maximum curvature (Corollary 5.3).

It is also easy to show a more general property of surfaces of revolution with this type of generating curve: If the viewing direction smoothly changes on the tangent plane of a selected rim point, this point will not become occluded if the viewing direction is approaching the direction of a side view. This fact can be used to decide how to change viewing directions on the tangent plane in order to approach a side view of the surface when no hyperbolic surface points are visible.

Our discussion above deals with a specific type of surface of revolution. However, it can be generalized to an arbitrary surface of revolution and to the case of straight homogeneous generalized cylinders where the axis is perpendicular to the cross-section. Consider the case where a rim point is selected that belongs to a parallel that is also a geodesic. If such parallels exist on the surface, our approach can be used to obtain both the top and the side views of the surface as well as its axis. Consider the case where the generating curve of the surface of revolution cannot be written in the form $y = f(x)$ (e.g., a torus). In this case, we can still recover the axis of rotation from the side view using occluding contour symmetries, and find points on the generating curve for which the tangent to the generating curve is parallel to the axis. These points belong to parallels that are geodesics and their principal directions correspond to the side and top views of the surface. Therefore, we can align the viewing direction with the top view for any type of surface of revolution.

Points on the rim that belong to geodesic parallels are also important because they can be used to recover the axis of surfaces of revolution and straight homogeneous generalized cylinders. The surface normal at these points lies on the plane of the parallels. The viewing direction corresponding to a side view also belongs to this plane. Therefore, we can recover the plane of the parallels. Since the axis of rotation is normal to this plane we can also recover the direction of the axis of the surface. It is in fact possible to detect such a point on the rim if it exists (without having already determined the axis).

The next section extends our basic shape recovery step by (1) selecting a new point on the surface in the vicinity of the previously selected point, and (2) applying the shape recovery step presented in Section 5.4 to the new point. We also briefly discuss how this two-step approach can be used to select rim points that belong to geodesic parallels.

5.6 Extending Surface Recovery to Neighboring Points

Our main objective is to recover the complete shape description for a single rim point. In this section we consider an extension to this approach—selecting a new point and applying the shape recovery process to that point. We must consider two important issues in order to demonstrate the effectiveness of such an extension:

- The extent of the viewing direction adjustments needed to align the viewing direction with one of the principal directions at the newly selected point.
- The extent of the viewing direction adjustments *required* by our basic shape recovery algorithm in order to produce reliable shape information for the newly selected point. This is because if the viewing direction adjustment is close to zero, then numerical problems are introduced in the calculations of the principal curvatures from Corollary 5.2.

We will discuss the issue of selecting new points for shape recovery based on these two issues. The process has as a primary goal the removal of the first point from the rim and its replacement by a new point at which the first step will again be applied.

Let p be the previously selected point. After applying the shape recovery step, the viewing direction ξ is aligned with one of the principal directions at p, say e_2. We have seen that if we change directions in $T_p(S)$, p will not leave the rim. Therefore, we must change viewing direction in some other plane containing e_2. The important issues here are (1) which plane should be selected for changing the viewing direction, and (2) how much should the viewing direction change in that plane? The motivation for our approach is to ensure that the shape recovery step for the new point will need only small viewing direction adjustments. In other words, we require that the new viewing direction does not form a large angle with one of the principal directions at the new point.

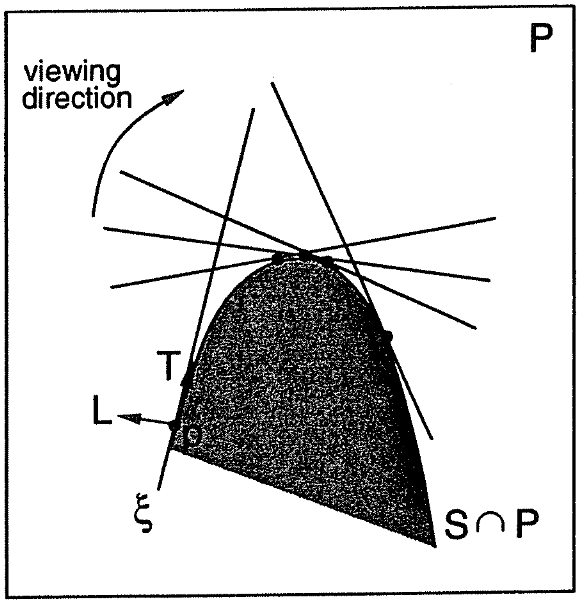


Figure 42: Removing p from the rim. The figure shows the intersection of a selected plane P with the surface. The viewing direction ξ changes in P and the visual rays graze the surface along the curve $\beta(s) = S \cap P$.

Suppose we have selected a particular plane P passing through p and containing e_2, and that we continuously change viewing direction in that plane. As the viewing direction changes, the visual ray contained in P will graze the surface along a curve $\beta(s)$ also contained in that plane (Figure 42). Now suppose that we stop at a new viewing direction ξ'. The visual ray will now be tangent to $\beta(s)$ at some new surface point. The shape recovery step will now be applied to this point, attempting to align the viewing direction with e_2 at the point. We must therefore examine how the angle of $\beta'(s)$ with e_2 varies with s. The basic idea is to examine the properties of $\beta(s)$ in light of the following efficiency and reliability requirements:

- The efficiency requirement is that $\beta(s)$ should always form an angle with e_2 that is as close to 0 as possible. This means that we require $\beta(s)$ to approximate a line of curvature.

- The reliability requirement is that $\beta(s)$ should form an angle of at least ϕ^*, for some predetermined constant ϕ^* that depends on the reliability of the shape recovery step. This means that we require $\beta(s)$ to form an angle of at least ϕ^* with the lines of curvature corresponding to e_2.

The compromise between these two requirements is to require $\beta(s)$ to form an angle of exactly ϕ^* with the corresponding lines of curvature. This means that $\beta(s)$ is a *loxodrome* for the surface, i.e., a line on S that forms a constant angle with the lines of curvature. Therefore we should trace S along such a curve while changing viewing directions.

We show in Appendix C that if the selected plane P is the normal plane (i.e., the plane defined by the viewing direction and the surface normal at p) and if the viewing direction change on this plane is small, then the viewing direction adjustments during the shape recovery step will in fact be smooth and depend entirely on intrinsic properties of the surface. Specifically, we show that these adjustments are (to a first approximation) proportional to the geodesic curvature of the lines of curvature at p and inversely proportional to the normal curvature of the lines of curvature at p. This is an important result because it allows us to predict the performance of our active viewing strategy based on knowledge of the intrinsic properties of the surface.

As an example, consider the case of surfaces of revolution. Suppose that the viewing direction is aligned with the principal direction corresponding to the parallels. Now suppose that the viewing direction is changed on the normal plane at p and eventually a new point p' is selected for shape recovery, as outlined above. If p belongs to a geodesic parallel, no viewing direction adjustments will be necessary during the shape recovery step at p', i.e., the viewing direction is also tangent to the parallel through p'. On the other hand, if the geodesic curvature of the parallel through p is non-zero, some viewing direction adjustments will be necessary. In fact it can be shown that if this process is repeated and points $p', p'', p''', \ldots,$ are selected these points will asymptotically approach a geodesic parallel if such a parallel exists. To illustrate this, let us assume for simplicity that the axis of the surface of revolution is vertical and the point initially selected is p. Then, the new point selected will be on a parallel below the parallel through p if the surface normal is pointing upwards. Therefore if there is a geodesic parallel below the parallel through p, the points selected will approach that geodesic parallel.

5.7 Experimental Results

In this section we demonstrate the applicability of our active shape recovery approach. We have implemented a prototype system that (1) automatically selects points on the rim of an object, (2) tracks

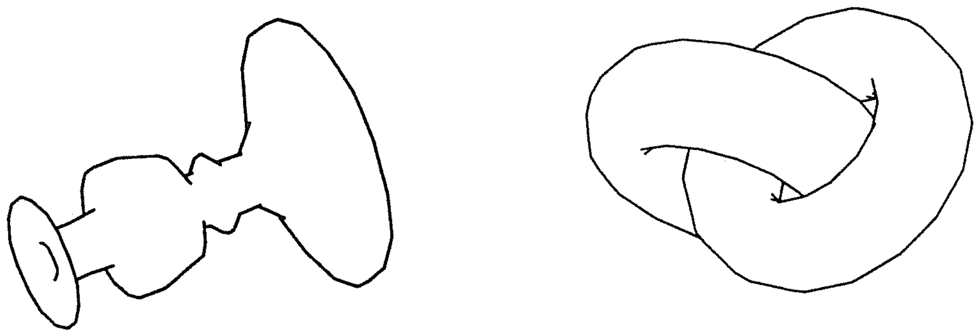


Figure 43: Models of a candlestick and two tori used for the simulations.

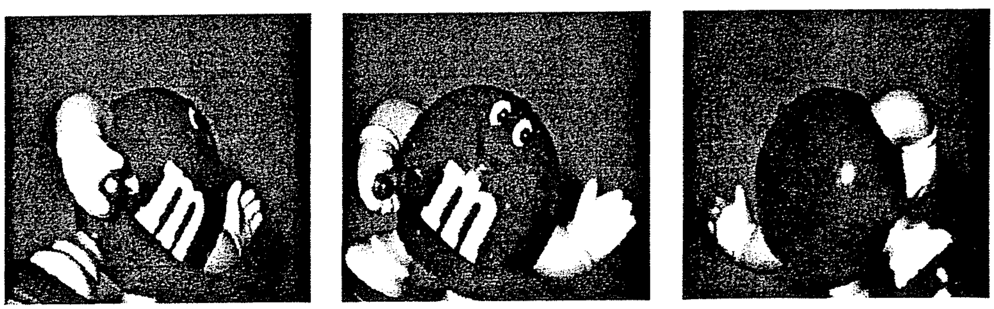


Figure 44: A sequence of 120 frames used in our experiments. Frames 1, 30 and 120 are shown.

these points while changing viewing direction on their tangent plane, and (3) computes the curvature of the occluding contour at the selected points in order to detect the viewpoints where it obtains an extremum value. We have applied our algorithms to simulated scenes and have also performed some preliminary experiments with a real scene. Figures 43 and 44 show the objects used in our experiments.

5.7.1 Simulated Scenes

The simulations were performed by synthetically generating images given a polyhedral object model[3] and a camera constrained to move only in a circle of viewing directions around the object. The occluding contour of the model for the current viewing direction was displayed and updated as the viewing direction changed smoothly. In our examples, the viewing direction was changed on a plane Π defined by a horizontal line in the image and the viewing direction (i.e., the projection of this plane to the image is a horizontal line).

Recall that the process of aligning the viewing direction with a principal direction at a point requires that the viewing direction changes on the point's tangent plane (Figure 37). Hence, the one degree of freedom in rotation allowed us to detect the principal directions only for points tangent to Π. Our system automatically identified these points by finding the points where the occluding contour was tangent to a horizontal line. These points were automatically detected, labeled and subsequently tracked while the viewing direction changed smoothly (Figures 45, 46).

Point tracking was performed using the Tangent Point Tracking Procedure of Section 3.5. Figures 45 and 46 show some of the tracked points for the two models. The points were initially selected and labeled for the viewing direction $\xi = 0$. Note that after a rotation of 3.93 radians the only unoccluded points are the points 0 and 6, exactly as predicted by Proposition 5.1 (i.e., the tangents to the occluding contour at these points do not intersect the contour).

Curvature computations were performed by first approximating the occluding contour in the neighborhood of the selected points using cubic B-splines [46]. The curvature was measured at the points where the tangent to the splines was horizontal. Even though splines have the effect of smoothing high curvature parts of a curve, we found that even with the actual rim curvatures being underestimated the curvature maxima were very distinct. In the case of polyhedral models, smooth viewpoint changes can result in an arbitrary number of model vertices entering and exiting the polyhedral rim. Hence, the shape of the rim changes in a very discontinuous fashion, a problem not encountered with curved objects where topological changes of the rim are not as frequent. This fact resulted in discontinuities in the curvature estimates, which ideally should vary continuously with viewpoint. However, Figures 47

[3] Our use of polyhedral models was only for convenience in generating the occluding contour. The implementation of our algorithms did not exploit the polyhedral property of the models.

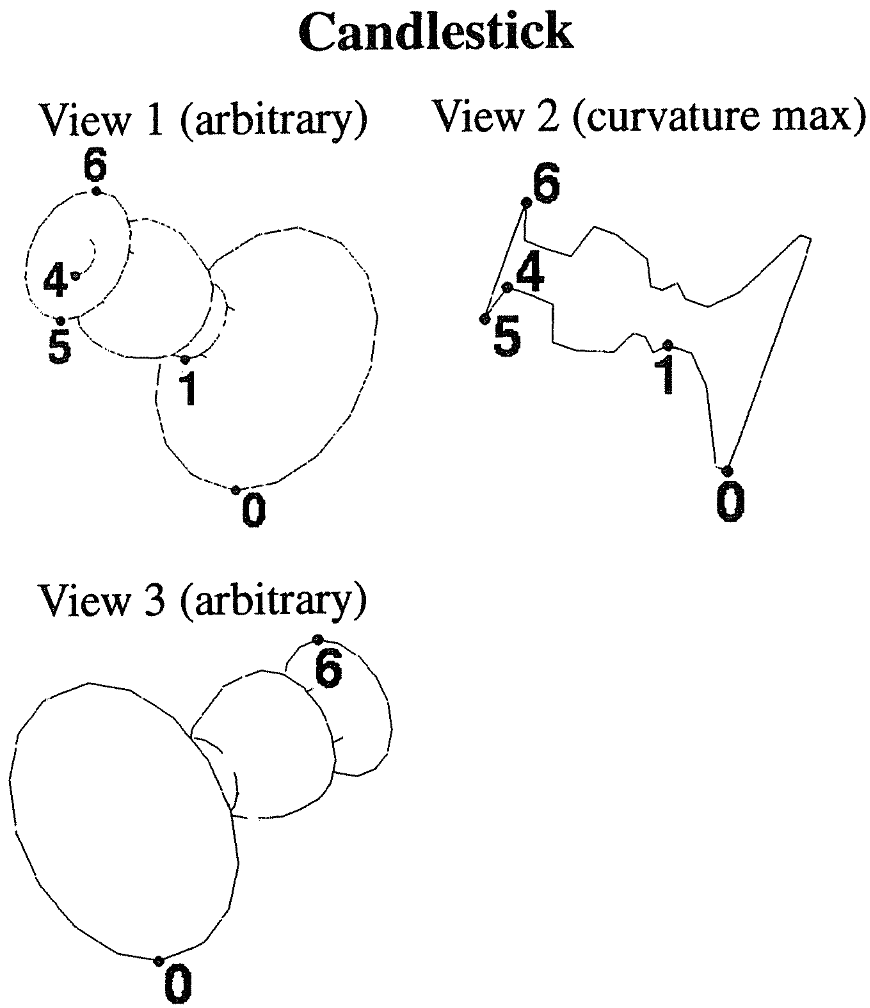


Figure 45: Snapshots of the occluding contour as the viewing direction changes. The numbered points are the points being tracked.

and 48 show that the major peaks and valleys of the curvature estimates are clearly visible even in the presence of the discontinuities caused by the polyhedral approximation.

Figures 47 and 48 also show how the absolute value of the curvature of the occluding contour at the selected points varies with viewpoint. Note that the candlestick and the torus are surfaces of revolution. Therefore, a "side" view corresponds to the viewing direction that is a principal direction for all points on their rim (i.e., the direction is tangent to the surface parallels). This is illustrated by the fact that

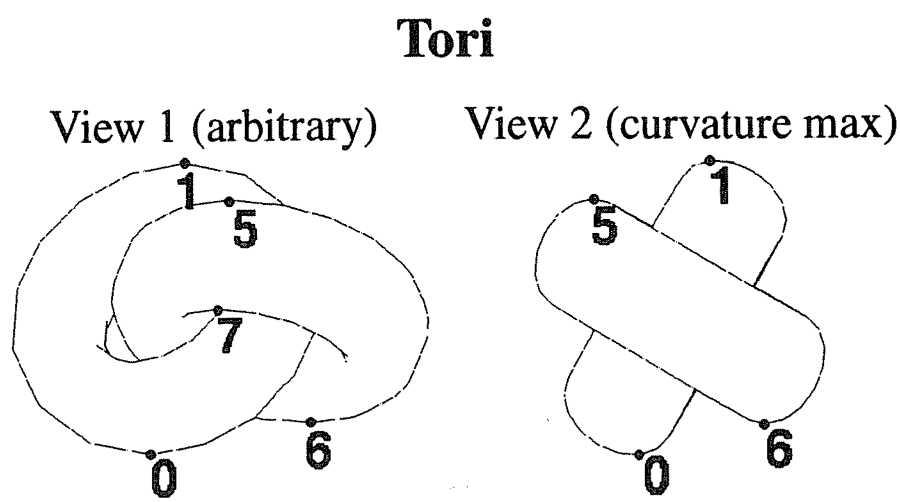


Figure 46: Snapshots of the occluding contour as the viewing direction changes. The numbered points are the points being tracked.

the curvature maxima and minima occur at approximately the same viewing directions for the selected points. View 2 of the candlestick and the tori shows the occluding contour from the viewpoint of maximum curvature for point 0. The views in fact correspond to side views of the surfaces as expected. Also, note that in the case of the candlestick, the curvature maxima are much larger than the minima (over an order of magnitude), whereas in the case of the tori the extremal values are not very different. This is because the difference in the values of the principal curvatures at the selected points on the candlestick is much larger than for the two tori.

5.7.2 A Real Scene

In order to perform preliminary experiments with a real scene we extended the simple tracking and curvature estimation algorithms used in our simulations, and applied them to the sequence shown in Figure 44. The sequence was produced by manually rotating an object after placing it on a horizontal turntable. The amount of the object's rotation between frames was assumed unknown. As in our simulations, this object motion allows the viewing direction to be aligned only with the principal

Candlestick

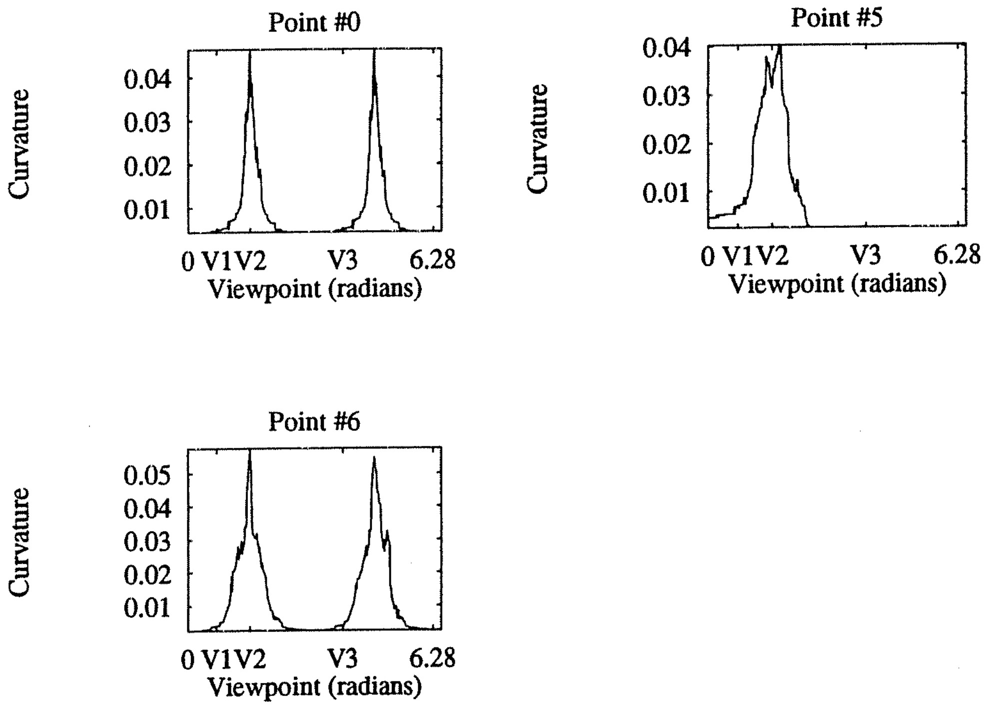


Figure 47: Variation of the absolute curvature with respect to viewpoint at the selected points on the occluding contour. The model was rotated a total of 2π radians. The curve for point 5 on the candlestick ends at the viewpoint where the point's occlusion is detected. Viewpoints $V1, V2, V3$ correspond to views $1, 2$ and 3 of the candlestick in Figure 45

directions of points on the object whose tangent planes are horizontal.

The occluding contour of the object was tracked across frames using a simplified implementation of a B-spline snake [44, 84]. The snake was interactively initialized near the object's contour. Point tracking was again performed by tracking the point on the snake whose tangent is horizontal (Figure 49). Figure 50(a) shows the variation of the curvature of the snake at the tracked point for one run of the tracking process. Figure 51 shows the views of the object corresponding to the minimum and maximum measured curvature.

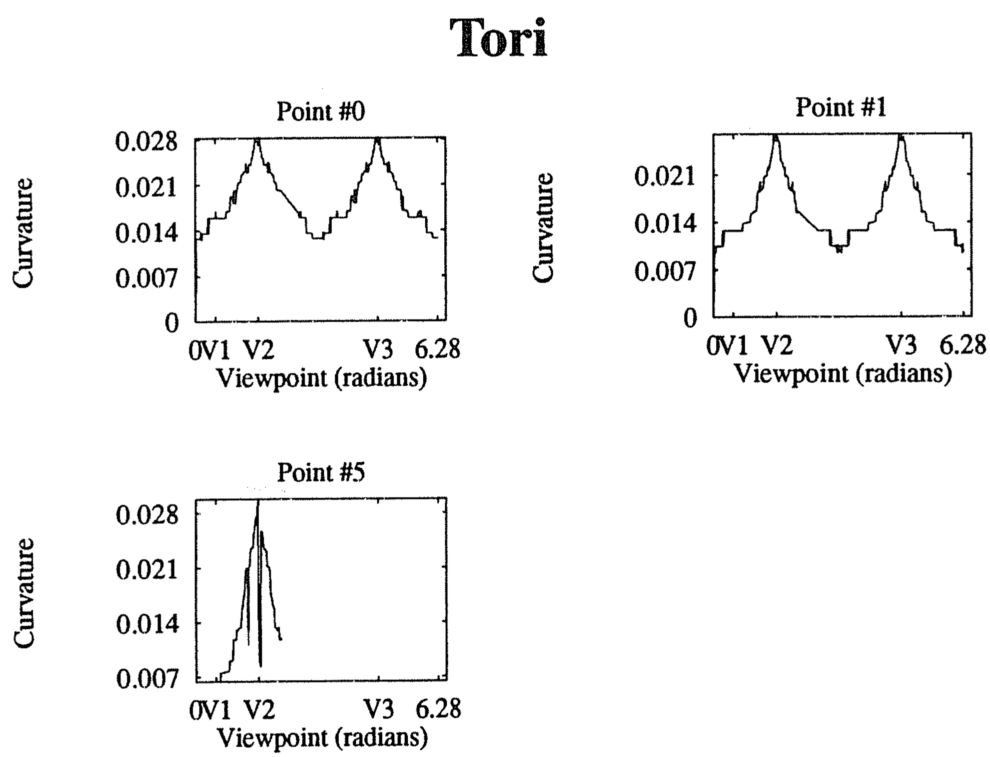


Figure 48: Variation of the absolute curvature with respect to viewpoint at the selected points on the occluding contour. The model was rotated a total of 2π radians. The curve for point 5 on the tori ends at the viewpoint where the point's occlusion is detected. Viewpoints $V1$ and $V2$ correspond to views 1 and 2 of the tori in Figure 46.

Curvature measurements were noisy mainly because of the snake's tracking behavior, which depended on the initial positioning of the snake and did not always lead to accurate approximations of the object's contour. Clearly, the curvature estimation process can be improved (especially near viewpoints corresponding to curvature maxima) by paying closer attention to the snake's tracking behavior. Our purpose here is simply to illustrate that the theoretically-predicted curvature variation at the tracked point can be observed in practice. Since the only computations apart from snake tracking involve measuring the curvature of the snake at a single point, our active reconstruction approach is amenable to a real-time implementation; snake trackers operating at video rates are already becoming available [28].

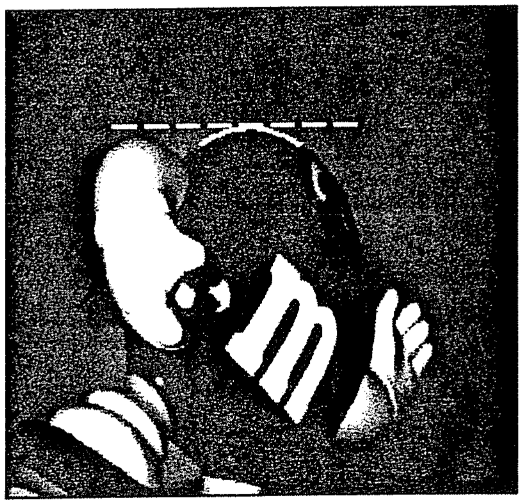
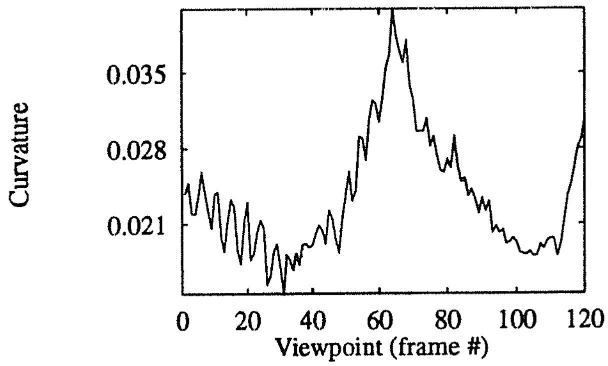


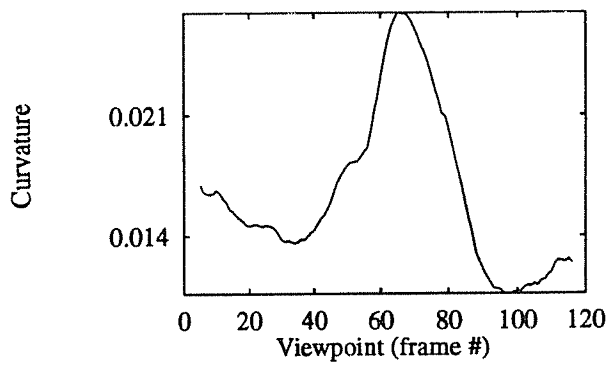
Figure 49: The point being tracked is the snake point whose tangent is horizontal.

Figure 50(a) shows that it is necessary to incorporate measurements from multiple adjacent views in order to detect the principal directions of the tracked point. This implies that viewpoint must move past the viewpoint corresponding to the point's principal direction before the curvature extremum can be reliably detected. Figures 50(b) and 52 give the results from another run of the tracking and curvature measurement process. In this run, the contour's curvature was averaged over nine frames. Although the snake approximated the object's contour in a different manner, the principal directions of the tracked point are again easily distinguishable.

Further improvements can be made to the principal curvature measurements at the selected point by incorporating information from multiple frames during the viewpoint's motion, and for more accurately localizing the point's principal directions. More specifically, when the relative changes in viewing direction between frames can be accurately measured, the principal curvatures and principal directions at the selected point can be predicted using three viewing directions on the tangent plane of the tracked point that satisfy the reliability requirement of Section 5.6 (i.e., they are not too close to each other) [96]. This observation leads to a prediction-verification scheme for improving the accuracy of our active approach, whereby predictions during the process of aligning with a principal direction are evaluated



(a)



(b)

Figure 50: (a) Curvature variation with viewpoint. (b) Curvature estimates averaged over nine frames.

against the outputs of the contour curvature estimator and the extremum detector. We expect this process to be useful primarily when the viewing direction is close to the principal direction of maximum absolute curvature where contour curvature measurements tend to be more reliable.

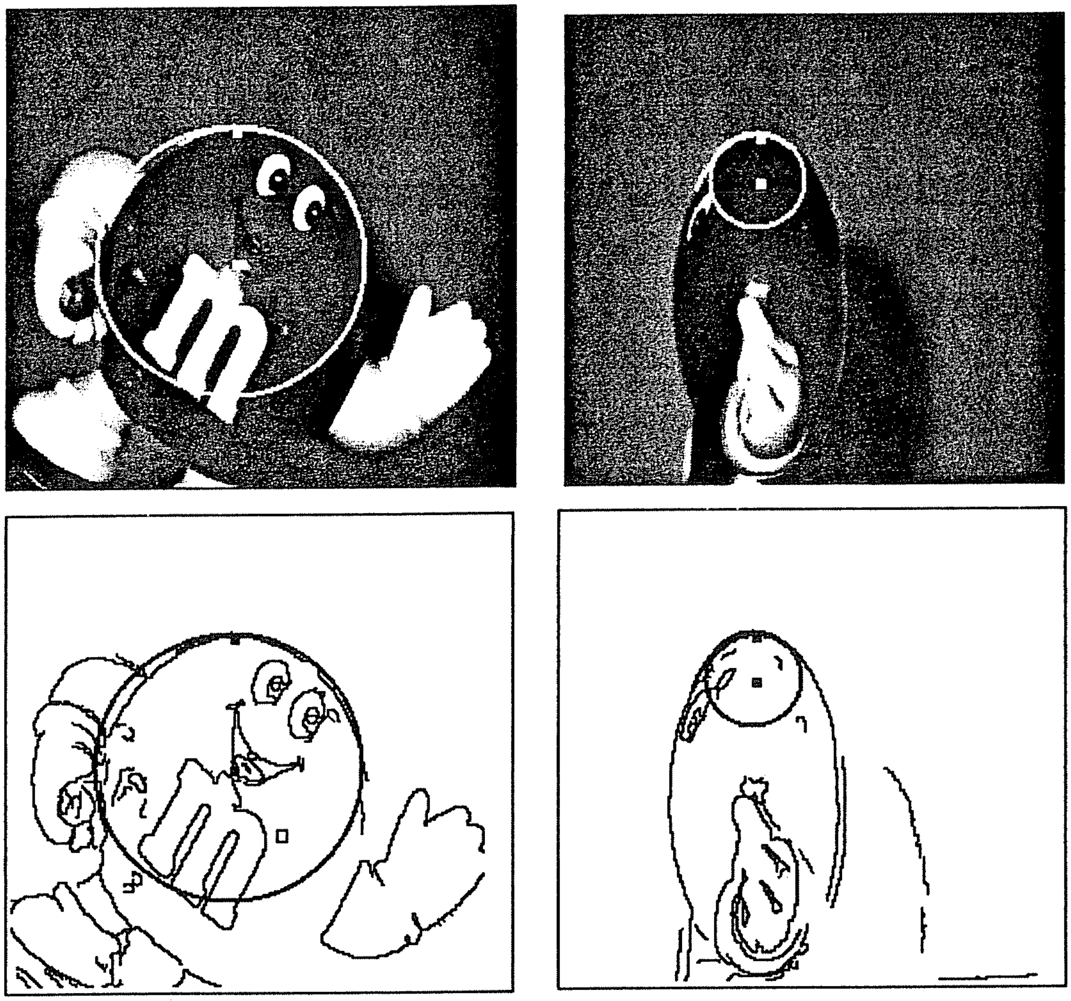


Figure 51: Viewpoints corresponding to the global minima and maxima of the curvature measurements. Also shown is the computed osculating circle at the tracked point, i.e., the circle that is tangent to the tracked point and has radius equal to $1/k_o$.

5.8 Summary

This chapter demonstrates that an active vision system can use a very simple exploration strategy to recover quantitative shape information at selected visible rim points. This strategy is based purely on the computation of a simple property of the occluding contour (curvature at a point). Our experimental results show that this strategy is readily implementable and because of its simplicity and its low

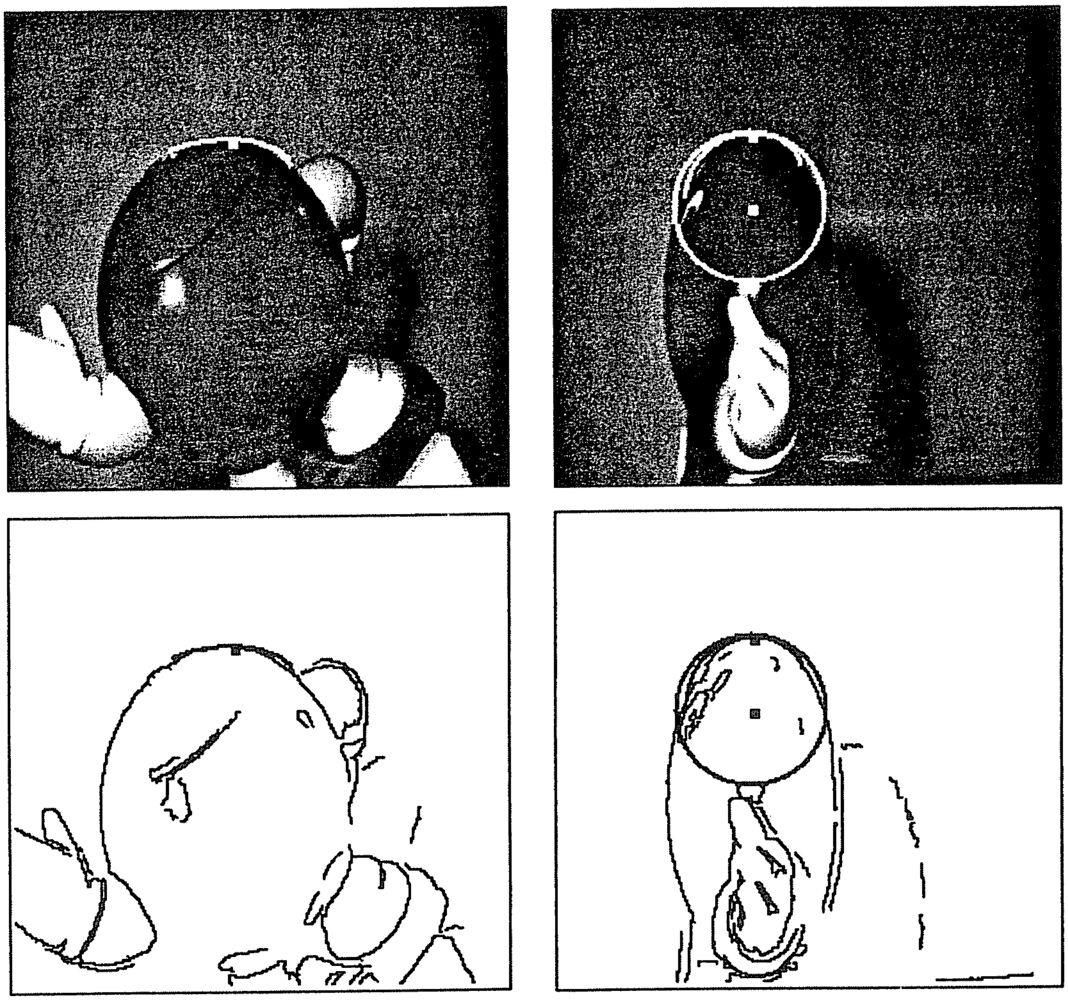


Figure 52: Viewpoints corresponding to the global curvature minima and maxima for a different run of the tracking process. The extrema were found after averaging the curvature measurements over nine consecutive frames. The osculating circle for the view corresponding to the detected curvature minimum is not shown because it is not fully contained in the image. Note that both curvature measurements have been underestimated, and the global curvature minimum now corresponds to the back side of the object.

computational requirements, is very suitable for real-time implementation.

The use of continuous, contour-driven viewpoint control is the most crucial aspect of the method. The ability to change viewpoint in a controlled manner makes it possible to reach the special viewpoint where the shape of the occluding contour provides complete and exact curvature information. Moreover, our approach demonstrates that recovering quantitative shape information from the occluding contour does not necessarily require knowledge of the velocities or accelerations of the viewpoint's motion, but only knowledge of the viewing direction. The reason is that viewpoint motion is not used to merely change the shape of the occluding contour (as in existing approaches), but it is used to change it in a well-defined way, factoring out the need for differential measurements involving viewpoint motion.

Current limitations of the approach are (1) the use of orthographic projection, (2) the ability to recover surface shape only at isolated surface points, and (3) its applicability to only elliptic or hyperbolic surface points. Extension of the method to perspective projection is possible. In particular, a formula analogous to Blaschke's formula governs the relation between the contour's curvature and the local surface shape [90]. If the distance to the selected visible rim point is kept constant during tangential viewpoint control (e.g., by exploiting the *kinetic depth effect* [16]), shape recovery can be performed in a manner identical to the one described in this chapter.

We believe that our active approach of moving towards viewpoints that are closely related to the geometry of the viewed surfaces is a very important and general one. Consider, for example, the problem of obtaining a "face-on" view of a planar curve (or a texture element). This problem has been studied extensively in the past and several approaches exist that *hypothesize* face-on views, based on information from a single viewpoint (e.g., [35, 83]). Exploration strategies similar to the one presented in this paper enable viewpoint to be changed in order to obtain a face-on view of a planar curve. Special viewpoints are also particularly appropriate in the context of recognition, since impose additional constraints on the projection of the objects to be recognized [183]. For example, we can force reduction in the dimensionality of the recognition problem for curved objects [96] by first moving to a view along an asymptotic direction at a parabolic visible rim point (only a one-dimensional set of such views is generically possible).

The problem of overcoming the second limitation still remains an open issue; the local nature of the strategy makes it applicable to the construction of only sparse surface representations. The next chapter shows that we can construct dense global surface representations by appropriately controlling viewpoint and by sacrificing the weak motion assumptions employed in this chapter (e.g., unknown viewpoint velocities and accelerations).

Chapter 6

Global Surface Reconstruction

In the previous chapters we focused on the application of the continuous, contour-driven viewpoint control framework to local shape recovery tasks. In this chapter we go a step further, asking what viewpoint control strategies become important for tasks requiring global shape recovery. Specifically, we design a continuous, contour-driven viewpoint control strategy for deriving a global and dense three-dimensional description of an object's surface from its occluding contour. The global surface reconstruction task we consider can be formulated as follows: "How should viewpoint be controlled to generate a dense sequence of images that allows reconstruction of an object's entire surface, or as much of the surface as possible?"

While local shape recovery strategies can be used to achieve global reconstruction from the occluding contour for special classes of surfaces such as surfaces of revolution (Chapter 5), the strategies must necessarily rely on class-specific assumptions that link the local shape of surfaces in the class to their global structure. Here we consider the general case for a generic object of arbitrary shape; the object is unknown, can be non-convex, and can self-occlude. A solution to this problem requires taking into account how an object's global geometry (e.g., the configuration of convex, concave, and hyperbolic regions on an object, and the manner in which they come in and out of view) affects the outcome of the exploration process. In particular, it requires making explicit how the interaction between continuous viewpoint control and an object's global shape affects the motion of the visible rim over the surface and its topological changes.

Our analysis builds on concepts already introduced in earlier chapters: The visible rim's and the occluding contour's viewpoint-dependent shapes are put in the context of global surface reconstruction and their relation to the global geometry of smooth surfaces is investigated. The central notion in the chapter is the notion of the *exploration frontier*. At any point in time during the exploration process, the exploration frontier is defined to be the boundary of the already-reconstructed regions on the object being explored. The purpose of the strategy we develop is to force the visible rim and, consequently, the exploration frontier, to slide over as much of the surface as possible.

One of the key theoretical results we obtain in this chapter is that continuous, contour-driven viewpoint control allows this appearance structuring process to be studied *locally* in both space and time. This allows us to achieve global reconstruction by focusing on those special viewpoints that cause changes in the topology of the visible rim, and those special regions on the surface where the rim's topological changes occur. The practical consequence is that global reconstruction can be achieved simply through repeated applications of an appropriate tangential viewpoint control motion.

The rest of the chapter is organized as follows. Section 6.1 motivates the main ideas of the approach, puts it in the context of previous work on global surface reconstruction, and highlights some of the major difficulties in solving the problem. Section 6.2 provides necessary geometrical background. Section 6.3 considers the problem of locally structuring the motion of the exploration frontier. This leads to a characterization of the reconstructible regions on an object's surface (Section 6.4). Section 6.5 then shows how the basic motions developed in Section 6.3 can be used to incrementally "grow" the reconstructed regions on an object's surface. The main result of the chapter is presented in Section 6.6.3, which presents a strategy for global reconstruction of arbitrary generic objects that uses the basic motions of Section 6.3. Simulation results of applying this strategy to a curved object are presented in Section 6.7. The chapter concludes with Section 6.8 which summarizes the main contributions of the chapter, its main limitations, and briefly discusses possible extensions.

6.1 Active Global Surface Reconstruction

Under continuous viewpoint control the visible rim "slides" over the surface and may change its connectivity, affecting the geometry and topology of the occluding contour [89–91]. When the viewpoint's velocities and accelerations are available, the contour's deformation contains all the information necessary to construct a dense surface representation for the points over which the visible rim slides [44, 60, 62, 63, 82, 146, 170, 191]. We exploit this property by formulating global surface reconstruction as the task of continuously controlling viewpoint so that the visible rim slides over the maximal, connected, and reconstructible object regions intersecting the visible rim at the initial viewpoint. This formulation allows us to separate the issue of controlling viewpoint from the issue of reconstructing the surface itself, i.e., processing the images produced during the viewpoint's motion.

Previous work on constructing global surface models of curved objects with few or no markings and discontinuities used primarily range data [45, 76, 117, 152, 166, 177]. Apart from requiring the availability of laser range-finders to recover an object's shape, these approaches do not adequately address the question of how the global geometry of an object's surface (e.g., self-occlusions) affects the completeness of the constructed three-dimensional models. Recently, the problem of constructing global models of curved objects from images of their silhouette or their occluding contour has been receiving a growing amount of attention [1, 40, 63, 82, 146, 152, 157, 191, 192]. All approaches on this subject deal with the problem of deriving an accurate surface representation from an image sequence generated through object-independent viewpoint control; no previous work has considered the problem of automatically generating a dense sequence of images that would guarantee the global reconstruction of an object with complex surface geometry from its occluding contour. The viewpoint control strategy described in this chapter complements previous approaches on recovering surface shape from the occluding contour by giving a way to generate such a sequence.

The main contribution of the work in this chapter is to make precise how viewpoint should change with respect to an unknown object to achieve global reconstruction of an object's surface. The strategy we develop can be used for constructing CAD models of objects whose geometry and appearance are not known beforehand. This is achieved by studying the interaction between the viewpoint controls, the global shape of the object, and its dynamic appearance. Unlike previous approaches where global reconstruction is not guaranteed, we show how continuous, contour-driven viewpoint control leads to a strategy which, even though it depends on local and efficiently-computable information, guarantees global reconstruction of unknown, arbitrarily-shaped generic objects.

Our strategy for controlling viewpoint is developed in the context of three increasingly more general reconstruction tasks: *The region reconstruction task*, where viewpoint is controlled to reconstruct a region around the visible rim; *the incremental surface reconstruction task*, where viewpoint is controlled to "push" the exploration frontier over the unreconstructed regions on the object; and the *global surface reconstruction task*, where all maximal reconstructible regions intersecting the visible rim at the initial viewpoint are reconstructed.

The main difficulty in solving these three reconstruction tasks is that although we have some control over the motion of the visible rim over the surface, this control is not complete; the motion of the visible rim also depends on the shape of the surface itself. In addition, the visible rim's topology can change, further complicating the reconstruction process. Consider, for example, the pipe-shaped object shown in Figure 53. In order to reconstruct its surface, the viewpoint's motion must force the visible rim to slide over all points along the dark curve drawn on the surface. When viewpoint (b) is the initial viewpoint, one way to proceed is to move upward, causing the visible rim to slide over the segment of the dark curve that is initially occluded. As viewpoint moves upward, however, the visible rim in the vicinity of the dark curve shrinks to a point and disappears due to self-occlusion (Figure 53(e)), changing the topology of the visible rim and making any further upward motion ineffective. Viewpoint must now move in some other way in order to continue the reconstruction process. Similar difficulties occur due to geometrical changes in the visible rim (i.e., even when no topological changes occur).

This simple example illustrates that a number of different motions may be necessary to achieve global surface reconstruction. It is therefore necessary to ask what motions are needed, whether the whole surface is always reconstructed, and whether the reconstruction process is guaranteed to terminate. These questions are precisely the reasons why provably-correct strategies for controlling viewpoint become important: Since the answers to these questions are not evident even for surfaces as simple as the pipe-shaped object of Figure 53, such strategies become useful if one hopes to use them for reconstructing the surface of complex, curved, real-world objects.

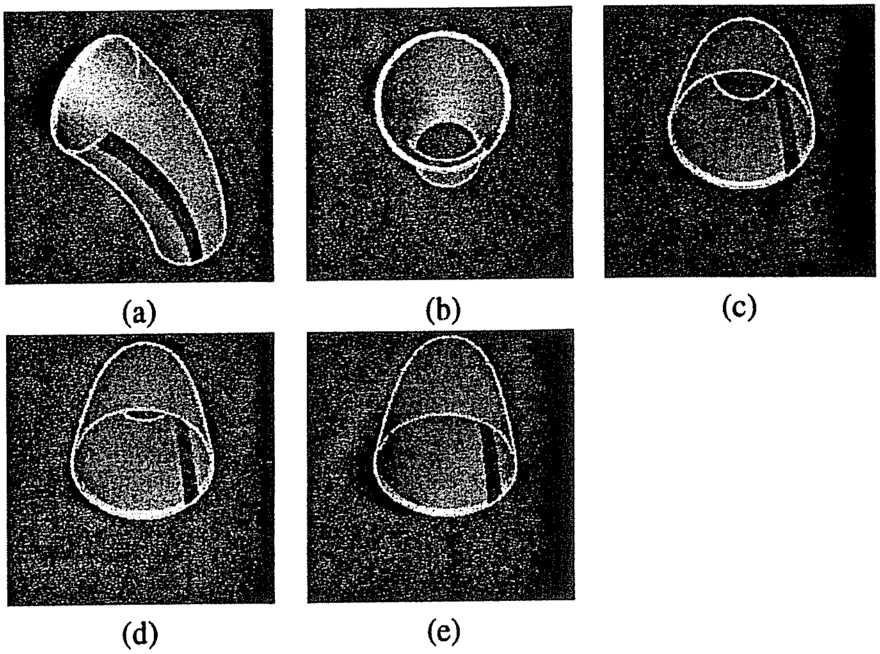


Figure 53: Difficulties in reconstructing the surface of a pipe-shaped object. (a) The dark curve is also drawn on the interior surface of the object, as shown in (b)-(e). (b) Initial view of the object. (c)-(e) During upward viewpoint motion the visible rim curve sliding in the vicinity of the dark curve shrinks to a point and disappears.

6.2 Occluding Contour Evolution Under Continuous Viewpoint Control

We begin by specifying the class of objects considered in this chapter. We assume the object being explored is bounded by a smooth, closed, and *generic* surface S. Informally, generic surfaces exemplify the notion of non-degeneracy: They are surfaces whose geometrical characteristics (e.g., the Gauss map) are not affected if the surface is infinitesimally perturbed[1]. Generic surfaces can approximate

[1]Generic surfaces have been the subject of research in singularity theory [85, 129, 130, 172]; their precise definition is quite technical (e.g., see [85]); the generic condition for a family of smooth maps can be described as an application of a finite number of transversality conditions [172]. Intuition about the genericity assumption can be obtained by considering simpler cases, such as surfaces that are graphs. In particular, a function $f : \Re^2 \to \Re^+$ is generic if (1) f has only non-degenerate critical points (i.e., the determinant of the Hessian is not zero), and (2) the critical values corresponding to distinct critical points are distinct [92]. Intuitively, if $(x, y, f(x, y))$ is the surface defined by f, planes parallel to the $x-y$ plane are never

arbitrarily closely any curved object with no surface discontinuities. We also assume that S is viewed under orthographic projection. Under continuous viewpoint motion, viewpoints can be thought of as tracing a curve $c(t)$ on the viewing sphere. We refer to viewpoints as points on this sphere, rather than as viewing directions, since this simplifies the exposition in this chapter.

The shape and topology of the visible rim and the occluding contour depend on S and the viewpoint. A suitable surface parameterization relating the shape of S, the visible rim, and the occluding contour is the *epipolar parameterization* [44, 62]. When viewpoint is continuously controlled and when topology of the visible rim does not change, the epipolar parameterization uses the epipolar plane correspondences to make precise the visible rim's sliding motion over the surface and its relationship to surface shape (Section 4.2). This allows the non-concave parts of the surface to be considered as a collection of regions, each of which is a family of visible rim curves [63] (Figure 54). When this parameterization can be constructed from an image sequence, the shape of the surface in those regions (i.e., the first and second fundamental forms) can be recovered [44, 62, 170]. The details of the shape recovery process are not important for our analysis and the reader is referred to [44, 62, 170].

The crucial point in the definition of the epipolar parameterization is that the epipolar parameterization imposes four strong constraints on the ability to recover a region Π around a visible rim point p:

Epipolar Reconstructibility Constraints

C0: A surface point p must be visible from some viewpoints on its tangent plane.

C1: p must not be the endpoint of a visible rim curve. This is because it is assumed that p is not on Π's boundary.

C2: If $\mathrm{v}(t)$ is the viewpoint's velocity, $T_p(S)$ must not contain $\mathrm{v}(t)$. This is because in that case point p remains on the visible rim.

C3: The topology of the visible rim curve containing p must not change in the neighborhood of p under an infinitesimal viewpoint motion. Only a finite collection of curves on the surface cannot satisfy this constraint. These curves bound the surface points not satisfying constraint C0 (Section 6.4).

---

tangent to the surface at planar points, such planes are tangent to the surface only at a finite number of isolated points, and no such planes can be simultaneously tangent to two points on the surface.

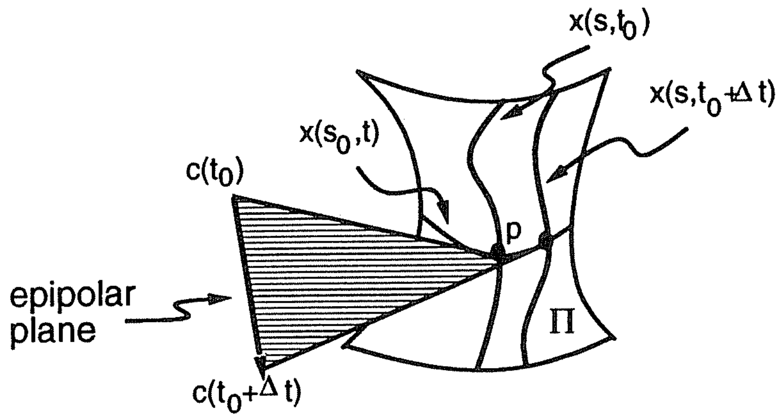


Figure 54: The epipolar parameterization. Curves $\mathbf{x}(s, t_0)$ and $\mathbf{x}(s, t_0 + \Delta t)$ are curves on the visible rim of the surface corresponding to viewpoints $c(t_0)$ and $c(t_0 + \Delta t)$, respectively. The tangent to the curve $\mathbf{x}(s_0, t)$ for $t = t_0$ is along the line through $c(t_0)$ and p. The curve's normal is in the *epipolar plane*, defined by the direction of motion, $\mathbf{v}(t)$, and the line $c(t_0)p$.

The Epipolar Reconstructibility Constraints show that the epipolar parameterization cannot be used to describe the surface in the neighborhood of every visible rim point. They also show that the surface region Π depends on how the visible rim curve $\mathbf{x}(s, t_0 + \Delta t)$ slides over the surface when Δt varies continuously. Consequently, the dynamics of the visible rim curves determine the regions reconstructed. These dynamics depend on the local and global shape of the surface as well as the viewpoint's motion.

The Epipolar Reconstructibility Constraints characterize the reconstructible regions on the surface, i.e., they tell us what is the most we can expect from any viewpoint control strategy that uses the occluding contour for reconstruction:

> **Reconstructible surface regions:** The reconstructible regions are the maximal connected sets of points for which all the Epipolar Reconstructibility Constraints can be simultaneously satisfied.

We will see that the reconstructible regions are bounded by points that satisfy constraint C0 but not constraint C3 (Section 6.4). Constraint C3 applies only to surface points belonging to visual event curves, briefly discussed below.

6.2.1 Visual Event Curves

The topology of the occluding contour of a smooth surface is stable for almost all viewpoints. Results from singularity theory show that the space of viewpoints can be partitioned into a collection of maximal connected cells within which the visible rim's and the occluding contour's topology remains constant [90, 127, 140]. *Visual events* are the boundaries of these cells. An infinitesimal perturbation of a viewpoint belonging to a visual event results in changes in the topology of the visible rim and of the occluding contour. Visual events are described in terms of the configuration of the occluding contour curves before and after the events are crossed (Figure 55). For generic surfaces, the visual events are associated with a collection of surface curves, called the *visual event curves*; only in the neighborhood of these curves can the visible rim's connectivity change when viewpoint crosses a visual event.

A presentation of the catalogue of all visual events is outside the scope of this thesis. Here we concentrate on the subset of the visual events that affect the outcome of the global reconstruction process. In particular, the visual event curves relevant to our analysis are[2] (1) the parabolic surface curves $\pi(s)$, associated with beak-to-beak and lip events, such that the line segment $\pi(s)c$ is along an asymptote at $\pi(s)$ for some viewpoint c, (2) the curves $\tau(s)$, associated with triple-point events, such that the line segment $\tau(s)c$ touches the surface at three distinct points for some viewpoint c, (3) the curves $\gamma(s)$, associated with tangent-crossing events, such that the line segment $\gamma(s)c$ touches the surface at two distinct points with identical tangent planes for some viewpoint c, and (4) the curves $\sigma(s)$, associated with cusp-crossing events, such that the line segment $\sigma(s)c$ touches the surface at two distinct points and is an asymptote at $\sigma(s)$ for some viewpoint c. A subset of these visual curves bounds the reconstructible regions. We characterize this subset in Section 6.4.

6.3 Viewpoint Control for Region Reconstruction

In this section we consider the region reconstruction task: Suppose the viewpoint is at position c, and let p be a visible rim point on the object's surface that is identified by its projection, q, on the occluding contour. The task is to continuously control viewpoint, starting from point c, in order to recover the local shape of the surface for all points in *some* neighborhood Π of p. Initially, the exploration frontier

[2] The reader is referred to Appendix D for an informal definition of these curves.

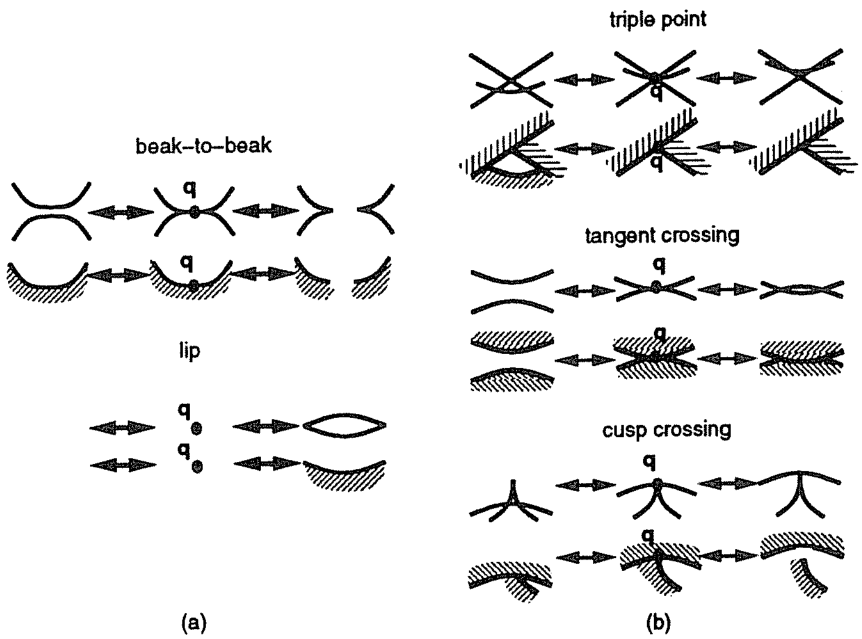


Figure 55: Visual events for a transparent surface. Also shown are examples of how these events appear when the viewed surface is opaque. The Epipolar Reconstructibility Constraint C3 is not satisfied for the visible rim point p that projects to q and is farthest away from the viewpoint's position. (a) Local events. In a beak-to-beak event two occluding contour curves meet at a point and then split off, generating two cusped contours. In a lip event a cusped contour appears out of nowhere. In both cases, p lies on a parabolic curve, π. (b) Multilocal events. In a triple-point event, points on three occluding contour segments project to a single point. Point p lies on a triple-point curve, τ. In a tangent-crossing event two contours meet creating a pair of T-junctions. p belongs to a tangent-crossing curve, γ. Finally, in a cusp-crossing event three occluding contour segments connected by two T-junctions split off with one of the segments ending with a cusp. p belongs to a cusp-crossing curve, σ.

coincides with the visible rim; a study of the region reconstruction task therefore leads to a strategy for locally controlling the motion of the exploration frontier by forcing it to slide over a neighborhood of one of its points.

We use the following two observations:

- If p is the endpoint of a visible rim curve, the epipolar parameterization cannot describe the surface in the neighborhood of p. However, there are other viewpoints on p's tangent plane at which p is not the endpoint of a visible rim curve, i.e., at which Epipolar Reconstructibility Constraint C1 is satisfied.

- The point p and the viewpoint may be such that the occluding contour's topology changes in the neighborhood of p under an infinitesimal viewpoint perturbation. For all points p except those lying on a subset of the visual event curves, we can satisfy Epipolar Reconstructibility Constraint C3 by moving to other viewpoints on p's tangent plane at which the contour's topology does not change in the neighborhood of p if these viewpoints are infinitesimally perturbed.

Based on these observations, for any given viewpoint we distinguish four types of visible rim points: *Ordinary points*, which satisfy constraint C3 and are not endpoints of a visible rim curve; *cusp points* and *T-junction points*, which satisfy constraint C3 and are visible rim endpoints projecting to a cusp and a T-junction on the occluding contour, respectively; and *degenerate points*, which are visible rim points not satisfying constraint C3. These four types of visible rim points are exhaustive and give rise to four instances of the region reconstruction task.

To perform region reconstruction we use a basic viewpoint control strategy to deal with the case where p is ordinary. The other three cases are treated by using three strategies, each corresponding to a particular type of visible rim point, that first change viewpoint to reach a viewpoint where p is ordinary, and then use the basic strategy in order to recover the shape of the surface in a neighborhood of that point. Three properties of these viewpoint controls should be pointed out that have important practical implications. These properties will be demonstrated later in the chapter.

- We will see in Section 6.6 that the strategies handling the case of degenerate visible rim points is sufficient for solving the global surface reconstruction task. This means that to achieve region reconstruction around an arbitrary visible rim point we do not have to determine whether that point is an ordinary, cusp, T-junction, or degenerate point, and we do not have to detect whether topological changes occur on the occluding contour. This property is crucial since the problems of locating or identifying cusps or degenerate points and detecting connectivity changes on the occluding contour can be very difficult in real images. Because of the sufficiency of dealing with the case of degenerate points and because this case also requires dealing with region reconstruction around ordinary points we only consider those two cases in this chapter.[3]

- The strategies used for the region reconstruction task use only two very simple types of motion:

[3] The reader is referred to [102] for a treatment of the remaining cases.

tangential viewpoint control, and motion on an appropriately selected normal plane.

- The strategies are presented in the context of recovering shape for a (possibly arbitrarily-small) region around a selected visible rim point. However, the analysis in Section 6.6, which studies the global reconstruction task, shows that when these strategies are applied in conjunction with a special set of rules, the regions reconstructed on the object's surface will not simply cover a small neighborhood of the selected point but will cover areas of the surface that are large enough to guarantee global reconstruction. For example, our simulation results of Section 6.7 show that a good approximation to the entire exterior surface of the curved pipe shown in Figure 53 and of its reconstructible interior can be obtained by applying these strategies three times.

6.3.1 Viewpoint Control for Reconstruction Around Ordinary Points

Under continuous viewpoint control, the points belonging to the visible rim are in a transitional visibility state. Some points become occluded under an infinitesimal motion along the viewpoint's path, and some remain visible but leave the visible rim. Hence, the task of forcing the visible rim to slide over a neighborhood of an ordinary visible rim point p requires (1) inducing the visibility of all points in a neighborhood of p that are occluded at the initial viewpoint, and (2) inducing the occlusion of all points in a neighborhood of p that are visible from the initial viewpoint.

Suppose viewpoint changes by tracing a smooth curve $c(t)$ with $c(t_0) = c$, and let $\mathrm{v}(t) = c'(t)$ be the instantaneous direction of motion. Given a segment $\beta(t)$ of the visible rim at viewpoint $c(t)$, the epipolar parameterization allows us to define the segment $\beta(t + \delta t)$ of the visible rim at $c(t + \delta t)$ that corresponds to $\beta(t)$. Theorem 6.1 shows that we can get a qualitative characterization of the motion of the visible rim over the surface by looking at the surface normal:

Theorem 6.1 (Visibility transition dynamics) *Suppose $N(p)$ is the surface normal at p and that the viewpoint is $c(t)$. If $\beta_1^+(t), \ldots, \beta_n^+(t)$ and $\beta_1^-(t), \ldots, \beta_m^-(t)$ are the (open) smooth segments of the visible rim that contain ordinary points p satisfying $N(p) \cdot \mathrm{v}(t) > 0$ and $N(p) \cdot \mathrm{v}(t) < 0$, respectively, then*

1. *all points $\beta_i^+(t + \delta t)$, $i = 1, \ldots, n$, are occluded from position $c(t)$.*

2. *all points* $\beta_i^-(t+\delta t)$, $i = 1, \ldots, m$, *are visible from position* $c(t)$.

3. *All ordinary visible rim points satisfying* $N(p) \cdot \mathrm{v}(t) = 0$ *will be contained in the visible rim at* $c(t+\delta t)$.

See Appendix E.1.1 for a proof. When the viewpoint changes continuously along a smooth curve $c(t)$, $t_{start} \le t \le t_{end}$, the visible rim segments will slide over the surface. If $\beta_i(t)$ is the segment containing the selected point p, $\beta_i(t)$ will trace a region Π on the surface around p that can be described using the epipolar parameterization. The boundary of this region consists of the segments $\beta_i(t_{start}), \beta_i(t_{end})$ contained in the visible rim at viewpoints $c(t_{start}), c(t_{end})$, respectively, and the traces of the endpoints of $\beta_i(t)$. The endpoints of $\beta_i(t)$ will either be points satisfying $N(p) \cdot \mathrm{v}(t) = 0$, or will be the endpoints of a visible rim curve[4]. The following viewpoint control strategy can now be used to reconstruct a surface region around p (Figure 56):

Ordinary Region Reconstruction Strategy

Step 1: Select a point p on the visible rim that is not the endpoint of a visible rim curve. This selection is done indirectly by selecting p's projection, q, on the occluding contour. Point q must not be the endpoint of an occluding contour curve.

Step 2: Compute the surface normal at p.

Step 3: (Reconstructing the occluded points near p) Select a direction v_1 for moving on the motion sphere that satisfies the inequality $N(p) \cdot \mathrm{v}_1 > 0$. Change viewpoints along v_1 while continuously monitoring the deformation of the occluding contour curve that initially contains q.

Step 4: (Reconstructing the visible points near p) Move back to the initial viewpoint and reapply Step 3 by selecting another direction of motion v_2 that satisfies the inequality $N(p) \cdot \mathrm{v}_2 < 0$.

Two components of the above strategy are purposely left unspecified. First, the choice of directions v_1 and v_2 in Steps 3 and 4 is required to satisfy a particular inequality, but no exact value is given. Second, no condition is specified for terminating the viewpoint motion in these two steps. In Section

[4]These points have also been called *frontier points* by Giblin and Weiss [61, 63] who recently provided a comprehensive study of the surface geometry around such points.

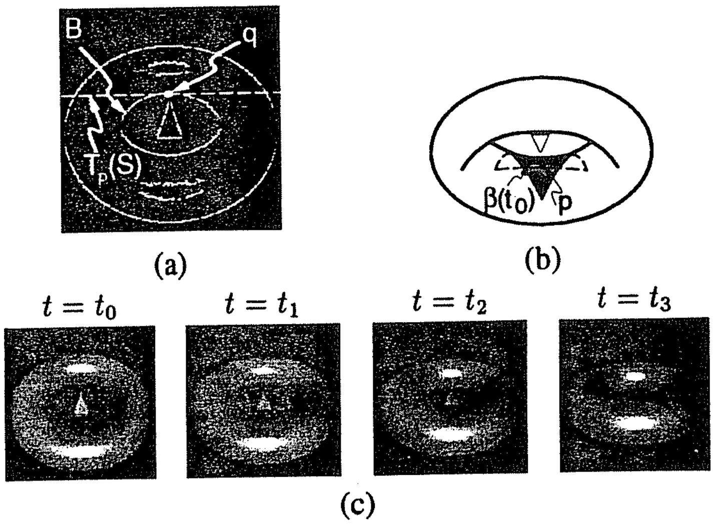


Figure 56: Reconstructing a region around an ordinary visible rim point on a torus. (a) The edges detected in the initial view, t_0. The small triangle in the middle of the torus points toward the direction of the line connecting the initial viewpoint and the center of the torus. The point selected is point p, shown in (b), in which the torus is viewed from below. The point is selected by selecting its projection, q, on the occluding contour from the initial viewpoint, $c(t_0)$. (c) Views of the surface as Step 3 of the Ordinary Region Reconstruction Strategy is applied. The tangent to the occluding contour at q is horizontal and, hence, the projection of the surface normal at p in the image is vertical. Viewpoint changes vertically downward. B is the projection of the visible rim segment $\beta(t_0)$ that contains p. Since $\beta(t)$ disappears during the viewpoint's motion, after the application of Step 3 the region reconstructed on the surface is bounded by the curves traced by the endpoints of $\beta(t)$ and by $\beta(t_0)$ (i.e., a triangle-like region). These curves belong to the exploration frontier. The region is shown as the lightly-shaded area in (b). Step 4 completes the reconstruction process around p by reconstructing a region on the other side of $\beta(t_0)$ (shown as the darkly-shaded area in (b).

6.6, where we consider the global surface reconstruction task, we show that when applying the Ordinary Region Reconstruction Strategy the viewpoint must be controlled according to a number of rules that "ground" these steps and force viewpoint to move on a normal plane of the surface at p.

6.3.2 Viewpoint Control for Reconstruction Around Degenerate Points

Suppose that the viewpoint moves along a curve $c(t)$, and that the topology of the visible rim changes in the vicinity of the visible rim point p at viewpoint $c = c(t_0)$. To reconstruct a region around p we first move to a viewpoint on $T_p(S)$ at which p is an ordinary point. This motion ensures that during the application of the Ordinary Region Reconstruction Strategy at the new viewpoint, all topological changes on the visible rim near p will be "delayed" until after a region around p is reconstructed. An important aspect of the analysis below is that this can be accomplished without answering the question of which topological changes might occur, and without detecting any such changes.

Topological changes of the visible rim occur only when the line connecting c and p has a high-order contact with the surface or when it contacts the surface at multiple points (Appendix D). Generically, if the viewpoint is infinitesimally perturbed to a new viewpoint, c', on $T_p(S)$, the line connecting c' and p will either have lower-order contact with the surface or will touch the surface at fewer points. Hence, if p is visible from the new viewpoint, p will become an ordinary visible rim point (Figure 57(a)). Unfortunately, p may no longer be visible. In this case, in order to make p ordinary the viewpoint must move to distant viewpoints on $T_p(S)$ from which p is visible (Figure 57(b)). We therefore need to specify how the viewpoint should move and when to stop.

The first question can be answered by moving either clockwise or counterclockwise on a circle in $T_p(S)$ around p. The direction of motion on this circle is not important. Viewpoint motion should stop when p becomes ordinary. It is easy to see that if there is an open arc of viewpoints on the viewpoint's motion circle from which p is visible, any viewpoint on that arc guarantees that p is an ordinary visible rim point at that viewpoint. To completely specify the viewpoint's motion, it remains to give a way of detecting when p becomes visible again. One approach is to first determine the three-dimensional coordinates of p, and then continuously check if any visible rim point with tangent plane coincident to $T_p(S)$ has those coordinates. Alternatively, to avoid dependence on such coordinate information we observe that the occluding contour must be tangent to $T_p(S)$ at p's projection. This leads to the following viewpoint-control strategy for reconstructing the surface around p (Figure 58):

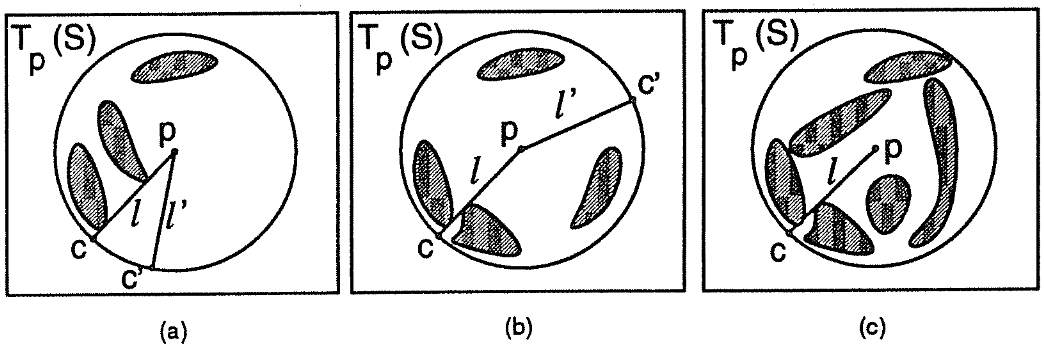


Figure 57: Forcing p to become ordinary. A top view of the tangent plane of an elliptic degenerate visible rim point p is shown. Shaded regions correspond to the intersections of $T_p(S)$ with the object. In this example, p belongs to a visual event curve associated with a triple-point event: the line through p and the viewpoint's position, c, touches the surface at three points. (a) A small viewpoint change on $T_p(S)$ makes p ordinary. (b) The geometry of the intersection $T_p(S) \cap S$ forces p to become occluded when small viewpoint changes are performed. However, there are viewpoints on $T_p(S)$ at which p is ordinary. (c) The geometry of the intersection $T_p(S) \cap S$ forces p to be occluded at all viewpoints except c.

Degenerate Region Reconstruction Strategy

Step 1: Let $p(t_0 - \delta t)$ be the visible rim point at position $c(t_0 - \delta t)$ that is matched to p by the epipolar parameterization. Compute the tangent plane at p as the limit $\lim_{\delta t \to 0} T_{p(t_0 - \delta t)}(S)$.

Step 2: Perform a small counterclockwise motion on $T_p(S)$. If p remains visible, set $q = p$ and continue with Step 5. Otherwise, return to the initial viewpoint, c.

Step 3: Perform a small clockwise motion on $T_p(S)$. If p remains visible, stop. Otherwise, return to the initial viewpoint, c.

Step 4: Move clockwise on a circle around p on $T_p(S)$ until either c is reached again or there is an ordinary visible rim point q whose tangent plane coincides with $T_p(S)$.

Step 5: If the initial viewpoint is reached, stop. Otherwise, apply the Ordinary Region Reconstruction Strategy to reconstruct a region around q, and continue with Step 4.

With the above strategy, region reconstruction is not achieved for p if and only if p is occluded from all but a finite set of viewpoints on its tangent plane. Such points never become ordinary during

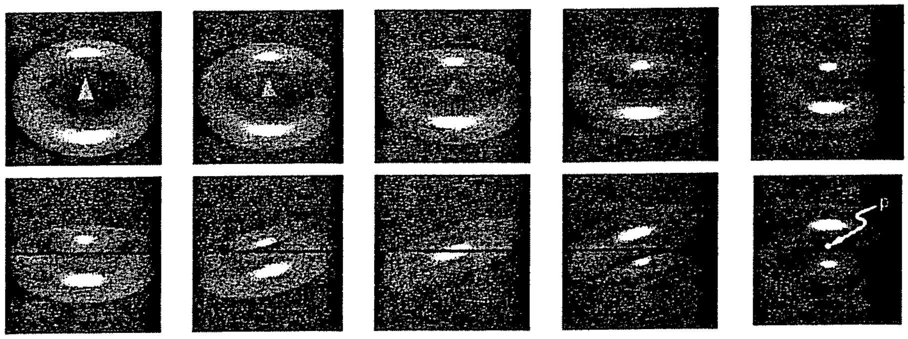


Figure 58: Reconstructing a region around a degenerate point on the torus. *Top row:* Viewpoint moves downward until the visible rim segment pointed by the triangle shrinks to a point p and disappears. The tangent plane at p is horizontal and perpendicular to the plane of the page. The visual event corresponding to the disappearance of that segment is a tangent-crossing event. Due to this event, the surface in the neighborhood of p cannot be reconstructed by performing a small viewpoint change. *Bottom row:* Moving on $T_p(S)$ in order to make p ordinary. The black horizontal line is the projection of $T_p(S)$ in the image. A clockwise viewpoint change is performed on $T_p(S)$ until an ordinary visible rim point with tangent plane identical to $T_p(S)$ is detected. After performing a 180° rotation, such a visible rim point is found; in this case the point is p. We can now use the Ordinary Region Reconstruction Strategy to reconstruct a surface region around p.

the viewpoint's motion on $T_p(S)$ (Figure 57(c)). This is not the fault of the Degenerate Region Reconstruction Strategy, however; in this case Epipolar Reconstructibility Constraint C3 cannot be satisfied because there are no motions that force the visible rim to slide over a neighborhood of such a point.

6.4 The Reconstructible Surface Regions

The analysis of the region reconstruction task gives us a way to characterize the reconstructible regions on the surface by characterizing their boundaries[5]. In particular, the strategies described in Section 6.3 allow us to reconstruct a region around all surface points except for (1) points that are never visible

[5]Our characterization is similar, in spirit, to the *visual hull* concept recently introduced by Laurentini [105]. The visual hull is the closest approximation to a non-convex polyhedral object that can be reconstructed from images of the object's silhouette. However, both our goals (characterization of reconstructible regions on smooth objects) and the tools we use for this characterization (differential geometry) differ.

from viewpoints on their tangent plane, and (2) points on visual event curves that are visible from only a finite number of viewpoints on their tangent plane. For these points, there is no viewpoint motion that will force the visible rim to slide over their neighborhoods. This leads directly to the following characterization of the reconstructible regions on the surface:

> **Reconstructible surface regions:** The reconstructible surface regions are the maximal connected sets of points that are visible from a one-dimensional set of viewpoints on their tangent plane.

Unless the surface is entirely reconstructible, each connected reconstructible surface region $\mathcal{R}$ forms an open set on the surface. Its boundary contains the segments of visual event curves whose points are visible only from a finite number of viewpoints on their tangent plane. By checking which visual event curves can contain such segments, it is easy to show the following:

Theorem 6.2 (Reconstructible region boundaries) *A point p is on the boundary of a reconstructible surface region if and only if it belongs to either*

- *a parabolic curve bounding a surface concavity,*
- *a curve $\tau(s)$ associated with a triple-point event,*
- *a curve $\gamma(s)$ associated with a tangent-crossing event, or*
- *a curve $\sigma(s)$ associated with a cusp-crossing event,*

and is visible from only a finite number of viewpoints on its tangent plane.

An intuitive description of Theorem 6.2 can be given as follows. To each point p in a reconstructible surface region we can associate a collection of *visibility arcs*. These arcs are simply the open and connected one-dimensional sets of viewpoints on p's tangent plane from which p is visible (Figure 59). When p asymptotically approaches one of the above visual event curves, the length of at least one of its visibility arcs decreases, diminishing to zero (Figure 59(a)-(c)); this can only happen for the visual event curves listed above. Now, if the lengths of *all* visibility arcs of p diminish to zero, the visual event curve approached by p belongs to the boundary of a reconstructible region (Figure 59(d)-(f)).

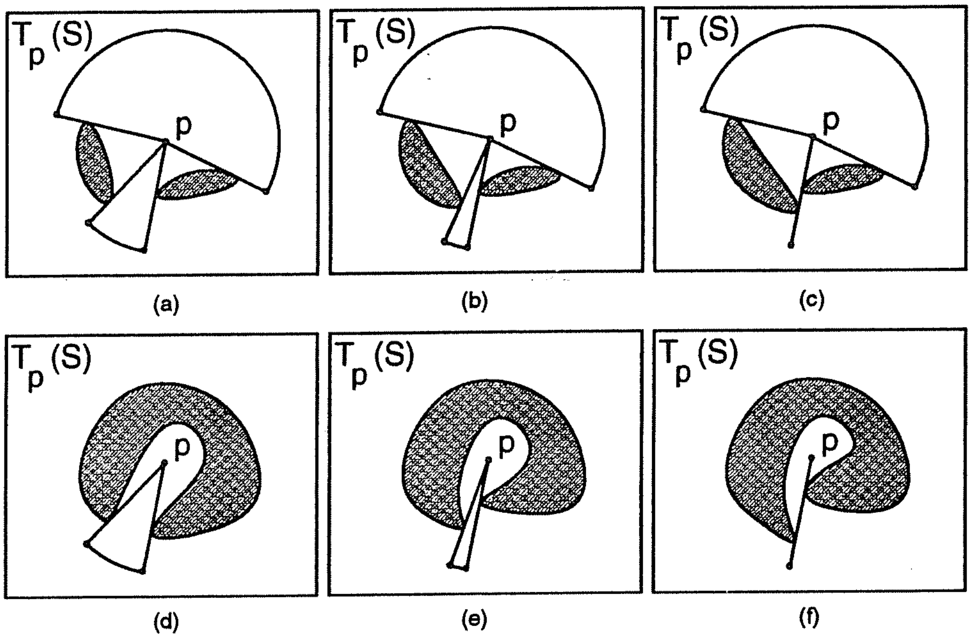


Figure 59: The visibility arcs of a point p. A top view of the tangent plane of p is shown. Shaded areas correspond to the intersections of $T_p(S)$ with the object. (a)-(c) Approaching a visual event curve τ associated with a triple-point event. Point p has two visibility arcs. As p approaches τ, one of the visibility arcs of p degenerates to a point. In this case, the point p in (c) belongs to τ, but is not contained in the boundary of a reconstructible surface region; the neighborhood around p can be reconstructed by moving to a viewpoint in the remaining visibility arc of p. (d)-(f) Approaching a visual event curve τ associated with a triple-point event. Point p now has one visibility arc. As p approaches τ, the visibility arc of p degenerates to a point. In this case, p asymptotically approaches the boundary of a reconstructible surface region.

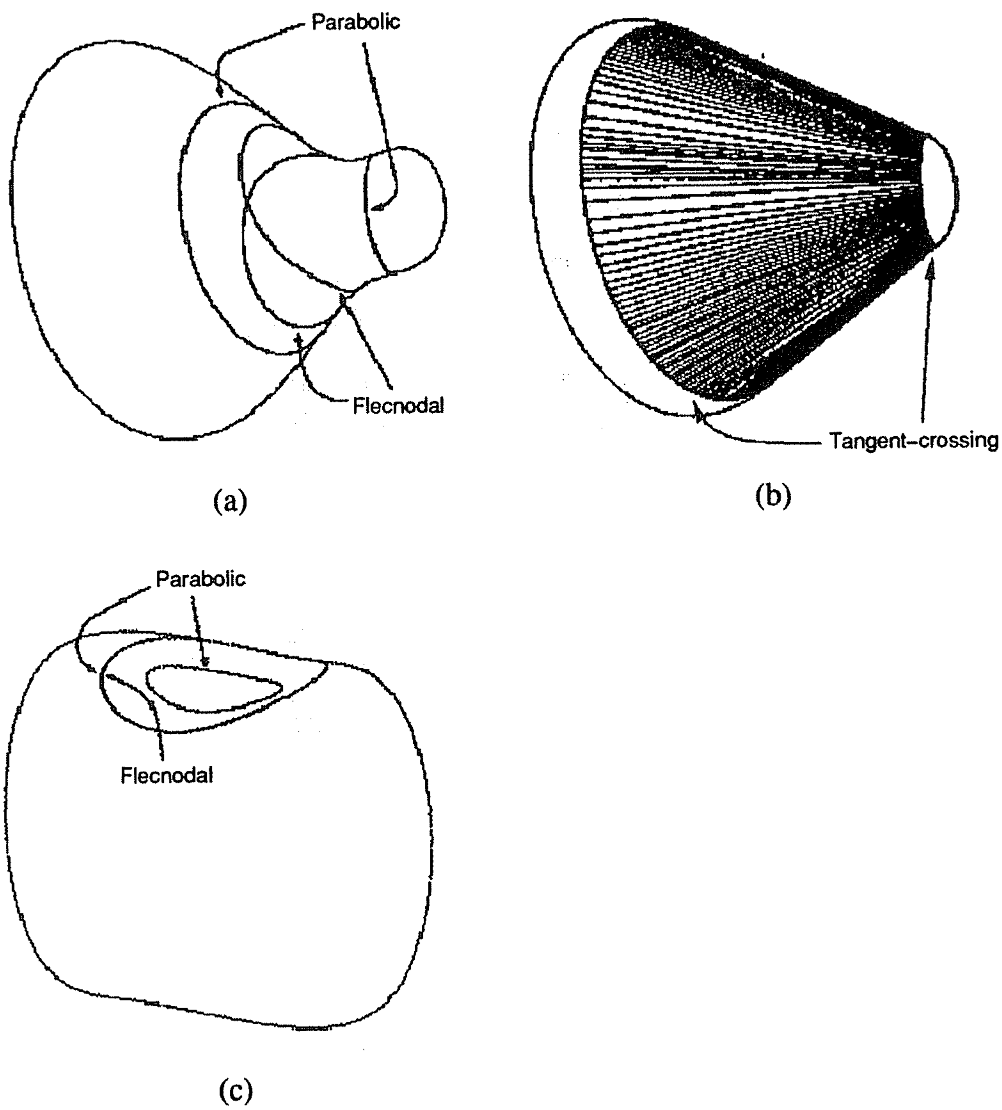


Figure 60: Applying Theorem 6.2 to some surfaces studied by Petitjean *et al.* [127] and Koenderink [90]. The figures are taken from [127]. The squash-shaped surface in (a) and (b) is completely reconstructible: No cusp-crossing or triple-point visual events can occur, and the surfaces do not have concavities. Furthermore, for each point on the two visual event curves corresponding to a tangent-crossing event (shown in (b)), we can associate at least one visibility arc. The dimple-shaped surface in (c) has one reconstructible region. This region is bounded by the parabolic curve bounding the concavity on the surface.

The visual event curves listed in Theorem 6.2 are therefore *potential boundaries* of a reconstructible surface region. They bound such regions only if they contain points with no visibility arcs. Figures 60, 61 show the reconstructible regions for three objects.

6.5 Viewpoint Control for Incremental Surface Reconstruction

The goal of the global surface reconstruction task is to reconstruct the reconstructible surface regions that intersect the visible rim at the initial viewpoint. To achieve this, we incrementally "grow" the regions initially reconstructed on the surface by answering two questions: (1) How can we force points on the exploration frontier to lie on the visible rim, and (2) how can we control viewpoint so that new regions around those points can be recovered?

The first question can be answered by considering the fact that the exploration frontier is traced by points on the visible rim from previous viewpoints. Hence, it suffices to move back to the viewpoint where a given frontier point belongs to the visible rim. This can be achieved by saving, along with each occluding contour image, the viewpoint corresponding to that image during the application of the Ordinary Region Reconstruction Strategy. Since there is a correspondence between the points on

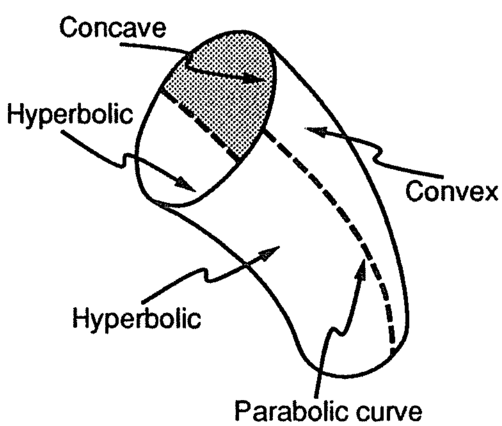


Figure 61: Reconstructible regions for the pipe surface of Figure 53. The surface has one reconstructible region, which is the union of the convex and hyperbolic areas of the pipe, and one unreconstructible region (shown in gray), corresponding to the pipe's upper interior surface, which is concave.

the exploration frontier and the images they project to, this information is sufficient to move to the viewpoint where a particular boundary point was on the visible rim. Furthermore, since any frontier point can be forced to become a visible rim point, and the strategies developed in Section 6.3 can be used to reconstruct a surface region around any visible rim point for which this is possible, the second question is easily answered by using the strategies already presented. These considerations lead to the following strategy:

Incremental Surface Reconstruction Strategy

Step 1: If the exploration frontier has not been reduced to a point, select a point p on the frontier and let c be the viewpoint at which p projected to the occluding contour.

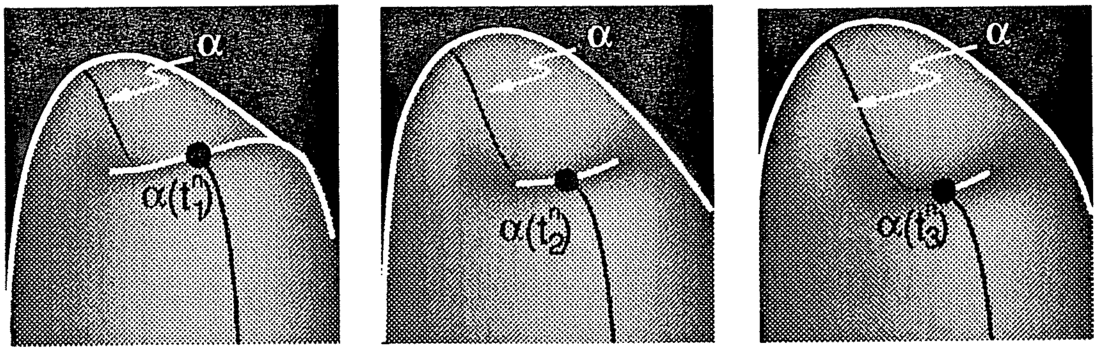
Step 2: Move to c.

Step 3: Use the region reconstruction strategy appropriate for performing the local surface reconstruction task around p, and continue with Step 1.

The specific algorithm for selecting the points p on the boundary of the already-reconstructed surface regions is not important for guaranteeing their successive expansion. However, in order to perform the global surface reconstruction task we must obey an additional rule when doing this selection; we discuss this rule in Section 6.6.

6.6 Global Surface Reconstruction

What kinds of strategies are needed to accomplish global surface reconstruction? In Section 6.1 we motivated the need for strategies for which (1) the reconstruction process does not terminate (i.e., at least one of the component strategies is applied an infinite number of times) if and only if there is no finite-length viewpoint path that accomplishes global surface reconstruction (*finite termination*), and (2) if the reconstruction process terminates, the reconstructed points must be the union of the reconstructible regions intersecting the visible rim at the initial viewpoint, and if it does not terminate the reconstructed points must asymptotically approach that set (*completeness*).



(a) (b) (c)

Figure 62: Difficulties involved in globally reconstructing a dimple-shaped surface. In the n-th iteration of the Incremental Surface Reconstruction Strategy the viewpoint is moving upward in order to reconstruct points in the neighborhood of the visible rim point $\alpha(t_1^n)$ on curve α. The visible rim eventually slides to the right, making the viewpoint's upward motion ineffective for reconstructing the surface in the vicinity of $\alpha(t_3^n)$.

In this section we show that global surface reconstruction can be achieved by (1) using the Incremental Surface Reconstruction Strategy, while (2) obeying a number of simple rules that constrain how that strategy is applied. The idea behind these rules is simple: Their goal is to constrain the viewpoint controls during each iteration of the Incremental Surface Reconstruction Strategy so that all four Epipolar Reconstructibility Constraints are satisfied for as long as possible during the viewpoint's motion, and for as large a region on the surface as possible. It is a basic result of our analysis that this approach guarantees global surface reconstruction.

Motivated by our characterization of the reconstructible regions and the difficulties illustrated in Figures 53 and 62 that we need to overcome, we develop these rules by considering the following three increasingly more general global reconstruction tasks:

- **Semi-global curve reconstruction task:** Suppose a curve is drawn on the surface so that it intersects the visible rim at the initial viewpoint (Figure 62(a)). The task is to reconstruct the segments of this curve that are connected, reconstructible, intersect the visible rim at the initial viewpoint, and terminate on a visual event curve that *potentially* bounds a reconstructible surface region, as in Figures 59(c, f).

- **Global curve reconstruction task:** Suppose a curve is drawn on the surface so that it intersects the visible rim at the initial viewpoint. The task is to reconstruct the segments of this curve that are connected, reconstructible, intersect the visible rim at the initial viewpoint, and terminate on the *boundary* of a reconstructible region.

- **Global surface reconstruction task:** Global surface reconstruction is a generalization of the global curve reconstruction task in the following sense. It is equivalent to reconstructing for every surface curve that intersects the visible rim at the initial viewpoint, a connected, reconstructible segment terminating on the boundary of a reconstructible region.

By obeying the rules we develop in this section, we "ground" the steps in the Incremental Surface Reconstruction Strategy and the Region Reconstruction Strategies that we left unspecified in Sections 6.3 and 6.5. In the following we keep our analysis at a fairly intuitive level, working through specific examples to motivate the rules. Proofs can be found in Appendix E.

6.6.1 Semi-Global Curve Reconstruction

Recall that during the application of the Incremental Surface Reconstruction Strategy we select a point p on the boundary of the already-reconstructed regions and then control viewpoint to reconstruct a new region around p. To achieve semi-global curve reconstruction, the length of the curve segment reconstructed at each iteration must diminish if and only if it asymptotically approaches a visual event curve *potentially* bounding a reconstructible surface region. The following theorem gives the three rules that must be obeyed in order to accomplish the semi-global curve reconstruction task.

Theorem 6.3 (Semi-global curve reconstruction rules) *Let α be a finite-length curve drawn on the surface. If α intersects the visual event curves at most a finite number of times, and the following three rules are obeyed, the semi-global curve reconstruction task will be accomplished:*

Rule 1: *When choosing the point p on which to apply the Region Reconstruction Strategies, always select a point of intersection of α with the visible rim.*

Rule 2: *Always apply the Ordinary Region Reconstruction Strategy after first moving to a viewpoint c corresponding to the middle of a visibility arc of p.*

Rule 3: *When applying the Ordinary Region Reconstruction Strategy to reconstruct a region around p starting from an initial viewpoint c, move around the surface on the normal plane at p and stop only after the endpoint of the segment of α being reconstructed coincides with a cusp, T-junction or degenerate visible rim point, or c is reached again.*

Rules 1 and 3 are obvious. For example, consider the semi-global curve reconstruction task for the surface in Figure 62(a). The n-th iteration of the Incremental Surface Reconstruction Strategy requires selecting a point on the visible rim in order to reconstruct a region in its neighborhood. Rule 1 simply states that the point selected should be $\alpha(t_1^n)$. This rule ensures that a segment around $\alpha(t_1^n)$ will be reconstructed even when that point is a degenerate visible rim point. Now suppose viewpoint starts moving in an upward direction according to the Ordinary Region Reconstruction Strategy in order to reconstruct a region around $\alpha(t_1^n)$. Rule 3 states that the viewpoint should move upward at least until the cusp endpoint of the visible rim coincides with the dark curve at $\alpha(t_3^n)$, as shown in Figure 62(c). Clearly, there is no reason for continuing to move upward after that point since reconstruction of a larger piece of the dark curve around that point will not occur. By constraining motion on a normal plane it also ensures that Epipolar Reconstructibility Constraint C2 will be satisfied for as many points as possible in the neighborhood of the selected point after a small viewpoint motion.

The utility of Rule 2 is not as obvious, although it is crucial for achieving the semi-global curve reconstruction task because it constrains the long-range effect of the viewpoint's motion on the set of reconstructed surface points. Intuitively, Rule 2, together with Rule 3, ensures that Epipolar Reconstructibility Constraints C1 and C3 are satisfied for as long as possible during the upward motion of the viewpoint. To see how this is achieved in the semi-global reconstruction of α, suppose the viewpoints corresponding to Figures 62(a)-(c) are $c(t_1^n)$, $c(t_2^n)$ and $c(t_3^n)$, respectively, and the line through $c(t)$ and $\alpha(t)$ is $l(t)$. To achieve semi-global curve reconstruction, the length of the segment between $\alpha(t_1^n)$ and $\alpha(t_3^n)$ must diminish if and only if $\alpha(t_1^n)$ asymptotically approaches a visual event curve that potentially bounds a reconstructible surface region. Now consider Figure 63. Since $\alpha(t_3^n)$ is a cusp point, the line $l(t_3^n)$ is along an asymptote at $\alpha(t_3^n)$. Therefore, if $\psi(t)$ is the angle between $l(t)$ and the corresponding asymptote at $\alpha(t)$, we can conclude that a necessary and sufficient condition for the curve point $\alpha(t_3^n)$ to become a cusp visible rim point is that $\psi(t)$ goes to zero as t approaches t_3^n.

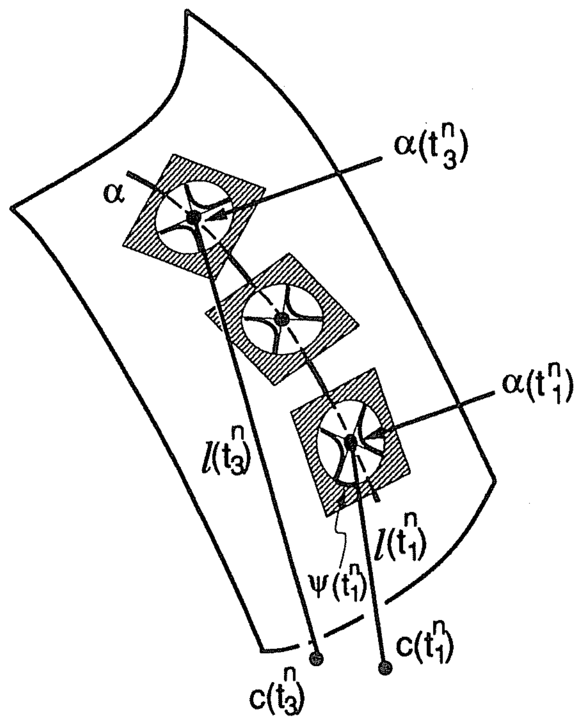
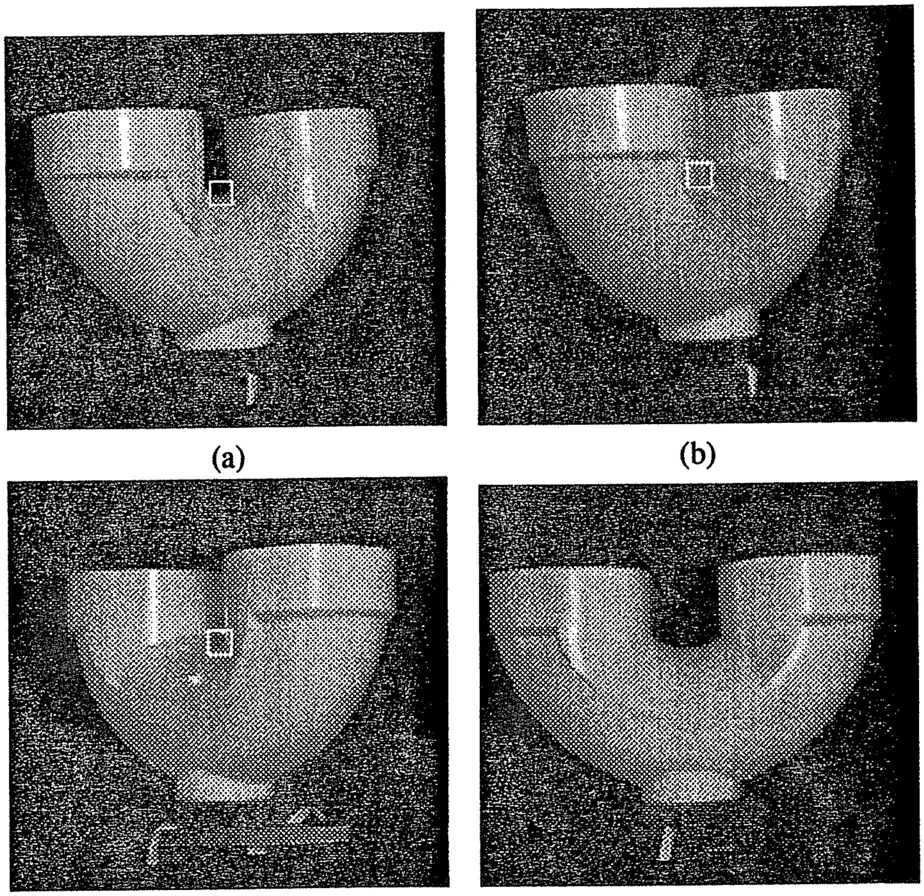


Figure 63: Geometry of the reconstruction of a segment of curve α in Figure 62. Viewpoints $c(t_1^n)$, $c(t_3^n)$ correspond to Figures 62(a) and (c), respectively. The tangent plane and Dupin's indicatrix of points $\alpha(t_1^n)$ and $\alpha(t_3^n)$ is also shown.

Clearly, if $\psi(t_1^n)$ is large, the length of the segment between $\alpha(t_1^n)$ and $\alpha(t_3^n)$ will also be large. It is therefore necessary to require $\psi(t_1^n)$ to be large. But how large can we make $\psi(t_1^n)$? If $\psi(t_1^n)$ is too large, the line $l(t_1^n)$ may approach the other asymptote at $\alpha(t_1^n)$; the best we can do is to ensure that $c(t_1^n)$ is in the *middle* of the visibility arc, which in this case is bounded by the two asymptotes at $\alpha(t_1^n)$. At that viewpoint, $\psi(t_1^n)$ will form equal angles with both asymptotes at $\alpha(t_1)$.

Obeying Rule 2 is quite easy: We determine the extent of the visibility arc containing $c(t_1^n)$, and then move to the middle of that arc. To measure the extent of the visibility arc, we can simply move on $T_{\alpha(t_1^n)}(S)$ first in a clockwise and then in a counterclockwise direction, stopping when a cusp or T-junction is formed at the projection of $\alpha(t_1^n)$. Figures 64-66 show results from a real-time implementation of this rule.



(a) (b) (c) (d)

Figure 64: Implementing the viewpoint control strategies required for global reconstruction: Moving to the middle of the visibility arc for a point on a pipe by tangential viewpoint control (Rule 2). The point is at the center of the white square. (a) Initial view. (b) Moving to one of the visibility arc's endpoints. Object rotations were performed at speeds of 40 degrees/second. The arc endpoints are reached when tracking fails. (c) Moving to the other visibility arc endpoint. The view was again automatically obtained, as in (c). (d) Moving to the middle of the point's visibility arc. The pan position corresponding to this view was obtained from the pan positions corresponding to the arc's boundaries.

By following the above rules, semi-global curve reconstruction is accomplished: The distance between $\alpha(t_1^n)$ and $\alpha(t_3^n)$ will diminish if and only if the visibility arc at $\alpha(t_1^n)$ degenerates to a point. This occurs only when $\alpha(t_1^n)$ approaches one of the visual event curves listed in Theorem 6.2.

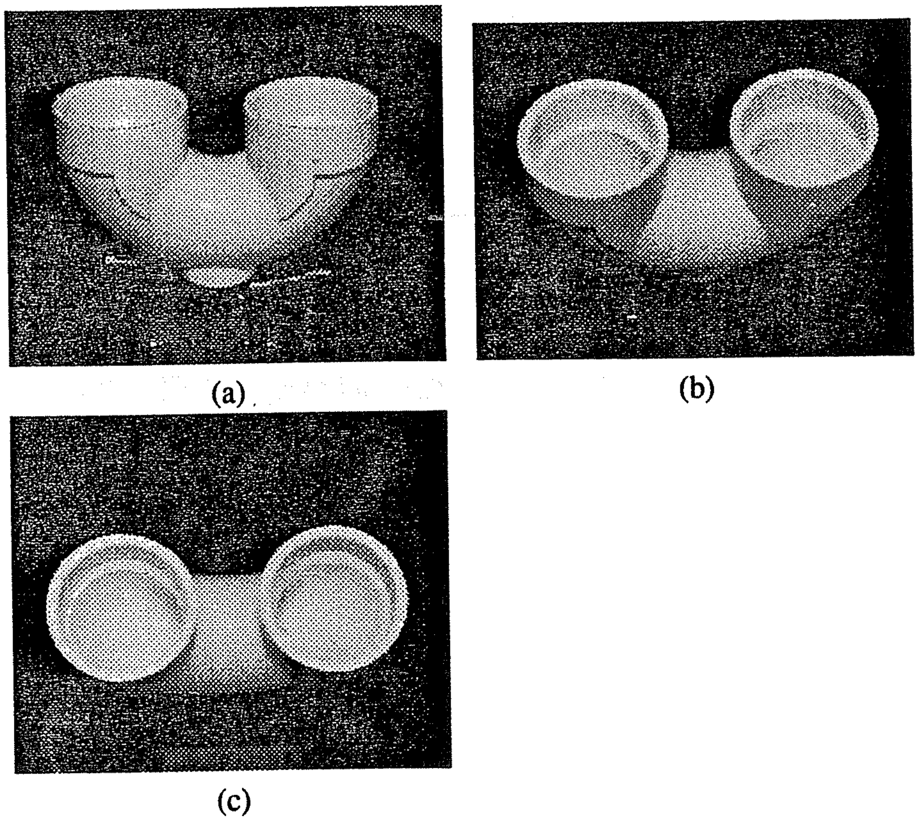


Figure 65: Implementing the viewpoint control strategies required for global reconstruction: Changing viewpoint on the normal plane of the selected point (Rule 3).

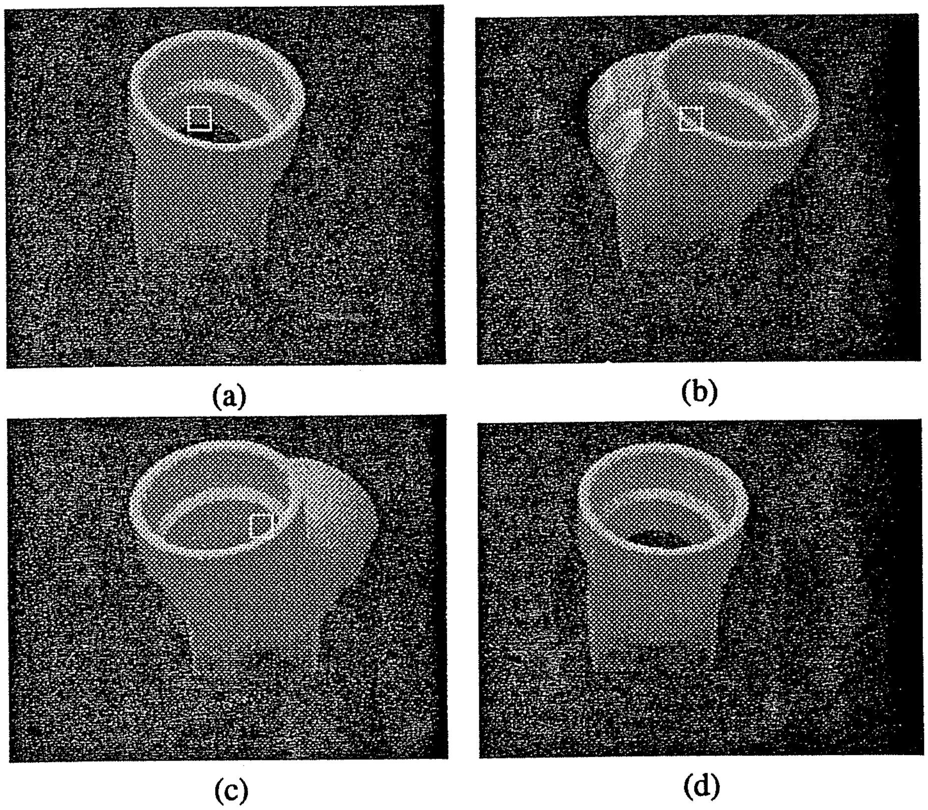


Figure 66: Implementing the viewpoint control strategies required for global reconstruction: Finding the middle of the visibility arc of a point on a different object. (g) Initial view. (h) View corresponding to one of the arc's endpoints. (i) View corresponding to the other arc endpoint. (k) View corresponding to the middle of the point's visibility arc.

6.6.2 Global Curve Reconstruction

In the global curve reconstruction task we must reconstruct the whole curve drawn on the surface if this is possible, or a segment of that curve whose endpoints either lie on or asymptotically approach the boundary of a reconstructible surface region. This task is harder to accomplish than semi-global curve reconstruction because if the whole curve cannot be reconstructed, the endpoints of the reconstructed segment must lie on visual event curves that are *actual*, not just *potential*, boundaries of a reconstructible surface region. To achieve this task the three rules guaranteeing semi-global reconstruction of a curve must be obeyed. The following theorem shows that in addition to these rules, a fourth rule is also necessary. Its proof follows from the proof of Theorem 6.3 and is omitted.

Theorem 6.4 (Global curve reconstruction rules) *Let α be a finite-length curve drawn on the surface. If α intersects the visual event curves at most a finite number of times, and the following four rules are obeyed, the global curve reconstruction task will be accomplished:*

Rule 1: *When choosing the point p on which to apply the Region Reconstruction Strategies, always select a point of intersection of α with the visible rim.*

Rule 2: *Always apply the Ordinary Region Reconstruction Strategy after first moving to a viewpoint c corresponding to the middle of a visibility arc of p.*

Rule 3: *When applying the Ordinary Region Reconstruction Strategy to reconstruct a region around p starting from an initial viewpoint c, move around the surface on the normal plane at p and stop only after the endpoint of the segment of α being reconstructed coincides with a cusp, T-junction or degenerate visible rim point, or c is reached again.*

Rule 4: *In order to reconstruct a region around the selected point p, always apply the Degenerate Region Reconstruction Strategy independently of whether p is ordinary, cusp, T-junction or degenerate.*

Rules 1-3 are identical to those used for accomplishing the semi-global reconstruction task. To see why the fourth rule is necessary for global curve reconstruction, suppose that only the first three rules are obeyed to perform the task for the curve in Figure 62. In this case, the length of the segment reconstructed at the n-th iteration of the Incremental Surface Reconstruction Strategy will diminish even when these endpoints asymptotically approach a point on a visual event curve that does not bound a reconstructible surface region (Figure 59(c)). These points have at least one visibility arc.

Rule 4 requires reconstruction of several regions around the selected point p, by moving to the middle viewpoints of *all* the visibility arcs of p and then applying the Ordinary Region Reconstruction Strategy starting at each one of those viewpoints. By obeying this rule, the length of the reconstructed segment can diminish only if *all* visibility arcs at p diminish. Since this occurs only when p approaches a point bounding a reconstructible surface region, Rules 1-4 guarantee global curve reconstruction. Adding Rule 4 also implies that the reconstruction process is simplified: The Ordinary Region Reconstruction Strategy and the Degenerate Region Reconstruction Strategy are sufficient to accomplish global curve reconstruction.

6.6.3 Global Surface Reconstruction

In this section we consider the global surface reconstruction task. To accomplish this task we must now reconstruct not only points lying on a single surface curve that intersects the visible rim at the initial viewpoint, but must also reconstruct points lying on *every* such curve that can be drawn on the surface. The following theorem shows how this task can be accomplished using the Incremental Surface Reconstruction Strategy and the Ordinary and Degenerate Region Reconstruction Strategies (Figure 67):

Theorem 6.5 (Global surface reconstruction rules) *If the following four rules are obeyed, global surface reconstruction will be accomplished.*

Rule 1: *Always choose the frontier points p on which to apply the Region Reconstruction Strategies so that the already-reconstructed surface regions expand in every direction after a finite number of iterations.*

Rule 2: *Always apply the Ordinary Region Reconstruction Strategy after first moving to a viewpoint c corresponding to the middle of a visibility arc of p.*

Rule 3: *When applying the Ordinary Region Reconstruction Strategy to reconstruct a region around p starting from an initial viewpoint c, move around the surface on the normal plane at p and stop* *only after the the visible rim segment initially containing p (and all visible rim segments splitting from or merging with it) disappears, or c is reached again.*

Rule 4: *In order to reconstruct a region around the selected point p, always apply the Degenerate Region Reconstruction Strategy independently of whether p is ordinary, cusp, T-junction, or degenerate.*

Rules 2 and 4 are identical to those for global curve reconstruction. Rule 3 is a generalization of the corresponding rule used for global curve reconstruction in the following sense. When performing global curve reconstruction, as in the example in Figure 62(a)-(c), viewpoint motion was required to stop only after an endpoint of the visible rim "slid over" the curve drawn on the surface. For global surface reconstruction, the same rule must hold for *every* curve that we can draw on the surface. This requires the viewpoint to move upward at least until the visible rim segment in Figure 62(a) containing $\alpha(t_1^n)$ disappears (or, equivalently, the two segments β_1, β_2 in Figure 62(c) disappear), or the initial viewpoint is reached again.

Rule 1 is also a generalization of the corresponding rule used for global curve reconstruction. It requires reconstruction of a region in the neighborhood of every curve that can be drawn on the surface, intersects the visible rim at the initial viewpoint, and intersects the boundary of the reconstructed points after a finite number of iterations of the Incremental Surface Reconstruction Strategy. This rule is necessary because if the reconstruction process does not terminate and the reconstructed region is expanded in only one direction, some pieces of the boundary of the already-reconstructed points will never approach the boundary of a reconstructible region.

By obeying Rules 1-4, global surface reconstruction is guaranteed. The Incremental Surface Reconstruction Strategy terminates after a finite number of steps precisely when the whole surface is reconstructible. Otherwise, the set of points reconstructed converges to the reconstructible surface regions intersecting the visible rim at the initial viewpoint. Furthermore, in this case, *no* strategy can achieve global surface reconstruction in a finite number of steps: When the visible rim touches a visual event curve on the boundary of a reconstructible surface region, it touches it at exactly one point, making it impossible to reconstruct the surface in every neighborhood of such a curve in a finite number of steps.

Incremental Surface Reconstruction Strategy	
Rules that must be obeyed	
Rule 1:	Select points uniformly on the exploration frontier
Rule 4:	Always apply the Degenerate Region Reconstruction Strategy independently of the type of p

Degenerate Region Reconstruction Strategy	
Rule that must be obeyed	
Rule 2:	Always move to the middle of a visibility arc

Ordinary Region Reconstruction Strategy	
Rule that must be obeyed	
Rule 3:	Move on a great circle around the surface and stop when either the initial position is reached again, or the visible rim segment initially containing p (and all segments splitting from it or merging with it) disappear

Figure 67: Strategies used to accomplish global surface reconstruction. Also shown are the rules that must be obeyed when each of the strategies are applied.

6.7 An Example: Reconstructing a Curved Pipe

In order to study the viewpoint-control strategies we have defined, we implemented a system that allows us to visualize (1) the motions generated during an object's global reconstruction, and (2) the surface regions reconstructed during those motions. The input to the system consists of a polyhedral representation of the object (typical sizes are 40,000 polygons), an initial viewpoint, and a point on the object's visible rim at the initial viewpoint. Given this input, the system automatically generates the motions prescribed by the Incremental Surface Reconstruction Strategy and Region Reconstruction Strategies while conforming to the rules guaranteeing global surface reconstruction in order to incrementally reconstruct regions on the object's surface (Figure 67).

The simulator also marks the parts of the surface over which the visible rim slides during the generated motions (i.e., it marks the portions of the object that would be reconstructed due to these motions). At each viewpoint, the object's visible rim and its occluding contour are computed using efficient ray-tracing techniques[6] [57]. Below, we briefly present results of using this system to simulate the reconstruction of the surface of the curved pipe shown in Figure 53. These results can be evaluated in three ways: (1) The surface regions reconstructed, (2) the number of times the Region Reconstruction Strategies were applied to complete the reconstruction process, and (3) the generated motions. The next three subsections consider each of these issues.

6.7.1 Reconstructed Regions

The global geometry of the curved pipe is shown in Figure 61. Its surface has only one reconstructible and one unreconstructible region. These regions are bounded by the two parabolic curves on the pipe's interior surface. In our simulation, the motions generated by the Region Reconstruction Strategies resulted in complete reconstruction of the pipe's exterior surface. The reconstructed portions of the pipe's interior surface (i.e., the surface of the hole) are shown in Figure 68. The figure shows that this part of the reconstructed region closely approximates the reconstructible hyperbolic region of the pipe's interior surface. In particular, the boundaries of the reconstructed region are close to the two parabolic curves bounding the pipe's interior reconstructible region. This result is a direct consequence of the convergence property of the strategy we use for global surface reconstruction: The strategies guarantee that the reconstructed region will grow arbitrarily close to the boundary of the pipe's reconstructible region as long as the Region Reconstruction Strategies are applied a sufficient number of times.

[6]Frames are typically generated at a rate of 1 frame/second.

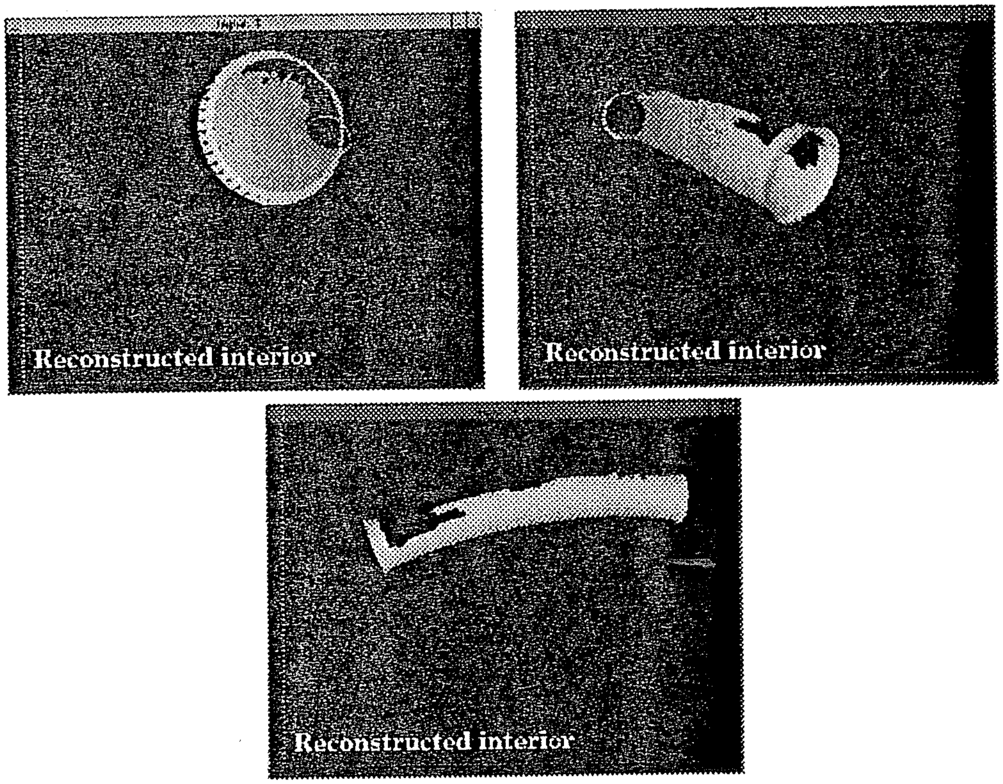


Figure 68: Three views of the region reconstructed on the pipe's interior surface.

6.7.2 Number of Applications of the Region Reconstruction Strategies

Even though the convergence property of the strategy we use for global surface reconstruction ensures that the regions incrementally grown on the object's surface will *eventually* approach an object's reconstructible region, no bounds are given on the number of times they might need to be applied to get good approximations to an object's reconstructible region. One of the most important observations that can be made from our simulations is that the number of applications needed for objects such as the curved pipe is small. For the curved pipe, only three applications of the Degenerate Region Reconstruction Strategy were sufficient to reconstruct the object's entire exterior surface and to get a good approximation to the object's reconstructible interior.

This result emphasizes an important property of the Region Reconstruction Strategies when they are applied in conjunction with Rules 1-4 of Theorem 6.5: Even though the motions generated by these strategies can be performed using only local computations in the image (e.g., tracking a single point while moving on its tangent plane), the regions reconstructed due to these motions can cover large areas of the surface.

6.7.3 Generated Motion

Figure 70 shows the path traced by the moving viewpoint during the reconstruction process. This path is a connected set of arcs contained in the viewpoint's motion sphere. The path was produced by applying the Degenerate Region Reconstruction Strategy at three points on the pipe's surface. The following pages show a sequence of snapshots taken from a real-time animation of the entire reconstruction process for the pipe. The yellow curves on the pipe's surface correspond to the visible rim. The visible rim point selected at each iteration of the Incremental Surface Reconstruction Strategy is indicated by a red arrow, pointing in the direction of the point's surface normal. The pipe's surface is rendered in two colors, indicating the unreconstructed (ivory) and reconstructed (green) regions of the surface. The reconstructed regions expand as the visible rim slides over the surface due to the viepwoint's motion.

As prescribed by the Incremental Surface Reconstruction Strategy and the Region Reconstruction Strategies, the generated motions involve (1) motion on the tangent plane of the selected visible rim point, (2) motion on a normal plane at a selected point, and (3) motion to a previous position on the traced path, in order to force a point on the boundary of a reconstructed region to become a visible rim point. A brief explanation of these motions for each of the three visible rim points used for the pipe is given below.

Point 1: Point 1 is on the visible rim at the initial viewpoint and was given as input to the system. The first motion performed involves motion on the point's tangent plane in order to determine the extent of the point's visibility arc. Since the point is on the object's convex hull it is visible from every viewpoint on its tangent plane, causing Point 1 to be circumnavigated during this step. After the point's circumnavigation, a motion on the point's normal plane is performed. In this phase a number of "shortcuts" that reduce the amount of motion can be used. In particular, the normal motion can terminate when the visible rim curves are entirely contained in a reconstructed region of the object's surface.[7]

Point 2: This point was selected from the boundary of the already-reconstructed surface region. To minimize viewpoint motion, the point selected was the point requiring the least amount of motion to force its inclusion on the visible rim. The motions generated from the application of the Degenerate Region Reconstruction Strategy illustrate that even though no information about the surface's global structure is used, the generated motions appear to take into account the object's global shape. For example, reconstruction of the pipe's interior around Point 2 causes the pipe's interior to be observed through both ends of the pipe.

Point 3: This point was selected from the boundary of the already-reconstructed surface region. The reconstruction process terminated immediately after the application of the Degenerate Region Reconstruction Strategy for this point. This occured because the strategy's application caused a minimal increase in the area of the already-reconstructed region on the surface.

The image sequence below is partitioned into the following four groups, each corresponding to a distinct step of the strategy we use for global surface reconstruction:

- *Finding the middle of the current visibility arc:* This involves moving on the tangent plane at the selected point.
- *Clockwise and counter-clockwise motion on normal plane:* This involves moving on a great circle in the direction of the surface normal at the selected point.
- *Moving on tangent plane to find new visibility arc:* This involves moving on the tangent plane at the selected point.
- *Forcing a region boundary point to become a visible rim point:* The region boundary point is indicated by a small yellow circle on the object's surface. Note that Point 2 is on the pipe's interior surface when this process begins. The yellow circle is used only to indicate its position relative to the visible rim.

[7]It can be shown that this premature termination of the viewpoint's motion does not affect the correctness of the reconstruction process.

The initial and final images in each group correspond to the views of the object before and after the step described by the group was applied, respectively.

6.8 Concluding Remarks

We have demonstrated that an active monocular observer can use a simple viewpoint control strategy to recover a global description of a generic, arbitrarily-shaped object from the occluding contour. The regions that are reconstructed on the object can be accurately characterized and depend only on global shape properties of the object.

Continuous, contour-driven viewpoint control plays a key role in the global surface reconstruction process. On one hand, it permits the use of shape-from-motion methods to extract shape information even with a single camera. On the other, its enables the use of local, contour-driven viewpoint controls that are simple enough to be executed in real time and that *guarantee* global reconstruction when they are appropriately combined. The reason is that viewpoint control is not used merely to change the shape of the occluding contour in an arbitrary manner (as in existing approaches), but it is used to change it in a well-defined way. This allows precise statements to be made about the progress of the global surface reconstruction process, and allows its outcome to be accurately controlled.

Current limitations of the method are: (1) the inability to change the camera's distance to the object, (2) the inability to reconstruct the entire surface of objects with concavities, (3) the assumption that the environment being explored contains a single object, (4) the assumption of generic objects, and (5) the lack of an analysis of how the camera's finite spatial resolution and finite temporal sampling rate affect the outcome of the reconstruction process. An extension of the method to perspective projection and to full three-dimensional motion (e.g., moving closer to the surface to see through a hole) is currently under development. Such motions will, at least in theory, eliminate all potential sources of occlusion and allow the reconstruction of all non-concave regions on an object's surface (provided the viewpoint can move arbitrarily close to the object). Furthermore, while the shape of a concave region cannot be recovered from images of the occluding contour, the contour can be used to constrain its shape [192].

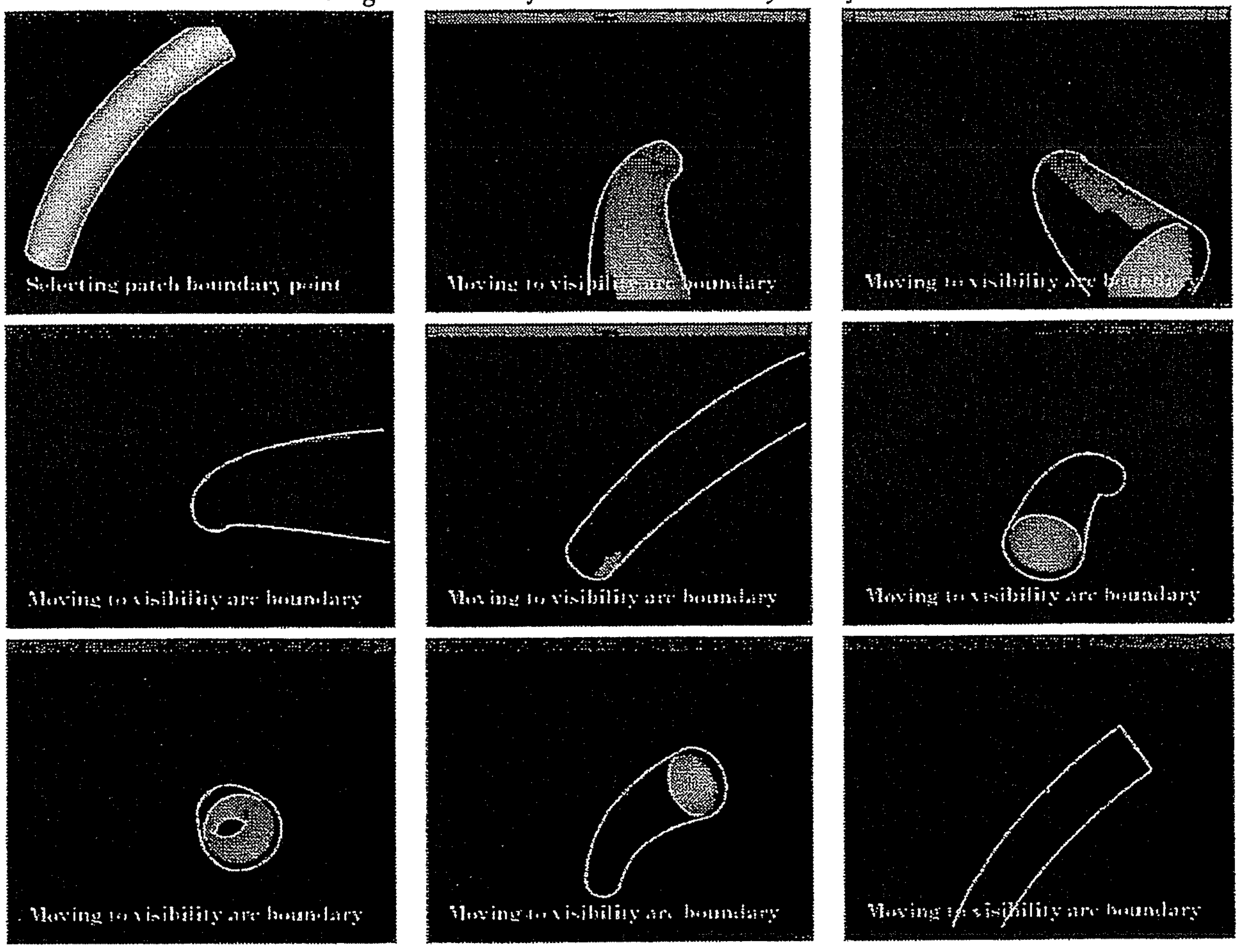
The single object assumption allows us to formulate reconstruction of an object's surface as the task of incrementally "growing" a reconstructed surface region. In practice, the object being explored may lie on a tabletop, in a robot end-effector, or in a pile of objects. In order to decide when to stop the region-growing process, the vision system must be able to determine which of the environment's visible surfaces belong to the object being reconstructed. When the object is grasped by a robot end-effector whose shape can be determined in advance, answering this question is, in principle, feasible. However, in general, when exploration is performed by moving in a static and completely unknown environment this question cannot be answered. The issue of under what conditions about the environment can the vision system decide to stop an object's exploration, requires further study.

Even though our study has focused on the exploration of generic objects, extensions are possible: The visual events defined for generic surfaces describe the visual events for the non-generic case; however, the visual event "curves" may degenerate to entire regions on an object's surface [127] (e.g., all points on a cylinder are parabolic whereas, generically, parabolic points always form curves on the surface). Reconstruction of the non-degenerate regions on the surface of non-generic objects should, in principle, be possible with the strategies developed in this chapter. The issue of how the existence of surface discontinuities affect the correctness of the developed exploration strategies also requires further investigation.

How can the strategy developed in this chapter be generalized to perform global visual search tasks (e.g., visually inspecting the surface of an object or the interior of a room, moving around an unknown three-dimensional obstacle during a motion planning operation)? What can be learned from this strategy that could be used to guide the motion of a range sensor in the construction of global object models? We believe that both of these questions can be addressed by extending the notion of the exploration frontier and studying its evolution in the context of these tasks. A key role for viewpoint control in these tasks is to force the visibility of an object's entire surface. An appropriate definition for the exploration frontier is to define it as the boundary of all surface points that were made visible from some viewpoint during the course of the exploration process. Controlling the frontier's dynamic evolution in this generalized sense requires controlling the motion of the *occlusion boundary*; this collection of curves bounds the visible from the occluded surface points at a given viewpoint, is simply a superset of the visible rim, and "slides" over the surface under continuous viewpoint control. Our analysis of the interactions between continuous, contour-driven viewpoint control, the surface's local and global geometry, and the dynamic evolution of the exploration frontier is likely to be a component in the study of these more general exploration tasks.

Region Reconstruction Process for Point 1

Finding the middle of the current visibility arc of Point 1



Moving on normal plane of Point 1

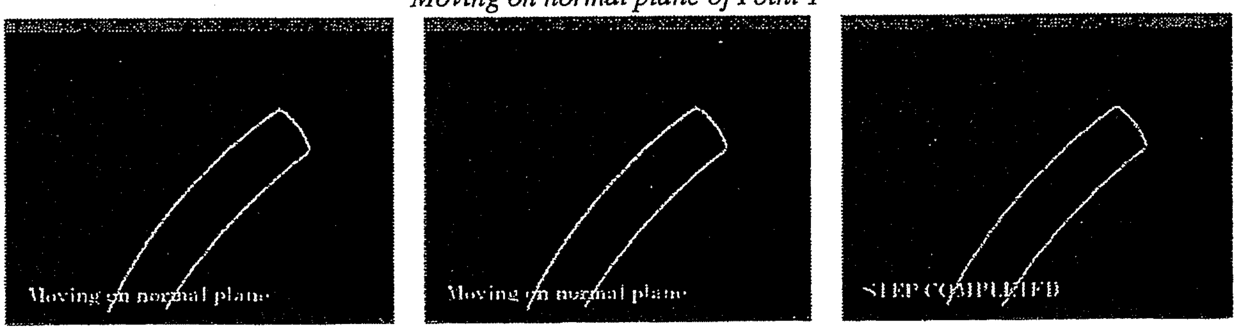


Figure 69: Reconstructing the surface of a pipe-shaped object.

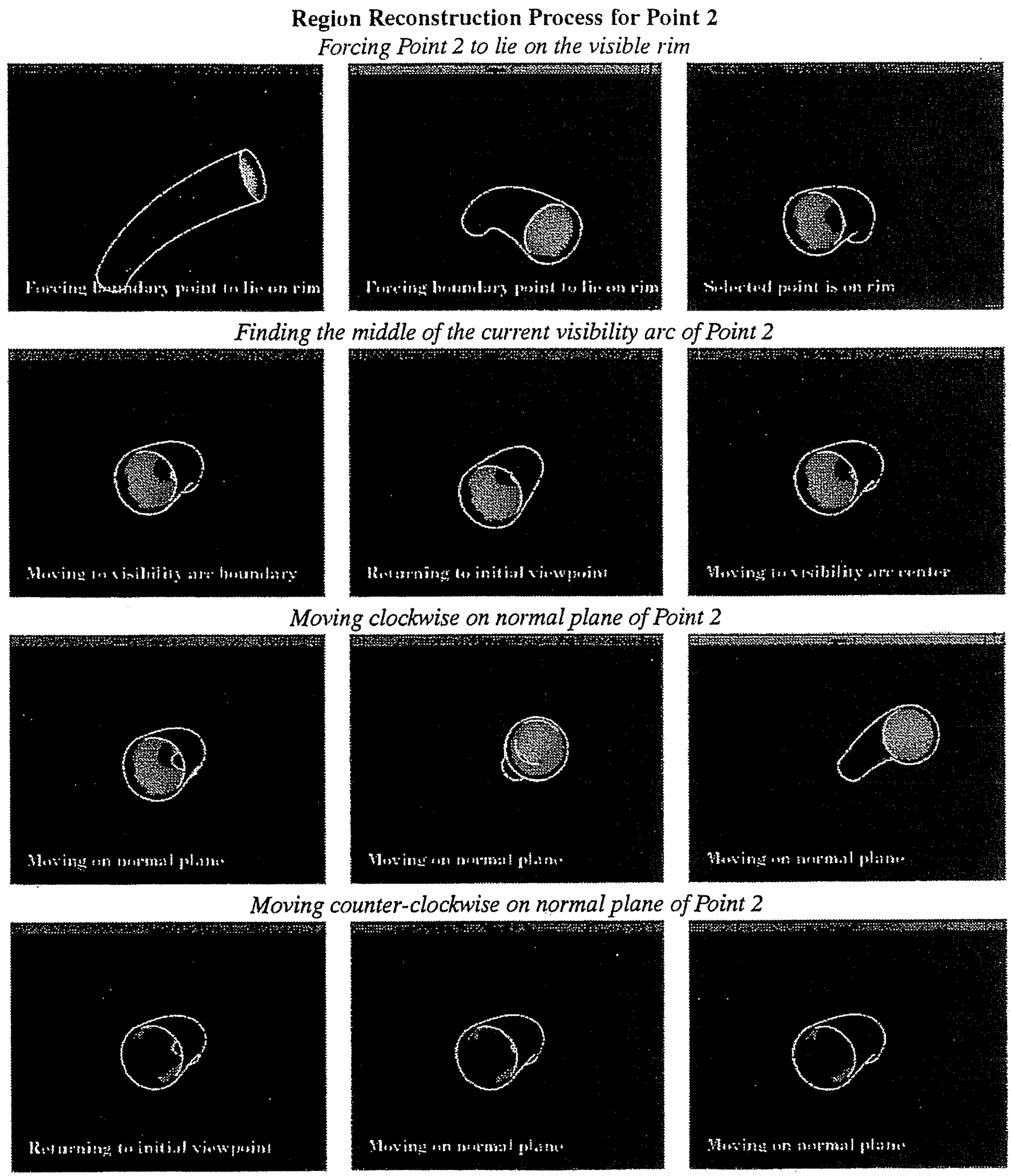
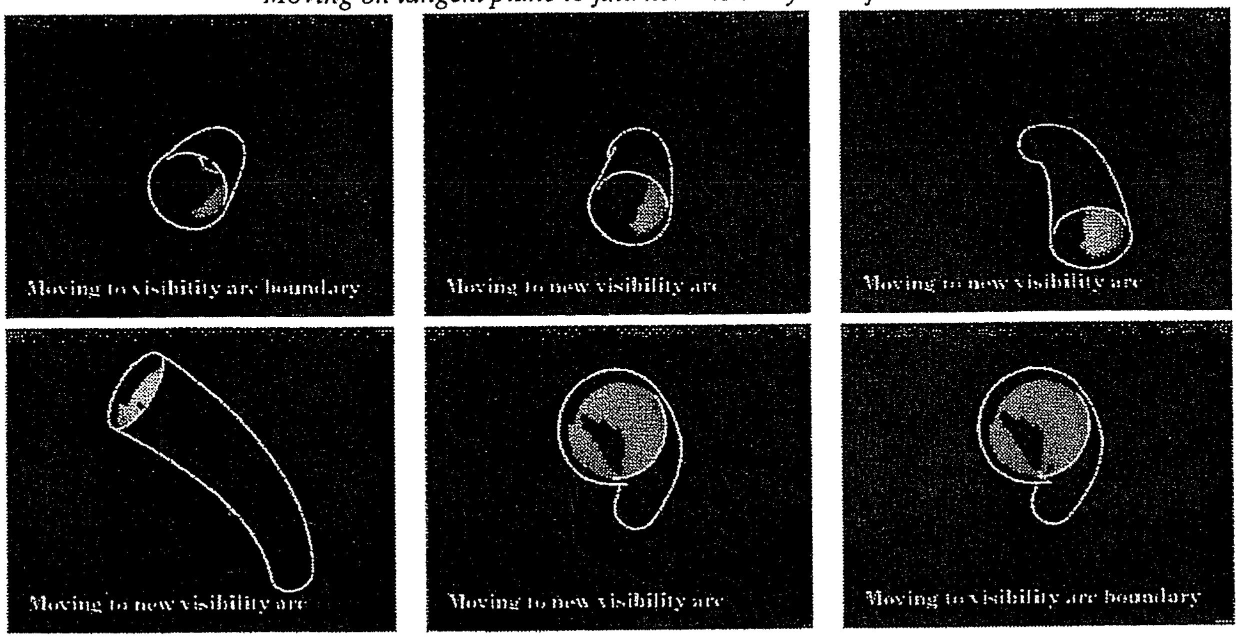


Figure 69 (cont.): Reconstructing the surface of a pipe-shaped object.

Moving on tangent plane to find new visibility arc of Point 2



Finding the middle of the new visibility arc of Point 2

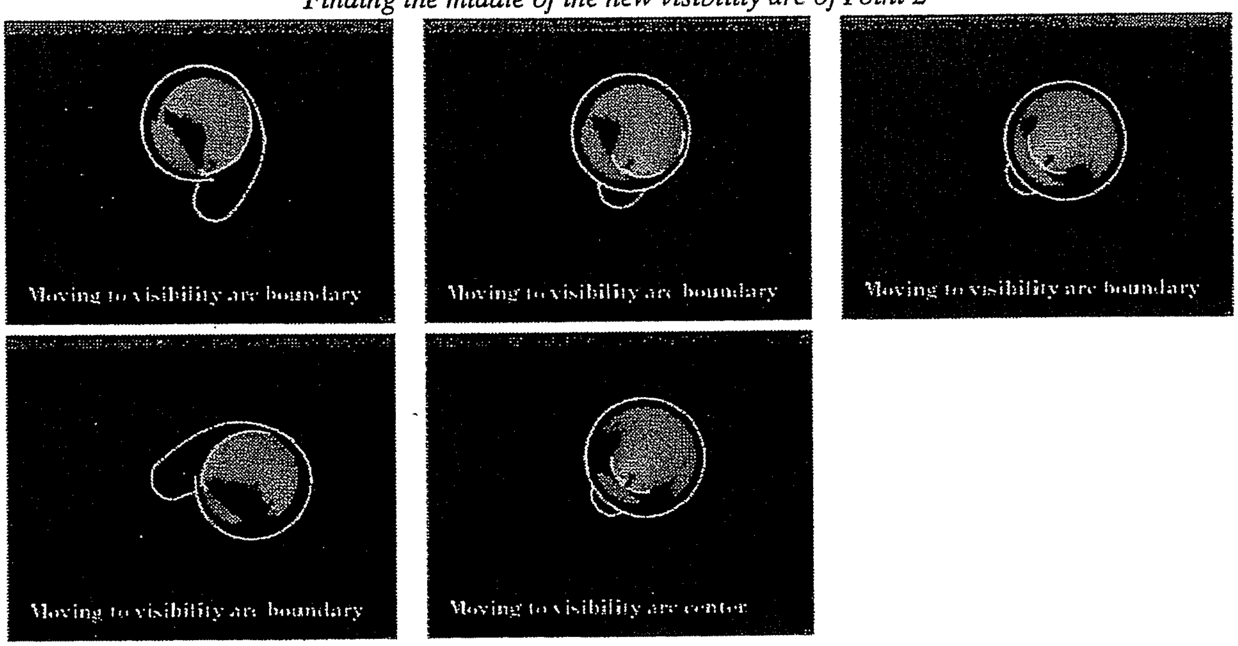


Figure 69 (cont.): Reconstructing the surface of a pipe-shaped object.

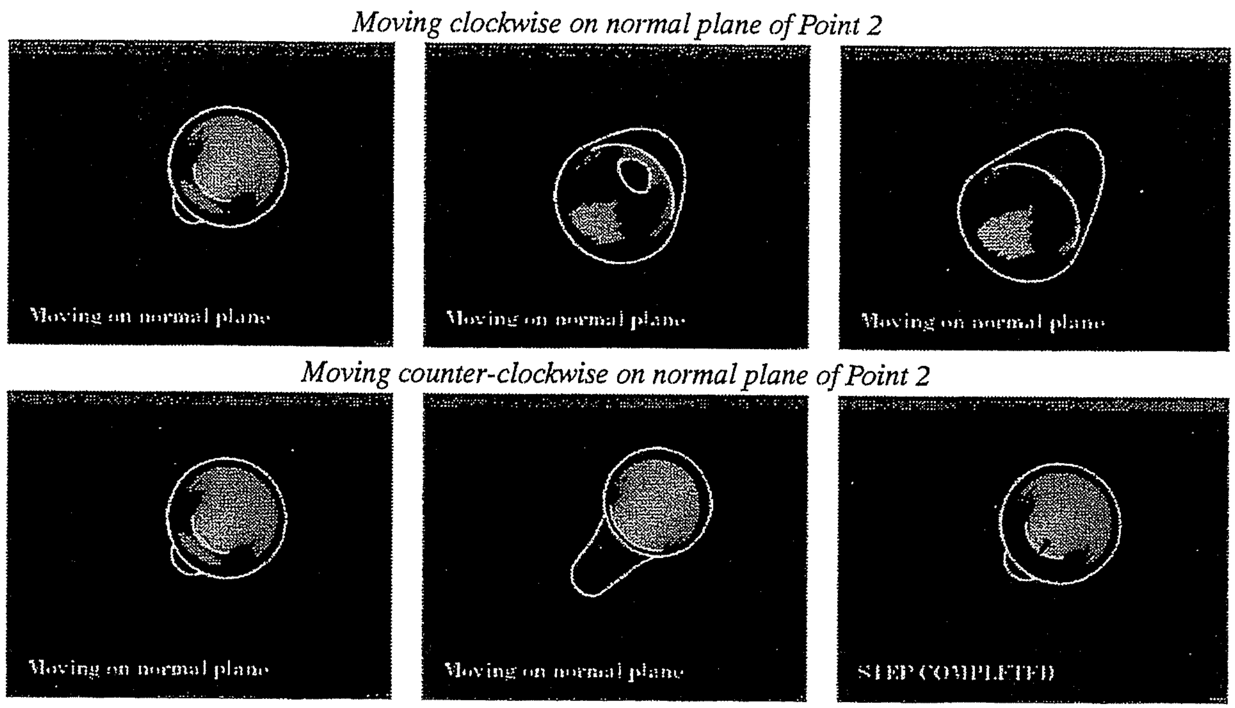
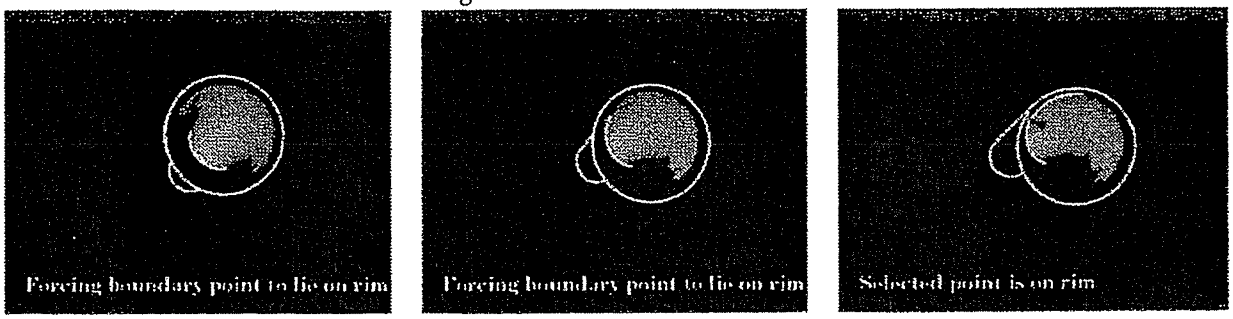


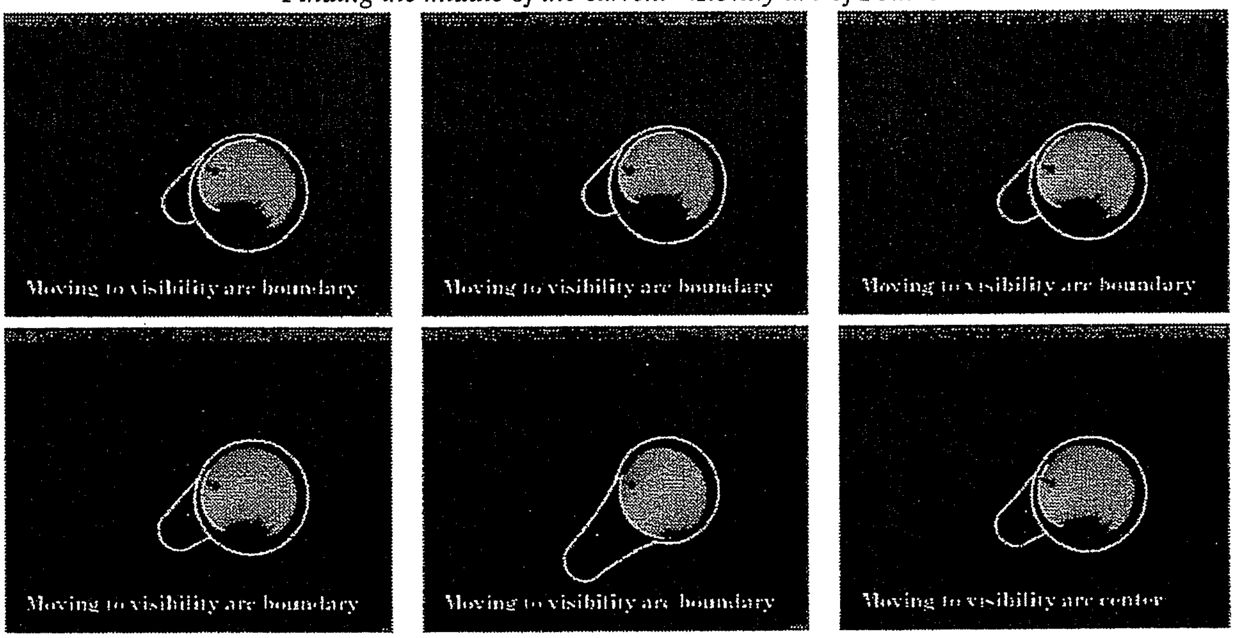
Figure 69 (cont.): Reconstructing the surface of a pipe-shaped object.

Region Reconstruction Process for Point 3

Forcing Point 3 to lie on the visible rim



Finding the middle of the current visibility arc of Point 3



Moving clockwise on normal plane of Point 3

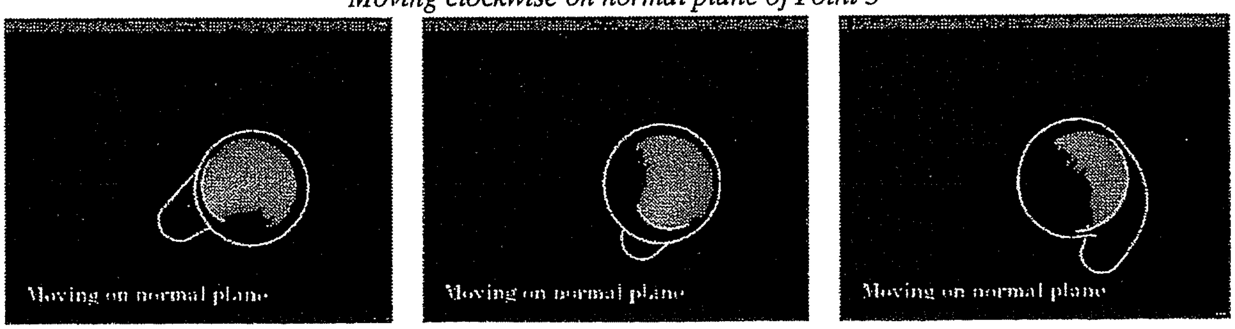
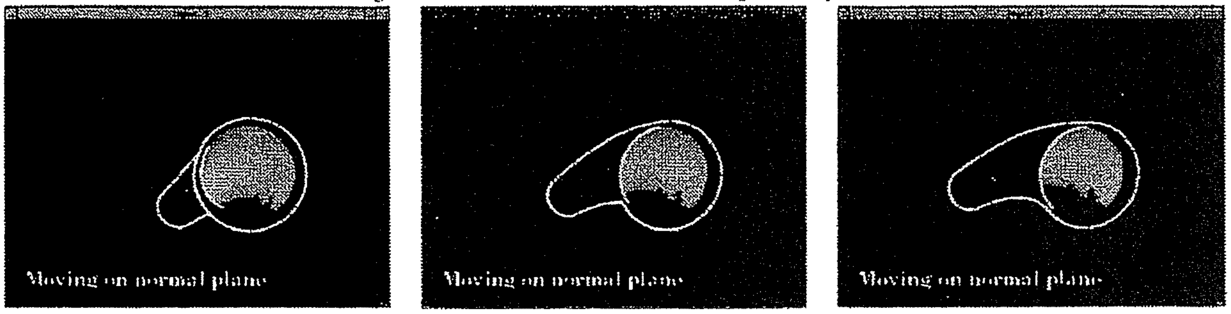


Figure 69 (cont.): Reconstructing the surface of a pipe-shaped object.

Moving counter-clockwise on normal plane of Point 3



Moving on tangent plane to find new visibility arc of Point 3

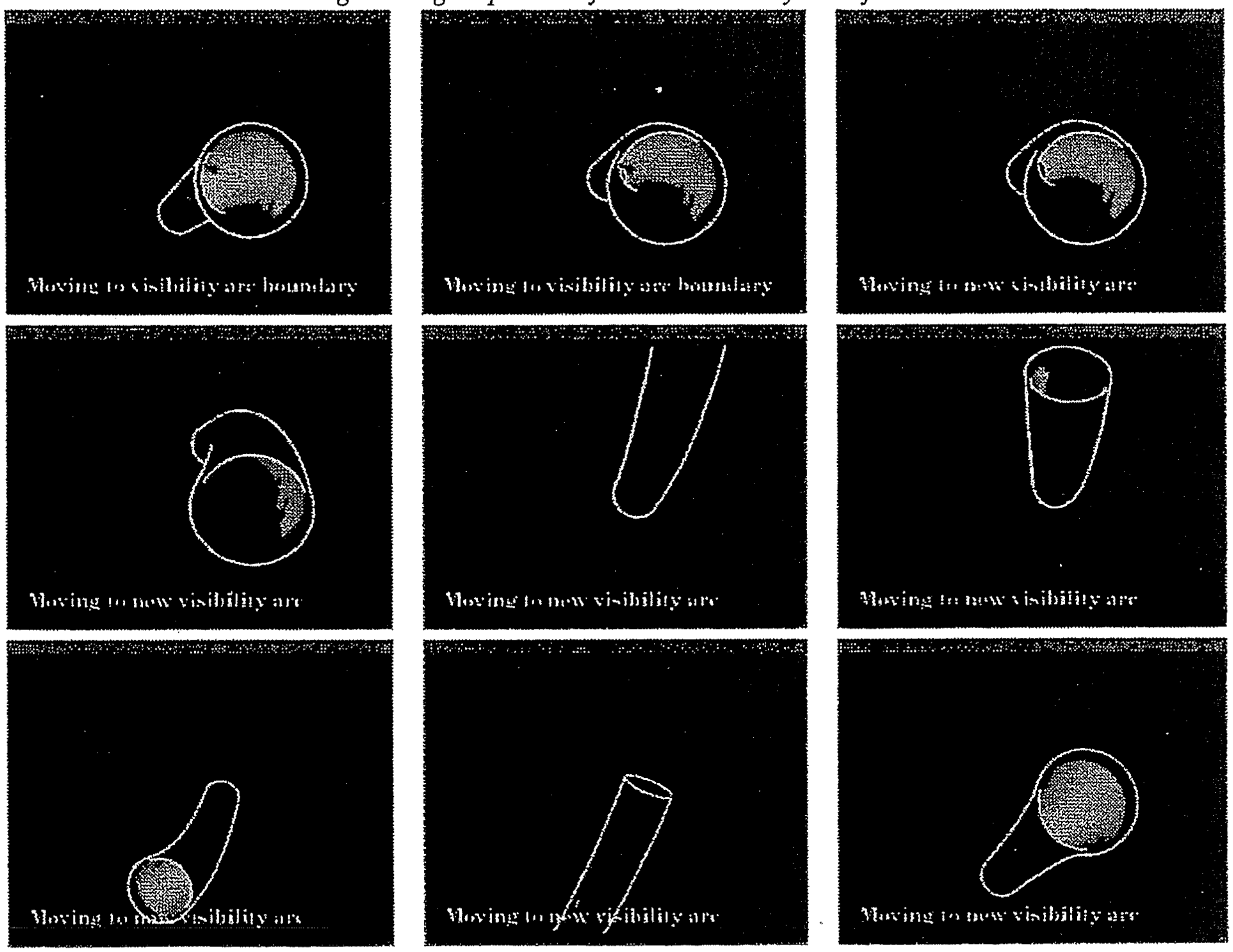


Figure 69 (cont.): Reconstructing the surface of a pipe-shaped object.

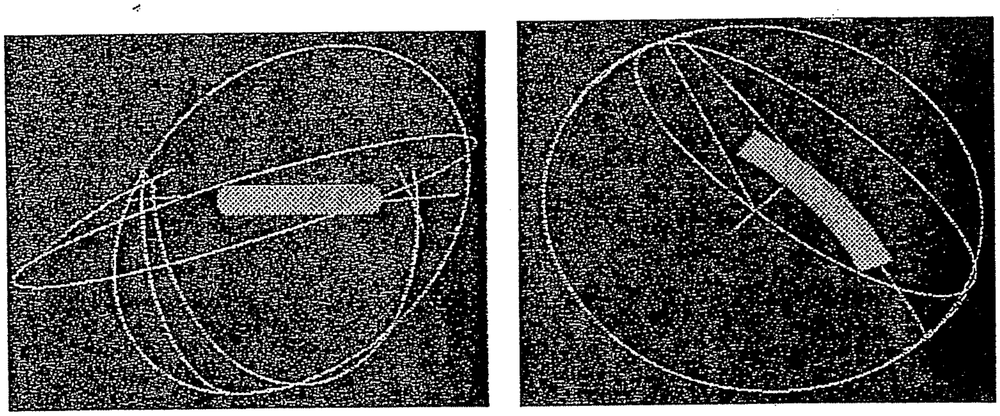


Figure 70: Two views of the path traced by the moving viewpoint. This path consists of arcs on the viewpoint's motion sphere. Also shown is the surface of the pipe.

Chapter 7

Conclusions and Future Work

This thesis has studied how controlled viewpoint movements can be used to explore an object's unknown geometry and to simplify computations required in the exploration process. Our focus has been on the extraction of qualitative, quantitative, local, as well as global information about the shape of curved objects with complex surface geometry (arbitrarily-shaped or generic). The mathematical tools we used were drawn from relationships between the geometry of smooth surfaces, their projected shape, and the way this shape changes when viewpoint changes. We obtained theoretical and algorithmic results (strategies for controlling viewpoint, geometrical analysis of their properties, and simulations) and performed initial practical demonstrations (application of our viewpoint control strategies to real scenes).

7.1 Main Contributions and Limitations

The contributions of this thesis can be summarized as follows:

- Introduction of a new framework for combining vision and action in the exploration of curved objects with complex surface geometry. In particular, this framework exploits the use of continuous, contour-driven viewpoint control both to simplify qualitative and quantitative local shape computations and to enable global shape recovery of arbitrarily-shaped generic objects.

- Formulation of viewpoint control as a process of constraining the deformation of the occluding contour and the motion of the visible rim over the surface. It is shown that this formulation allows the use of results from the local and global differential geometry of surfaces to formalize the connection between vision processing and action in the exploration process.

- Demonstration that continuous, contour-driven viewpoint control leads to local geometrical analysis, local processing in the image, and simple, locally-controlled motions even when global shape recovery tasks are performed. In particular, it is argued through a study of specific shape recovery tasks that (1) analysis of global shape recovery tasks becomes mathematically tractable, and (2) tangential viewpoint control becomes a key elementary motion in the exploration of curved objects.

- Demonstration that in the context of specific exploration tasks, continuous, contour-driven viewpoint control has certain advantages over existing approaches for exploring curved objects. In particular, by developing strategies that exploit continuous, contour-driven viewpoint control the following results were obtained:

  - Occluding contour detection can be achieved without extrinsic camera calibration and without *a priori* identification of surface markings; existing approaches employ at least one of these two assumptions.

  - Surface curvature at a point can be estimated without measurement of camera velocities or accelerations, and without *a priori* identification of surface markings; existing approaches require velocity measurements and either acceleration measurements or *a priori* identified markings.

  - Global reconstruction from images of the occluding contour can be provably achieved for generic objects of arbitrary shape; no existing approaches provide guarantees about the outcome of the global reconstruction process for curved objects that are non-convex and can self-occlude, using either the occluding contour or range data.

The results presented in this thesis have several limitations. First and foremost, we assume that viewpoint is controllable. Our framework is therefore not applicable to the analysis of image sequences that have been generated in advance, or to cases where the degrees of freedom for changing viewpoint in the manners prescribed are not available. This thesis also takes a rather extreme view on the issue of *utility* of viewpoint controls. Large (but well-quantified) viewpoint motions may potentially be generated to obtain better measurements and allow weaker calibration assumptions to be used in local shape recovery tasks. Optimality of motion path length for global shape recovery has also been ignored although, clearly, only local decisions can be made. In time-critical situations, large motions will eventually limit the performance of the local shape recovery strategies we described, and the development of locally-optimal strategies for global exploration will become necessary. While the occluding contour is an invaluable source of shape information, it is by no means the only shape cue potentially available in an image. The interaction between various shape cues (e.g., contours, texture, discontinuities, specularities) when viewpoint changes and between different sensors (e.g., vision and range) is only now beginning to be understood. A deeper understanding of such interactions and of the rôle of controlled viewpoint movements in this broader context will inevitably become important in the exploration of objects from diverse real-world environments.

7.2 Future Directions

The work presented in this thesis can be extended along several directions. These include addressing theoretical and practical issues related to system-building and application of the contour-driven viewpoint control framework to novel tasks:

- **Occluding contour tracking:** The ability to track occluding contour curves across frames is key to the exploration processes we described. While fast and robust curve tracking methods have been reported [28], the contour's shape-dependent deformations and connectivity changes cannot be handled by current trackers. The viewpoint control strategy of Chapter 4 allows surface shape information in the form of parallax measurements [44, 109] to be incorporated in the tracking process to predict such curve deformations and connectivity changes.

- **Implementation of viewpoint controls:** We plan to implement the developed viewpoint control strategies by mounting either a camera or an object on a six-degree-of-freedom robotic manipulator. This requires further investigation of effects of singularities and joint limits on planar viewpoint control, issues in real-time manipulator control, as well as how hand-eye calibration requirements can be relaxed.

- **Constructing Euclidean object models for reverse engineering:** The integration of the developed viewpoint control strategies, their extension to non-generic object classes, as well as the incorporation of existing shape-from-contour techniques in the exploration process for producing CAD models of objects is an important direction for future research. We are also interested in investigating how *a priori* information about an object's surface, e.g., information about the primitive shapes contained in a Constructive Solid Geometry representation of the object, can aid the exploration process.

- **Constructing affine object models using an uncalibrated camera:** A key problem in the construction of Euclidean curved surface models is the need for camera calibration. Several recent approaches have shown how affine representations of point sets can be constructed by a moving uncalibrated camera; these representations can replace Euclidean representations both in recognition as well as computer graphics tasks (e.g., real-time animation). Unfortunately, no such methods exist for obtaining affine models of curved objects. We are in the process of extending the occluding contour detection strategy of Chapter 4 to construct affine representations of curved objects using an uncalibrated camera.

- **Performing global qualitative exploration tasks:** In principle, the inspection of an unknown object's surface (e.g., for finding the bar code identifying the object) should not require reconstruction of the object's surface or precise camera calibration. We plan to extend our analysis of structuring the motion of the exploration frontier to deal with such global qualitative exploration tasks. A key issue in this context is how to represent the exploration frontier when the camera is uncalibrated, e.g., by constructing an affine representation of the frontier.

- **Active object recognition:** Active object recognition is a task that lends itself both to the study of motion utility and to the study of special viewpoints. The underlying issue here is efficiency: Viewpoint should be changed only when such a change will lead to faster recognition. On one hand, this requires deriving an estimate of object hypotheses that would be generated if a single image was used for recognition. On the other hand, it requires using contour-driven viewpoint control to reach special viewpoints that (1) are closely related to the object's global geometry (e.g., views along an object's supporting plane), and (2) are guaranteed to reduce the number of generated object hypotheses (e.g., by reducing the dimensionality of the hypothesis space).

- **Motion planning and exploration in three-dimensional environments:**

  The problem of provably-correct motion planning in an unknown and unstructured 3D environment for a freely-moving robot is still an open problem. In previous work [101] we have shown that this problem requires, in general, exploration of an environment's obstacles. To perform such an exploration process the ability to control distance to the obstacle surfaces must be taken into account. While the strategy developed for global surface reconstruction can be extended to take distance changes into account, several questions still remain. These include dealing with the inability to reconstruct object concavities, the inability to move arbitrarily close to an obstacle, and the extension of the developed strategies to non-generic objects.

Appendix A

Proofs of Chapter 4 Theorems

A.1 Proof of Theorem 4.1

(1) Since $p(0)$ is visible along $\xi(0)$, the open line segment $p(0)q(0)$ does not intersect the interior of the surface. Furthermore, if $p(0)$ is non-stationary, E is not tangent to S at $p(0)$. In this case, $S \cap E$ is a smooth curve that is convex in the neighborhood of $p(0)$. To establish the epipolar plane correspondences for $p(t)$, all points $p(\tau), 0 \leq \tau \leq t$, must be convex, and no point $p(\tau)$ can be occluded by a distant surface point. Hence, the set $\{p(\tau)|\tau \in [0,t]\}$ is connected.

(2) If $p(0)$ is stationary, $p(t) = p(0)$, and hence their projection coincides. If $p(0)$ is non-stationary, their image difference is equal to the projection of $p(t) - p(0)$ on the image plane at $\xi(t)$. Since $p(t)$ is on the occluding contour, $\xi(t)$ is perpendicular to $n(p(t))$. The image plane is also perpendicular to $\xi(t)$, and hence it is parallel to $n(p(t))$. Eq. 4 now follows.

(3) From Eq. 4 it follows that $\|q(t) - \hat{q}(t)\| = 0$ if and only if $[p(t) - p(0)] \cdot n(p(t)) = 0$. When $p(t) \neq p(0)$, the distance becomes zero only when the segment $p(t)p(0)$ is tangent to λ at $p(t)$. Let $p(t_1)$ be the first point whose tangent is on $p(t_1)p(0)$ when λ is traversed clockwise starting from $p(0)$. Since λ is convex, for any point $p(t')$ on λ after $p(t_1)$ the line through $p(t')$ and $p(0)$ intersects S on both sides of $p(t')$. Hence, if λ contains a point $p(t'')$ after $p(t_1)$ whose tangent is on $p(t'')p(0)$, that point will never be visible. This implies that $p(t_1)$ is the *only* point in a clockwise traversal of λ for which $q(t_1)$ and $\hat{q}(t_1)$ coincide. If λ is open the above proof must be repeated for a counter-clockwise traversal of λ. Therefore, there are at most three points on λ for which $q(t)$ and $\hat{q}(t)$ coincide and three

viewing directions along which this can occur, namely the tangents at those points.□

A.2 Proof of Corollary 4.1

It follows that the points of λ between $p(0)$ and $p(t)$ coincide with their convex hull. Therefore, the line segment $p(t)p(0)$ intersects the interior of S only in the neighborhood of $p(t)$, and its projection in the image coincides with the projection of the visible surface points on λ that are close to $p(t)$.□

A.3 Proof of Corollary 4.2

(1) Parameterize λ by arc length and define $s(t)$ so that $p(t) = \lambda(s(t))$ and $s(0) = 0$. The extremum condition follows directly by differentiating Eq. 4. The details of this derivation are omitted. To obtain a lower bound on the extrema of $f(t^*)$, let λ' be the segment of λ between $p(0)$ and a distance maximum, let κ be the maximum absolute curvature of λ', let $\tilde{p}$ be the point on λ where this occurs, and let $\mathbf{t}$ be its tangent. Consider the smallest disc that is centered at $p(0)$ and contains λ' in its interior. The disk will touch λ' at a point $p(t^*)$ for which f is maximized, and will have radius $R = f(t^*)$. It follows that $f(t^*) \geq 1/\kappa$.

To obtain a bound on κ, we use Meunsier's formula [38]. According to the formula, $\kappa = \kappa_n / \sin(\phi)$, where κ_n is the normal curvature at $\tilde{p}$ along $\mathbf{t}$, and ϕ is the angle between E's normal and the surface normal at $\tilde{p}$. It follows that $\kappa \leq \kappa_{max}/\sin(\phi)$. Eq. 5 follows by combining the bounds for κ and R.

(2) *(Sketch)* Since λ is convex and no distance maximum exists for p between p_1 and p_2, no point other than p_{mid} is contained in the intersection $P_1 \cap P_2$ (Figure 4.1(g)).□

A.4 Proof sketch of Theorem 4.2

(1) If $q(t_1)$ is stationary, $q(t_2)$ and $q(t)$ are projections of the same surface point. Properties 4.1 and 4.2 imply that $\tilde{p} = p(t_1)$ and $\tilde{q}(t) = \hat{q}(t)$. When $q(t_1)$ is non-stationary, the proof is identical to that given in Theorem 4.1. (2) In general, $\tilde{p}$ does not lie on λ. If it does, the proof is identical to that in Theorem 4.1. If it lies in the surface interior or exterior, the proof of Theorem 4.1 can be modified to show that

at most two and four viewing directions, respectively, make $\|q(t) - \tilde{q}(t)\|$ zero.$\square$

Appendix B

Proofs of Chapter 5 Theorems

B.1 Proof of Corollaries 5.1-5.3

If ξ is along e_1, $\phi = 0$ in Eq. (14). Corollary 5.1 immediately follows. For Corollary 5.2, note that k_{n_1} is derived using Corollary 5.1, and that ϕ is known. Since $K \neq 0$, Eq. (14) is well-defined and we can use it with k_{n_1} as the unknown. The other principal direction is also computable since e_2 must lie on $T_p(S)$ and be perpendicular to ξ. In fact, ξ is perpendicular to the rim even though this does not hold in general [90].

Finally, the derivative of $k_o(\phi)$ is

$$k_o'(\phi) = \left(\frac{1}{k_{n_2}} - \frac{1}{k_{n_1}}\right) \frac{\sin 2\phi}{\left(k_{n_1}^{-1} \sin^2 \phi + k_{n_2}^{-1} \cos^2 \phi\right)^2} \quad (15)$$

In the case of an elliptic, non-umbilic point, $k_{n_1} \neq k_{n_2}$ and therefore k_o' becomes 0 for $\phi = 0$ or $\phi = \pi/2$, i.e., when ξ is along a principal direction. If p is hyperbolic, the expression in the denominator tends to 0 as ϕ approaches $\arctan \sqrt{(k_{n_1}/ - k_{n_2})}$ which is the angle between e_1 and the asymptote of the surface at p. In the interval $[\arctan \sqrt{(k_{n_1}/ - k_{n_2})},\ \arctan \sqrt{(k_{n_1}/ - k_{n_2})} + \pi]$ p becomes occluded and therefore $k_o(\phi)$ is undefined. In the interval where $k_o(\phi)$ is defined, Eq. (15) shows that $k_o(\phi)$ has a maximum only for $\phi = 0$ or $\phi = \pi$. Finally, when p is umbilic, $k_{n_1} = k_{n_2}$ by definition, and $k_o'(\phi)$ is identically zero. □

B.2 Proof of Proposition 5.1

(1) (*Only If*) Consider the intersection of S with $T_p(S)$. If the intersection contains only the point p, then the intersection of any line $m \in T_p(S)$ with S will either be empty or equal p. Recall that while changing viewing direction, $T_p(S)$ is viewed "edge-on" and its projection is the line l. Since l is the projection of all lines in $T_p(S)$ (except those lines parallel to ξ), it follows that l will only intersect the occluding contour at q.

(*If*) Assume there is a viewing direction in $T_p(S)$ from which p is not visible. Let ξ' be the first such direction while moving clockwise (Figure 71). The viewing direction ξ' must contact S at p and at at least one more point, say s. Now consider the intersection of S with $T_p(S)$. The intersection will consist of a set of closed curves and isolated points. Since p is elliptic there must exist a small disk in $T_p(S)$ centered at p that does not contain any other points of S. Therefore p and s must be in different components. We distinguish two cases, namely whether s is an isolated point or a point on a curve. If s is an isolated point then $T_p(S)$ must be tangent to S at s. But then s is also part of the rim when viewed along the original direction ξ. In addition, s must be on the visible rim because it is the first point that occludes p when changing viewing direction from ξ to ξ'. This implies that the projection of an imaginary line joining s and p will contact the occluding contour at two points, the projections of s and p.

If s is not an isolated point, let $Q \subseteq S \cap T_p(S)$ be the closed curve containing s. Now consider the family of lines parallel to ξ. A line of the family will contact Q, say at point r. Without loss of generality assume that r is the first such contact point when Q is traced in a counterclockwise fashion starting from s. This point, by definition, must be on the rim of S when the viewing direction is ξ. If it is also on the visible rim (Figure 71(b)), the projection of the imaginary line joining r and p must intersect the occluding contour at at least two points (i.e., at the projections of r and p).

Now suppose r is not on the visible rim. To treat this case, note that by definition, s must be visible when the viewing direction is ξ. Let s' be the first occluded point on Q when Q is traced in a counterclockwise fashion starting from s (Figure 71(c)). The point r' occluding s' must necessarily belong to the visible rim. Therefore, the projection of the imaginary line connecting r' and p intersects the occluding contour at at least two points.

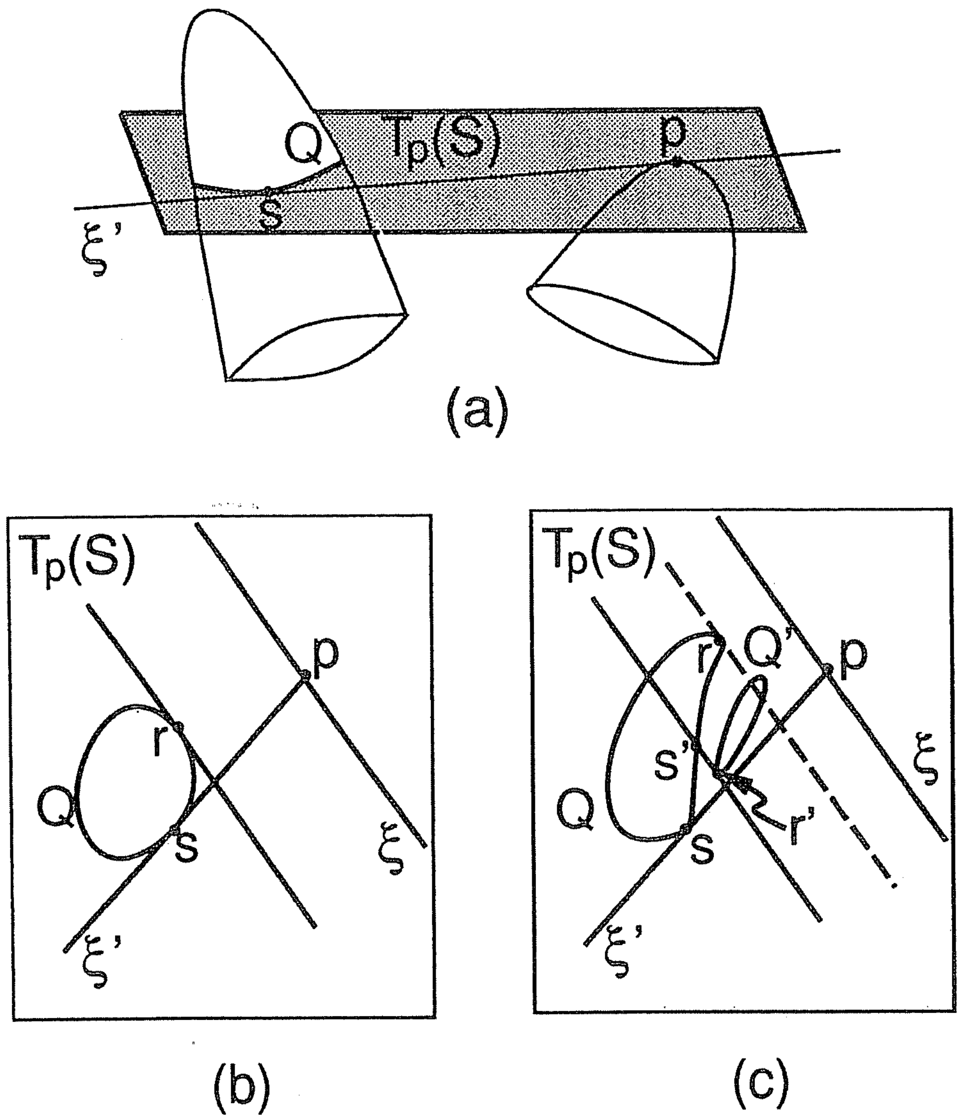


Figure 71: The effects of global occlusion. (a) A "side" view of $T_p(S)$. Viewing direction ξ' is the first direction in which p becomes occluded. s is the point occluding p from that direction. Q is the component of $S \cap T_p(S)$ containing s. (b),(c) "Top" views of the tangent plane of S at p. v is the original viewing direction.

(2) Consider any point q on C that is also contained in the convex hull of C. Since C cannot be a straight line, q is, by definition, the only point in common between C and the tangent at q. But then q also satisfies the conditions of (1) above. □

Appendix C

Extent of Viewing Direction Adjustments for Local Shape Recovery

Let us assume we have recovered the principal curvatures at p and the viewing direction ξ is along the principal direction e_2 at p. Now assume that the viewing direction changes on the plane of ξ and the surface normal at p in order to introduce new points to the rim. We show that the viewing direction adjustment that will be needed during the shape recovery step is proportional (at a first approximation) to k_{g_2} and inversely proportional to k_{n_2}, the geodesic and normal curvatures of the line of curvature corresponding to e_2. This is an important result because it allows us to predict the performance of this active viewing strategy based on intrinsic properties of the viewed surface. It follows that the performance of our strategy smoothly degrades as the surface becomes more complicated (i.e., k'_{g_2} and k'_{n_2} become large). We first present some concepts from differential geometry for the study of curves on surfaces.

C.1 The Local Geometry of Surface Curves

Let $\alpha(s) : I \to \Re^3$ be a curve parameterized by arc length (i.e., $|\alpha'(s)| = 1$). Consider the unit tangent and unit normal vector, $t(s)$ and $n(s)$ respectively, at point $\alpha(s)$. We can describe the curve with two quantities, its curvature $\kappa(s)$ and torsion $\tau(s)$, where $\alpha''(s) = \kappa(s)n(s)$ and $(t(s) \wedge n(s))' = \tau(s)n(s)$. The vectors $t(s), n(s), t(s) \wedge n(s)$ describe an orthogonal coordinate frame, the *Frenet frame* centered

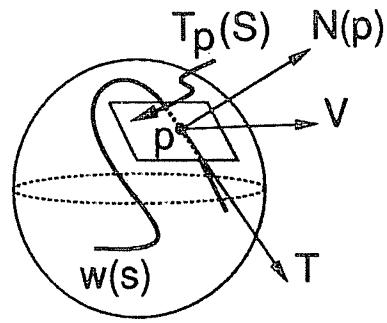


Figure 72: The Darboux Frame. $w(s)$ lies on a sphere S, and p is a point of $w(s)$. $N(p)$ is the surface normal, $T = w'(s)$, and $V = N(p) \wedge T \in T_p(S)$. The Darboux frame is the orthonormal coordinate frame of T, V, N.

at $\alpha(s)$. This coordinate frame can be used to locally describe the curve based on the values of κ and τ at $\alpha(s)$.

Now let S be a smooth, oriented surface, and let $\bar{\alpha}(s)$ be a smooth curve on S. We can locally describe $\bar{\alpha}(s)$ using a coordinate frame similar to the Frenet frame called the *Darboux frame* (Figure 72). Consider a point p on $\bar{\alpha}(s)$. The Darboux frame is defined by $N(p)$, the normal to the surface, $T(p)$, the tangent to $\bar{\alpha}(s)$, and $V(p) = N(p) \wedge T(p)$. Note that the $T - V$ plane is the plane tangent to S. The vector $\bar{\alpha}''(s)$ defining the curvature of $\bar{\alpha}(s)$ can be analyzed in terms of two components, a tangential component (i.e., on $T_p(S)$) in the direction of V, and a normal component in the direction of N. Therefore we can define the curvature of $\bar{\alpha}(s)$ in terms of the curvatures of its projections k_g, k_n on the tangent plane of S and on the $T - N$ plane, respectively. k_g is called the *geodesic curvature* of $\bar{\alpha}(s)$ and k_n is the curvature of the normal section of S in the direction of T. Intuitively, the geodesic curvature measures how "far off" the $T - N$ plane the curve actually lies. We show that the geodesic curvature of the lines of curvature is closely related to the strategy employed to select new points for shape recovery. Intuitively, the geodesic curvature of the lines of curvature measures how the arc length of a curve in the e_1 direction changes as one moves along the e_2 direction.

The curve $\bar{\alpha}(s)$ can be locally described by the vectors T, N, V and their derivatives. These derivatives can also be expressed in terms of the three frame vectors:

$$\frac{dT}{ds} = k_g V + k_n N \tag{16}$$

$$\frac{dV}{ds} = -k_g T - \tau_g N \tag{17}$$

$$\frac{dN}{ds} = -k_n T + \tau_g V \tag{18}$$

where τ_g is called the *geodesic torsion* of $\bar{\alpha}$.

C.2 The Dependence of the Viewing Direction Adjustments on k_{g_2}

Intuitively, the dependence on k_{g_2} is not unexpected: Recall that k_{g_2} measures how far off the plane of ξ and $N(p)$ the line of curvature actually lies. On the other hand, the curve $\beta(s)$ traced by the visual ray that originally passed through p lies on that plane (Figure 73). Therefore, one should expect a connection between the angle of $\beta'(s)$ and e_2 k_{g_2}. The following result shows that there is a very simple relation between them:

Proposition C.1 *(1) Let $\beta(s)$ be the intersection of S with the plane defined by ξ and $N(p)$ ($\beta(0) = p$, $\beta'(0) = \xi$). If ξ is along the principal direction e_2 then*

$$k_{g_2} = \frac{d\phi}{ds} \tag{19}$$

where k_{g_2} is the geodesic curvature of the line of curvature along e_2 at point p, and $\phi(s)$ is the angle between $\beta'(s)$ and the second principal direction at $\beta(s)$.

(2) Let $\xi' = \beta'(s)$ be the new viewing direction on the plane of ξ and $N(p)$. If θ is the angle between ξ and ξ', then for values of θ close to 0 we have

$$\phi(\theta) \approx \frac{k_{g_2}}{k_{n_2}} \sin\theta \tag{20}$$

Proof: (1) The Darboux trihedron for $\beta(0)$ is composed of the vectors $T(0) = \beta'(0)$, $N(\beta(0)) = N(p)$, and $V(0) = N(\beta(0)) \wedge T(0)$, where $N(\cdot)$ is the Gauss map for the surface. We use a second-order Taylor series expansion and Eqs. (16)-(18) to find $T(s) = \beta'(s)$ with respect to $T(0) = \xi$:

$$T(s) - T(0) \approx s\frac{dT}{ds} + \frac{s^2}{2}\frac{d^2T}{dT^2} \tag{21}$$

$$\approx s(k_gV + k_nN) + \frac{s^2}{2}(k_gV + k_nN)' \tag{22}$$

$$\begin{aligned} &\approx -\frac{s^2}{2}\left(k_g^2 + k_n^2\right)T + \\ &\cdot\left[sk_g + \frac{s^2}{2}\left(k_g' + k_n\tau_g\right)\right]V + \\ &\left[sk_n + \frac{s^2}{2}\left(k_n' - k_g\tau_g\right)\right]N \end{aligned} \tag{23}$$

where all coefficients of s are evaluated at $\beta(0)$. Now note that $\beta(s)$ is always on the $T-N$ plane and therefore $T(s)\cdot V(0) = T(0)\cdot V(0) = 0$, or $(T(s) - T(0))\cdot V(0) = 0$ for all s. Constraining the V-component of Eq. (23) to be identically equal to zero we get

$$k_g = 0 \tag{24}$$

The geodesic curvature k_g can be expressed in terms of the geodesic curvatures of the lines of curvature using Liouville's formula:

$$k_g = k_{g_1} cos\psi + k_{g_2} sin\psi + \frac{d\psi}{ds} \tag{25}$$

where ψ is the angle between $\beta'(0)$ and e_1. But $\beta'(0)$ is equal to e_2, and therefore $\psi = \pi/2$. Noting that $\phi = \pi/2 - \psi$ and combining Eqs. (24) and (25), we get the desired result. □

(2) ξ' will be tangent to $\beta(s)$ for some s. Therefore, $\xi' = T(s)$. We use Eq. (23) to get a first order approximation of s for values close to 0:

$$s \approx \frac{[T(s) - T(0)]\cdot N}{k_{n_2}} \tag{26}$$

Note that $[T(s) - T(0)]\cdot N$ equals $\sin\theta$, where θ is defined as above. Now using Eq. (19) and a first-order approximation for $\phi(\theta)$ we get the desired result. □

Finally, we can draw three conclusions from Eq. (20):

- If $k_{g_2} = 0$, no viewing direction adjustments will be done during the shape recovery phase if the viewing direction changes in the plane of ξ and $N(p)$ in the second step. The curve $\beta(s)$ traces a part of the line of curvature associated with e_2. This can happen only if that line of curvature is also a geodesic.

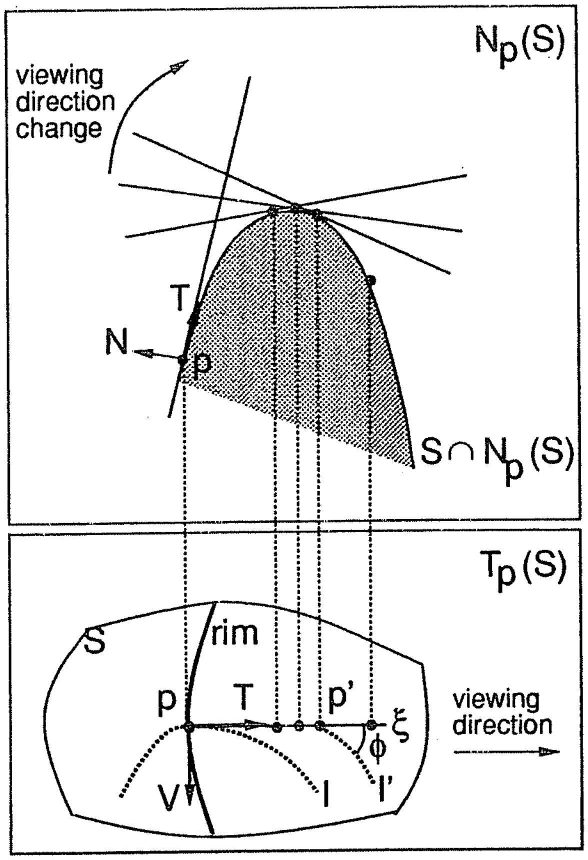


Figure 73: Changing directions on the T-N plane. Top: $N_p(S)$ is the T-N plane. The visual ray initially grazes the surface at p in the direction T. N is the surface normal at p. As the viewing directions change on this plane, the visual ray traces the curve $\beta(s) = S \cap N_p(S)$. p' is the new point selected for shape recovery. Bottom: A view of the tangent plane at p. The plane $N_p(S)$ and the traced curve are viewed "edge-on". The viewing direction change stops when the visual ray grazes p'. l and l' are the lines of minimum curvature passing through p and p', respectively. The shape recovery step will require a rotation by an angle of ϕ on the tangent plane at p' in order to align the viewing direction ξ with e_2 at p'.

- $\phi(\theta)$ can grow arbitrarily large with decreasing values of k_{n_2}. If k_{n_2} is close to 0, the surface is locally flat in the e_2 direction. Therefore, in such a case the approximation is not valid. However, this problem is inherent to the use of the occluding contour for shape recovery in the case of almost flat surfaces. The reason is that if the surfaces are locally flat, surface points will enter and leave the rim at arbitrarily large rates. This problem will also exist for methods that measure image velocities in the vicinity of the rim (e.g. [44]), since they require that the image points or features are not widely separated on the surface.

- Eq. (20) can also be used as a means to approximate k_{g_2}: After a small rotation by θ in the plane of $N(p)$ and ξ, the shape recovery step will produce a value for $\phi(\theta)$. Hence, we can use the equation to approximate k_{g_2}. This means that we will be able to completely describe the line of curvature corresponding to e_2 in the vicinity of the previously selected point.

Appendix D

Visual Events and their Associated Visual Event Curves

In this Appendix we briefly describe the visual events relevant to the analysis in Chapter 6, as well as their associated visual event curves. See [127] for an excellent intuitive description of the visual events, their associated surface curves, and their relation to the shape of the object. More details can also be found in [85, 90, 129, 130, 139, 140].

Under orthographic projection, the cells for which the occluding contour's topology is constant occupy two-dimensional regions on the viewing sphere. Visual events, i.e., the boundaries of these cells, correspond to viewpoints where the projection mapping is singular. For generic objects, visual events are defined either by curves on the viewing sphere or by the intersection of two such curves [89, 127]. Since we only use results from the analysis of visual events defined by curves on the viewing sphere, we restrict our discussion below to those events.

Each visual event can be defined as the intersection of the viewing sphere with a special *ruled* surface, i.e., a surface that can be represented as a 1-parameter family of lines called *rulings*. The rulings are lines that contact the object at multiple points, or have a high-order contact with the object at one or more points.[1] These contact points form certain characteristic curves associated with the visual event, called *visual event curves*.

[1] A line is said to have n-th order contact with a surface at a point p when all directional derivatives at p along the line up to (but not including) order n are zero.

Visual events are classified into *local* and *multilocal* events, depending on whether the lines of their associated ruling touch the surface at one or more points. *Local events* occur when the lines have a high order contact with the surface at exactly one point. In this case the viewpoint is contained in a ruled surface that touches the surface along a single curve defining the visual event. *Multilocal events* occur when the lines of the ruling contact the surface at at least two points. The viewpoint in this case belongs to a ruled surface that touches the object along two or more curves. Figure 74 shows the topological changes corresponding to all possible local and multilocal events.

The local events are the *swallowtail*, *beak-to-beak* and *lip* events (Figure 74(a)). These events correspond to viewing directions where there is a visible rim point p for which the line l defined by p and the viewing direction has fourth order contact with the surface. The swallowtail event occurs when p is on a *flecnodal curve* [89], while the lip and the beak-to-beak events occur when p is on a parabolic curve.

The multilocal events are the *triple-point*, *tangent-crossing*, and *cusp-crossing* events [127] (Figure 74(b)). Triple-point events occur at viewpoints where there is a triplet of collinear rim points whose supporting line l is parallel to the viewing direction. These three points project to a single point on the occluding contour. The line l has second order contact with each of the points. The ruled surface associated with a triple-point event contains rulings that touch the object at three distinct points; it is formed by sweeping l while maintaining three-point contact with the object. These points trace three curves on the surface; we define $\tau(s)$ to be the curve containing the point farthest away from the viewpoint's position.

A tangent-crossing event occurs at viewpoints where there is a pair of rim points with a common tangent plane parallel to the viewing direction. The ruled surface associated with this event touches the surface along two curves; we define $\gamma(s)$ to be the curve containing the point farthest away from the viewpoint's position. Finally, a cusp-crossing event occurs at viewpoints for which there is a line l parallel to the viewing direction that touches the surface at two visible rim points, one of which is a cusp visible rim point. In this case, l has third order contact with the surface at the cusp point and second order contact at the other one. Again, the ruled surface associated with a cusp-crossing event is created by sweeping l while maintaining one second-order and one third-order contact with the surface. The ruled surface touches the object along two curves; we define $\sigma(s)$ to be the curve containing the point

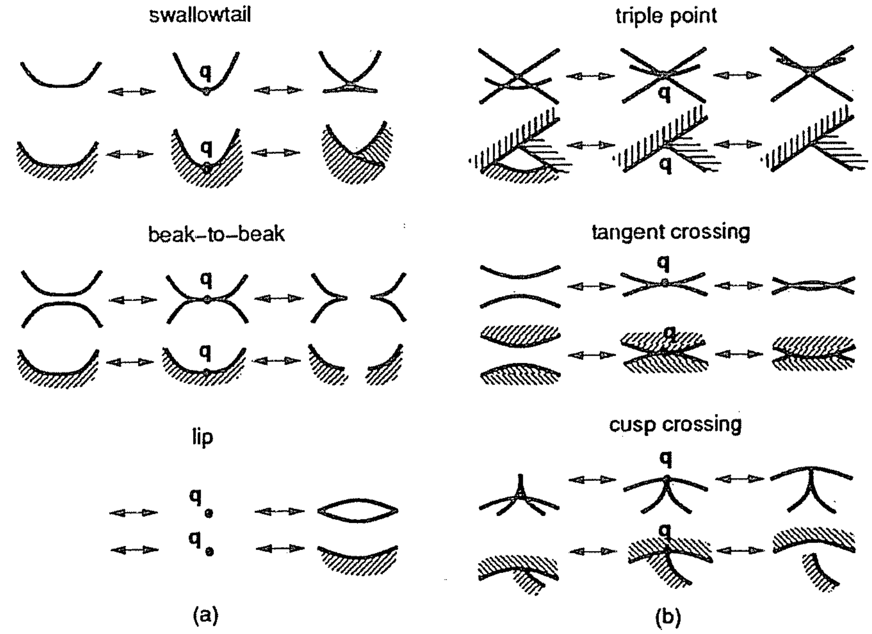


Figure 74: Topological transitions of the occluding contour that correspond to the visual events for a transparent object (adapted from [133]). Also shown are examples of how these transitions appear when the viewed object is opaque. The Epipolar Reconstructibility Constraint C3 is not satisfied for the visible rim point p that projects to q and is farthest away from the viewpoint's position. For viewpoints on the visual events, the line l that is parallel to the viewing direction and contains p has a high order contact with the surface, or touches the surface at multiple points. (a) Local events. In a swallowtail event the occluding contour develops a singularity and then it breaks off into three segments forming two cusps and a T-junction. In a beak-to-beak event two occluding contour curves (of which only one is the projection of a visible rim curve) meet at a point and then split off, generating two cusped contours. In a lip event a cusped contour appears out of nowhere. (b) Multilocal events. In a triple-point event, points on three occluding contour segments project to a single point. In a tangent-crossing event two contours meet creating a pair of T-junctions. Finally, in a cusp-crossing event three occluding contour segments connected by two T-junctions split off with one of the segments ending with a cusp.

farthest away from the viewpoint's position.

From the description of the visual events given above, it follows that we can define the following visual event curves on the surface:

- the parabolic curves of the surface
- the flecnodal curves of the surface

- the curves $\tau(s)$ associated with triple-point events
- the curves $\gamma(s)$ associated with tangent-crossing events
- the curves $\sigma(s)$ associated with cusp-crossing events

The analysis of the visual events and their associated visual event curves of generic surfaces generalizes to non-generic surfaces in the following sense. Visual events for non-generic surfaces also occur when a line parallel to the viewing direction has high order of contact with the surface, or when that line touches the surface at multiple points. The same catalog of visual events is still valid [127]. However, degeneracies may also occur: The visual event "curves" defined above may in fact become two-dimensional regions on the surface (e.g., all points on a cylinder, which is not a generic object, are parabolic). See [127] for a discussion of this issue in the case of algebraic surfaces which are not generic in general, and for the equations describing the visual event curves for these surfaces.

Appendix E

Provable-Correctness of Global Reconstruction

E.1 Proofs of Section 6.3 Theorems

E.1.1 Proof of Theorem 6.1

Let p be an ordinary visible rim point. Consider the epipolar plane, Λ, defined by the vector $\mathrm{v}(t)$ and the line segment connecting p and $c(t)$, and the intersection, α, of S with Λ in the neighborhood of p (Figure 75).

Suppose that the viewpoint does not move on the tangent plane of the surface at p, i.e., $N(p)\cdot \mathrm{v}(t) \neq 0$. In this case, $\Lambda \neq T_p(S)$ and α is a regular curve in the neighborhood of p [89]. In addition, suppose α is parameterized so that the curve normal at p points toward the surface interior. Since p is an ordinary point, p must be a convex point of α. Furthermore, the open line segment connecting p and $c(t)$ does not intersect the surface, implying that p does not become occluded by a distant point of S under an infinitesimal viewpoint change on Λ.

The visibility of p (and of points on α close to p) in this case is determined by the sign of $N(p) \cdot [p - c(t)]$ [89]. Therefore, changes in the visibility state of p under infinitesimal observer motion occur due to changes in the sign of this dot product. Since p is a visible rim point, this dot product is zero at position $c(t)$. Therefore, the visibility of p under an infinitesimal viewpoint change depends on

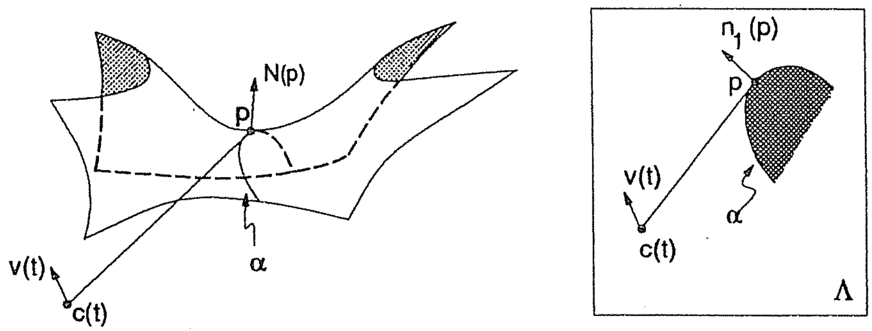


Figure 75: Inducing the visibility of points in a neighborhood of an ordinary hyperbolic point p. (a) The curve α is the curve of intersection of S with the epipolar plane Λ. If $\mathrm{v}(t) \cdot N(p) > 0$, the visible rim point contained in Λ will move toward the previously-occluded portion of α under an infinitesimal viewpoint change along $\mathrm{v}(t)$. (b) A face-on view of the plane Λ. The outward normal $n_1(p)$ of α at p is the projection of $N(p)$ on the plane Λ. The geometry of the intersection of S with the epipolar planes corresponding to visible rim points *close* to p is also similar to the one shown.

the sign of the derivative $\{N(p) \cdot [p - c(t)]\}'$. We have:

$$\frac{d}{dt}\{N(p) \cdot [p - c(t)]\} =$$
$$N(p) \cdot \frac{d}{dt}[p - c(t)] + [\frac{d}{dt}N(p)] \cdot [p - c(t)] =$$
$$- N(p) \cdot \mathrm{v}(t)$$

If the observer moves on p's tangent plane, p may become occluded by points in the neighborhood of p but will always remain on the rim. It will remain visible unless p is hyperbolic and the line connecting $c(t)$ and p is along an asymptotic direction of the surface at p (Figure 14). This, however, cannot occur since p is ordinary. □

E.2 Proofs of Section 6.6 Theorems

E.2.1 Proof of Theorem 6.3

Let $\alpha(t^n)$ be the point on α selected at the n-th iteration of the Incremental Surface Reconstruction Strategy. First, note that if α is not reconstructed in a finite number of steps, the limit, $\alpha(t^\infty)$, exists:

Since the points $\alpha(t^n)$ belong to a curve of finite length, and the length of the reconstructed portion of the curve is an increasing function, the sequence $(\alpha(t^n))_n$ has a limit point.

To prove the theorem we study how the visibility arcs of $\alpha(t^n)$ change as n goes to infinity. The endpoints of these arcs belong to two types of lines on $T_{\alpha(t^n)}(S)$: *Bitangent lines* through $\alpha(t^n)$, i.e., lines that touch the surface at at least two distinct points, one of which is $\alpha(t^n)$, and asymptotes of $\alpha(t^n)$. The finiteness assumption of Theorem 6.3 ensures that these lines and the visibility arcs they delimit are well-defined for large n:

Lemma E.1 *If α intersects the visual event curves at most a finite number of times, and if M is sufficiently large, the configuration of the asymptotes and bitangents through $\alpha(t^n)$ (i.e., their number and relative ordering) does not change for $n > M$.*

Proof: Consider the configuration of asymptotes and bitangents through $\alpha(t)$ on $S \cap T_{\alpha(t)}(S)$, for $0 \leq t \leq T$. The configuration will change at a point $\alpha(t_c)$ if and only if the number of contacts with the surface of lines through $\alpha(t)$ on $T_{\alpha(t)}(S)$ changes or the degree of such contacts changes. By definition, this can only happen if $\alpha(t_c)$ belongs to a visual event curve. Since there is only a finite number of such intersection points, all $\alpha(t^n)$ will fall in a single interval between these points, if $n > M$ and M is sufficiently large. □

To show that the semi-global curve reconstruction task is achieved for α, we consider the angles formed by the bitangent lines and asymptotes at $\alpha(t^n)$ as n goes to infinity as well as their relationship to the continuously-changing viewpoint. This relationship is re-established through tangential viewpoint control at each iteration of the Incremental Surface Reconstruction Strategy.

In particular, after we select point $\alpha(t^n)$ at the n-th iteration, the following steps are then taken: (1) The viewpoint moves to a previous viewpoint, c^n_{vis}, from which $\alpha(t^n)$ was on the visible rim. (2) Since Rule 3 is obeyed, we can assume without loss of generality that $\alpha(t^n)$ is a cusp, T-junction or degenerate visible rim point at c^n_{vis}. In either case, Rule 2 forces the viewpoint to move to the middle, c^n_{mid}, of a visibility arc of $\alpha(t^n)$. (3) The Ordinary Region Reconstruction Strategy is applied to reconstruct a new segment of α, ending at $\alpha(t^n + \delta t^n)$. The Ordinary Region Reconstruction Strategy terminates when $\alpha(t^n + \delta t^n)$ belongs to the visible rim, and Rule 3 allows us to assume without loss of generality that if c^n_{stop} is the viewpoint when this occurs, $\alpha(t^n + \delta t^n)$ is either a cusp, T-junction or degenerate point.

We now use the following two observations:

1. The visibility arc containing c^n_{mid} can be of three types: Type AA, whose endpoints lie on the two asymptotes at $\alpha(t^n)$; type AB, whose endpoints lie on one asymptote of $\alpha(t^n)$ and one bitangent line through $\alpha(t^n)$ on $T_{\alpha(t^n)}(S)$; and type BB, whose endpoints lie on two bitangent lines through $\alpha(t^n)$ on $T_{\alpha(t^n)}(S)$.

2. The line through c^n_{stop} and $\alpha(t^n + \delta t^n)$ is either a bitangent line (B) through $\alpha(t^n + \delta t^n)$ on $T_{\alpha(t^n+\delta t^n)}(S)$, or an asymptote (A) at $\alpha(t^n + \delta t^n)$.

In all, there are six different combinations of the above cases, which we represent by the string **X-Y** where X $= \{AA, AB, BB\}$, and Y $= \{A, B\}$. Without loss of generality we can prove the proposition for each of the above cases separately. Only three of the six cases are considered below. The remaining three cases can be treated in an identical manner.

Case AA-A

This case is shown in Figure 76(a).

The line $c^n_{stop}\alpha(t^n + \delta t^n)$ is an asymptote at $\alpha(t^n + \delta t^n)$. Without loss of generality assume that $\alpha(t^n)$ is also hyperbolic, and let l^n be the corresponding asymptote at $\alpha(t^n)$. Since $\lim_{n\to\infty} \alpha(t^n) = \lim_{n\to\infty} \alpha(t^n + \delta t^n)$, we get

$$\lim_{n\to\infty} \angle(l^n, c^n_{stop}\alpha(t^n + \delta t^n) = 0 \tag{27}$$

Because the motion of the visible rim during the execution of the Ordinary Patch Reconstruction Behavior between c^n_{mid} and c^n_{stop} describes S in the neighborhood of $\alpha(t^n)$, it follows that[1]

$$\lim_{n\to\infty} c^n_{stop} = \lim_{n\to\infty} c^n_{mid} \tag{28}$$

From Eqs. (27) and (28) we conclude that

$$\lim_{n\to\infty} \angle(l^n, c^n_{mid}\alpha(t^n)) = 0 \tag{29}$$

[1]To see this, note that the observer's motion from c^n_{mid} to c^n_{stop} is along a smooth curve, c. Since the segment $\alpha(t)$, $(t^n < t < t^n + \delta t^n)$ can be described by the epipolar parameterization, the observer's motion can be described by the curve $c(t), t^n < t < t^n + \delta t^n$ such that $c(t)$ is the observer's position when $\alpha(t)$ is on the visible rim. The mapping $m : \alpha \to c$ is smooth, and consequently $||c^n_{stop} - c^n_{mid}|| \to 0$ if and only if $||\alpha(t^n + \delta t^n) - \alpha(t^n)|| \to 0$.

Since c^n_{mid} is the midpoint of a visibility arc of $\alpha(t^n)$ of type AA, we have

(30) $$\angle(l^n, c^n_{mid}\alpha(t^n)) = \phi^n/2$$

where ϕ_n is the angle between the two asymptotes at $\alpha(t^n)$. Hence, from Eqs. (29) and (30) it follows that

(31) $$\lim_{n\to\infty} \phi^n = 0$$

This implies that the direction of the asymptotes tends toward the first principal direction, e_1^∞, of the surface at $\alpha(t^\infty)$ (Figure 76(a)).

The angle ψ^n between the asymptotes and the first principal direction, e_1^n, of the surface at $\alpha(t^n)$ is given by

(32) $$\tan\psi^n = \sqrt{-\frac{k_1^n}{k_2^n}} = \tan(\phi^n/2)$$

where k_1^n, k_2^n are the first and second principal curvatures of the surface at $\alpha(t^n)$, respectively.

Since the surface is smooth, k_2^n is always bounded. Hence, from Eqs. (31) and (32) it follows that $\lim_{n\to\infty} k_1^n = 0$. For generic surfaces, this implies that $\alpha(t^n)$ is on a parabolic curve bounding a surface concavity.□

Case AB-A

This case is shown in Figure 76(b). The line $c^n_{stop}\alpha(t^n + \delta t^n)$ is an asymptote at $\alpha(t^n + \delta t^n)$. Without loss of generality assume that $\alpha(t^n)$ is also hyperbolic, and let l^n be the corresponding asymptote at $\alpha(t^n)$. Since $\lim_{n\to\infty} \alpha(t^n) = \lim_{n\to\infty} \alpha(t^n + \delta t^n)$, we get

(33) $$\lim_{n\to\infty} \angle(l^n, c^n_{stop}\alpha(t^n + \delta t^n)) = 0$$

As in Case **AA-A**, we have

(34) $$\lim_{n\to\infty} \angle(l^n, c^n_{mid}\alpha(t^n)) = 0$$

Since the visibility arc containing c^n_{mid} is of type AB, the following equations hold:

(35) $$\angle(l^n, c^n_{mid}\alpha(t^n)) \geq \phi^n/2$$

(36) $$\angle(l_1^n, c^n_{mid}\alpha(t^n)) = \phi^n/2$$

(37) $$\angle(l_2^n, c^n_{mid}\alpha(t^n)) = \phi^n/2$$

where l_1^n and l_2^n are the asymptote and bitangent line, respectively, bounding the visibility arc containing c_{mid}^n, and ϕ^n is the angle they form. From Eqs. (34) and (35) it follows that

$$\lim_{n\to\infty} \phi^n = 0 \tag{38}$$

Since l_2^n is a bitangent line, it contacts the surface at an additional point, q^n. Lemma E.1 tells us that we can assume without loss of generality that for $n > M$, the points q^n belong to a single curve.[2] Consequently, the limit $\lim_{n\to\infty} q^n$ exists and is equal to some point q^∞. We now distinguish two cases:

- $q^\infty \neq \alpha(t^\infty)$. Then the line $q^\infty\alpha(t^\infty)$ touches the surface at two distinct points. Furthermore, from Eqs. (36)-(38) we can conclude that the line $q^\infty\alpha(t^\infty)$ coincides with the line $\lim_{n\to\infty} l_1^n$, which is an asymptote at $\alpha(t^\infty)$. Hence, $\alpha(t^\infty)$ belongs to the curve σ associated with a cusp-crossing visual event. Furthermore, the visibility arc containing c_{mid}^n diminishes as $\alpha(t^n)$ approaches $\alpha(t^\infty)$ (i.e., it approaches a visual event curve that is a *potential* boundary of a reconstructible surface region).

- $q^\infty = \alpha(t^\infty)$. Since l_1^n and l_2^n bound the region on $T_{\alpha(t^n)}(S)$ from which $\alpha(t^n)$ is visible, and their angle, ϕ^n, tends to zero, it follows that (1) the limit $\lim_{n\to\infty} l_2^n$ exists and is an asymptote at $\alpha(t^\infty)$, and (2) the angle between $\alpha(t^n)$'s asymptotes tends to zero. From the analysis of Case AA-A it now follows that $\alpha(t^n)$ belongs to a parabolic curve bounding a surface concavity.

□

Case BB-B

This case is shown in Figure 76(c).

The line $c_{stop}^n\alpha(t^n + \delta t^n)$ is a bitangent line at $\alpha(t^n + \delta t^n)$. Without loss of generality assume that the topology of the intersection, $T_{\alpha(t)}(S) \cap S$, does not change for $t^n < t < t^n + \delta t^n$, and let l^n be the

[2]Since the configuration of asymptotes and bitangents does not change for $n > M$, the number of bitangent lines in $T_{\alpha(t^n)}(S)$ will be equal to some fixed constant k. Furthermore, these lines can be grouped into k continuous families of lines whose contacts with the surface trace continuous curves. The line l_2^n will belong to one of those k families. We can therefore partition the sequence $(\alpha(t^n))_n$ into k subsequences in which l_2^n always belongs to the same family. For each such subsequence, l_2^n contacts the surface at an additional point q^n that belongs to the continuous trace of contacts of a single family of bitangent lines.

corresponding bitangent line at $\alpha(t^n)$. Since $\lim_{n\to\infty}\alpha(t^n) = \lim_{n\to\infty}\alpha(t^n+\delta t^n)$, we get

(39) $$\lim_{n\to\infty}\angle(l^n, c^n_{stop}\alpha(t^n+\delta t^n)) = 0$$

As in Case AA-A, we have

(40) $$\lim_{n\to\infty}\angle(l^n, c^n_{mid}\alpha(t^n)) = 0$$

Since the visibility arc containing c^n_{mid} is of type BB, the following equations hold:

(41) $$\angle(l^n, c^n_{mid}\alpha(t^n)) \geq \phi^n/2$$

(42) $$\angle(l^n_1, c^n_{mid}\alpha(t^n)) = \phi^n/2$$

(43) $$\angle(l^n_2, c^n_{mid}\alpha(t^n)) = \phi^n/2$$

where l^n_1 and l^n_2 are the two bitangent lines bounding the visibility arc containing c^n_{mid}, and ϕ^n is the angle they form. From Eqs. (40) and (41) it follows that

(44) $$\lim_{n\to\infty}\phi_n = 0$$

Since l^n_1 and l^n_2 are bitangent lines through $\alpha(t^n)$, they contact the surface at points q^n_1 and q^n_2, respectively, distinct from $\alpha(t^n)$. As in Case AB-A, we assume without loss of generality that the points q^n_1 and q^n_2 belong to the trace of two continuous curves on the surface. Hence, the limits $q^\infty_1 = \lim_{n\to\infty} q^n_1$ and $q^\infty_2 = \lim_{n\to\infty} q^n_2$ exist.

We now distinguish four cases:

- $\alpha(t^\infty), q^\infty_1, q^\infty_2$ are distinct. In this case the lines $\alpha(t^\infty)q^\infty_1$ and $\alpha(t^\infty)q^\infty_2$ are bitangent lines. From Eqs. (42)-(44) we conclude that the two lines are identical, and hence $\alpha(t^\infty)q^\infty_1$ is tangent to the surface at three distinct points. Furthermore, $\alpha(t^\infty)$ is not between points q^∞_1 and q^∞_2. Consequently, $\alpha(t^\infty)$ belongs to a visual event curve τ_1 or τ_2 associated with some triple-point visual event. Furthermore, the visibility arc containing c^n_{mid} diminishes as $\alpha(t^n)$ approaches that point (i.e., it approaches a visual event curve that is a *potential* boundary of a reconstructible surface region).

- $q^\infty_1 = q^\infty_2 \neq \alpha(t^\infty)$. This implies that the topology of the intersection $T_{\alpha(t^\infty)}(S)\cap S$ in the neighborhood of q^∞_1 is different from that of $T_{\alpha(t^n)}(S)\cap S$ for any large n. This occurs only when

$T_{\alpha(t^\infty)}(S)$ is tangent to the surface at point q_1^∞ [94], and q_1^∞ is hyperbolic. Hence, $T_{\alpha(t^\infty)}(S)$ is tangent to the surface at two distinct points and, consequently, $\alpha(t^\infty)$ belongs to a visual event curve γ_1 or γ_2 associated with a tangent-crossing event. Furthermore, the visibility arc containing c^n_{mid} diminishes as $\alpha(t^n)$ approaches that point (i.e., it approaches a visual event curve that is a *potential* boundary of a reconstructible surface region).

- $\alpha(t^\infty) \in \{q_1^\infty, q_2^\infty\}$ and $q_1^\infty \neq q_2^\infty$. Suppose $\alpha(t^\infty) = q_1^\infty$. Since the lines l_1^n, l_2^n bound a region in which $\alpha(t^n)$ is visible and $q_1^\infty = \alpha(t^\infty)$, it follows that l_1^∞ is an asymptote at $\alpha(t^\infty)$. But Eq. (44) implies that l_1^∞ also touches the surface at q_2^∞. Hence, $\alpha(t^\infty)$ belongs to a visual event curve σ_{cusp} associated with a cusp-crossing visual event. Furthermore, the visibility arc containing c^n_{mid} diminishes as $\alpha(t^n)$ approaches that point (i.e., it approaches a visual event curve that is a *potential* boundary of a reconstructible surface region).

- $\alpha(t^\infty) = q_1^\infty = q_2^\infty$. Since the lines l_1^n, l_2^n bound a region in which $\alpha(t^n)$ is visible, it follows that l_1^∞, l_2^∞ are the two asymptotes at $\alpha(t^\infty)$. Eq. (44) and our analysis in Case AA-A imply that $\alpha(t^\infty)$ is on a parabolic curve bounding a surface concavity.

□

E.2.2 Proof of Theorem 6.5

Suppose Γ^n is the set of surface points reconstructed after n iterations of the Incremental Reconstruction Behavior. Furthermore, suppose the reconstruction process never terminates. Let $\Gamma^\infty = \cup_{n\to\infty}\Gamma^n$, and take q to be a limit point of Γ^∞. We show that q is a limit point of a reconstructible surface region.

Let β be a curve lying in the open set $\overline{\Gamma^\infty} - \Gamma^\infty$, connecting q to the visible rim at the initial viewpoint. By definition, any point on β except its endpoint, q, will be reconstructed after a finite number of iterations of the Incremental Reconstruction Behavior. We distinguish two cases, depending on whether or not q belongs to a curve γ_1 or γ_2 corresponding to a tangent-crossing visual event.

First, suppose q does not belong to such a curve. In this case, we can find for any $\epsilon > 0$, a neighborhood Π of q of radius less than ϵ, such that Π does not intersect any visual event curves γ_1, γ_2. Since the observer obeys Rule 1 of Theorem 6.5 and β always intersects the boundary of reconstructed points, it follows that after a finite number of iterations of the Incremental Reconstruction Behavior, a point p will be selected that is contained in Π. After selecting p, the observer executes the Degenerate Patch Reconstruction Behavior to reconstruct the surface around p.

Now, we can define a smooth curve $\gamma(t)$ $(0 < t < \epsilon)$, such that $\gamma(0) = p$, $\gamma(\epsilon) = q$, and $\gamma(t)$ is contained in Π for all t. By its definition, q will not be contained in the patch reconstructed by the observer. Let r be the first point on γ intersecting the boundary of this patch. If $r = \gamma(t_r)$, the topology of the intersection $T_{\gamma(t)}(S) \cap S$ is the same for all $0 < t < t_r$. Hence, we can use the arguments in Theorem 6.3 to conclude that q is a limit point of a reconstructible surface region.

Now suppose q belongs to some curve, γ_1 or γ_2, corresponding to a tangent-crossing visual event, and let $\epsilon > 0$. Only a finite number, K, of such curves can intersect at q. These curves partition every neighborhood Π of q of radius less than ϵ into K regions, in a star-shaped fashion. After a finite number of iterations of the Incremental Reconstruction Behavior, a point p will be selected that is contained in one of those K regions. Without loss of generality, we may assume that p does not belong to the boundaries of these regions. We can now define a curve γ connecting p to q such that its trace is contained in Π, and intersecting a visual event curve, γ_1 or γ_2, only at q. This reduces the theorem's proof to the previous case.$\square$

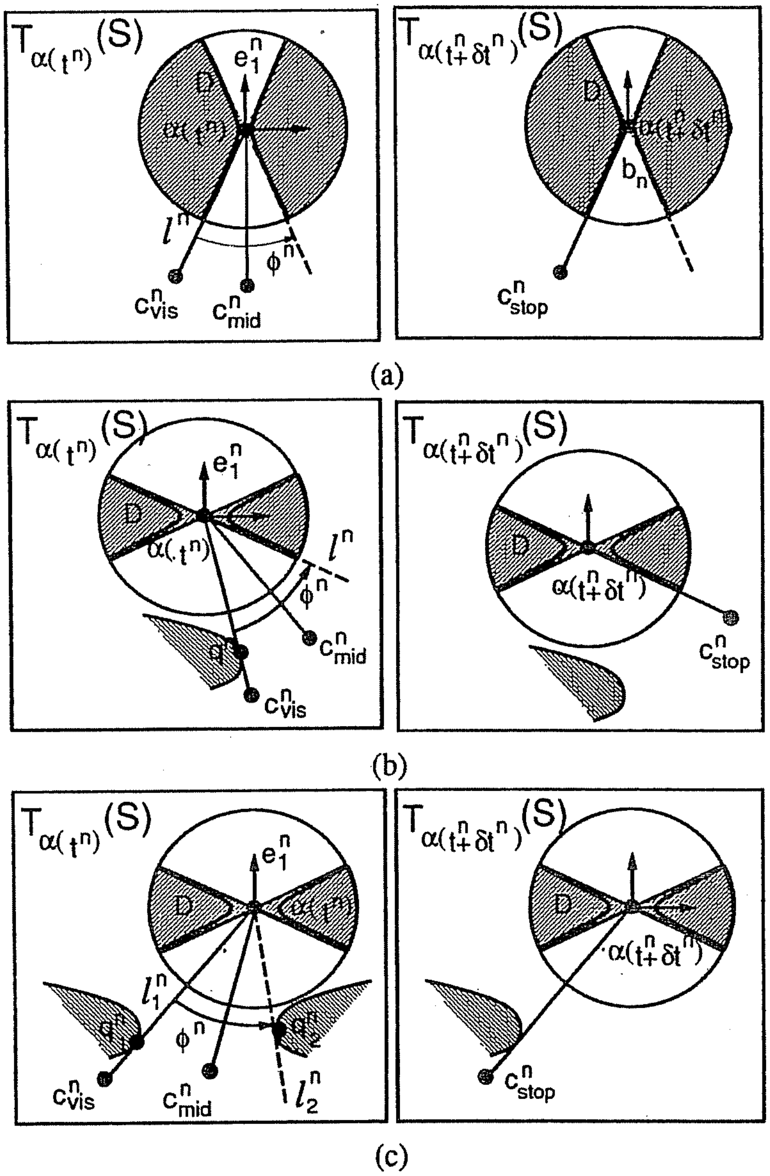


Figure 76: Representing the configurations of the asymptotes and bitangent lines through $\alpha(t^n)$ and $\alpha(t^n + \delta t^n)$. For each of the three cases, the left figure corresponds to a "top" view of the plane $T_{\alpha(t^n)}(S)$ while the right figure corresponds to a "top" view of the plane $T_{\alpha(t^n+\delta t^n)}(S)$. (a) Case AA-A. (b) Case AB-A. (c) Case BB-B.

Bibliography

[1] Narendra Ahuja and Jack Veenstra. Generating octrees from object silhouettes in orthographic views. *IEEE Trans. Pattern Anal. Machine Intell.*, 11(2):137–149, 1989.

[2] Peter K. Allen. Integrating vision and touch for object recognition tasks. *Int. J. Robotics Res.*, 7(6):15–33, 1988.

[3] Peter K. Allen and Paul Michelman. Acquisition and interpretation of 3-d sensor data from touch. In *Proc. Workshop on Interpretation of 3D Scenes*, pages 33–40, 1989.

[4] Yiannis Aloimonos. Visual shape computation. *Proc. IEEE*, 76:899–916, 1988.

[5] Yiannis Aloimonos. Perspective approximations. *Image and Vision Computing*, 8:179–192, 1990.

[6] Yiannis Aloimonos. Purposive and qualitative active vision. In *Proc. Int. Conf. on Pattern Recognition*, pages 346–360, 1990.

[7] Yiannis Aloimonos, editor. *Active Perception.* Lawrence Erlbaum Associates, 1993.

[8] Yiannis Aloimonos and Michael Swain. Shape from patterns: regularization. *Int. J. Computer Vision*, 2:171–187, 1988.

[9] Yiannis Aloimonos, Isaac Weiss, and Amut Bandyopadhyay. Active vision. In *Proc. 1st Int. Conf. on Computer Vision*, pages 35–54, 1987.

[10] Yiannis Aloimonos (guest editor). Special issue on purposive, qualitative, active vision. *CVGIP: Image Understanding*, 56(1), 1992.

[11] Ruzena Bajcsy. Active perception vs. passive perception. In *Proc. IEEE Workshop on Computer Vision*, pages 55–59, 1985.

[12] Ruzena Bajcsy. Active perception. *Proc. IEEE*, 76(8):996–1005, 1988.

[13] Dana H. Ballard. Reference frames for animate vision. In *Proc. Int. Joint Conf. Artificial Intelligence*, pages 1635–1641, 1989.

[14] Dana H. Ballard. Animate vision. *Artificial Intelligence*, 48:57–86, 1991.

[15] Dana H. Ballard and Christopher M. Brown. Principles of animate vision. *Computer Vision, Graphics, and Image Processing: Image Understanding*, 56(1):3–21, 1992. Special Issue on Purposive, Qualitative, Active Vision.

[16] Dana H. Ballard and Altan Ozcandarli. Eye fixation and early vision: Kinetic depth. In *Proc. 2nd Int. Conf. on Computer Vision*, pages 524–531, 1988.

[17] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of optical flow techniques. *Int. J. Computer Vision*, 12(1):43–77, 1994.

[18] H. G. Barrow and J. M. Tenenbaum. Interpreting line drawings as three-dimensional images. *Artificial Intelligence*, 17:75–116, 1981.

[19] Ronen Basri and Shimon Ullman. The alignment of objects with smooth surfaces. In *Proc. 2nd Int. Conf. on Computer Vision*, pages 482–488, 1988.

[20] John S. Bay. A fully autonomous active sensor-based exploration concept for shape-sensing robots. *IEEE Trans. Syst. Man Cybern.*, 21(4):850–860, 1991.

[21] P. A. Beardsley, A. Zisserman, and D. W. Murray. Navigation using affine structure from motion. In *Proc. European Conf. Computer Vision*, pages 85–96, 1994.

[22] Paul J. Besl. Active, optical range imaging sensors. *Machine Vision and Applications*, 1:127–152, 1988.

[23] Paul J. Besl. Geometric modeling and computer vision. *Proc. IEEE*, 76(8):936–958, 1988.

[24] Paul J. Besl and Ramesh C. Jain. Invariant surface characteristics for 3d object recognition in range images. *Computer Vision, Graphics and Image Processing*, 33:33–80, 1986.

[25] A. Blake, M. Brady, R. Cipolla, Z. Xie, and A. Zisserman. Visual navigation around curved obstacles. In *Proc. IEEE Robotics Automat. Conf.*, pages 2490–2495, 1991.

[26] Andrew Blake. Computational modeling of hand-eye coordination. In Yiannis Aloimonos, editor, *Active Perception*, pages 227–244. Lawrence Erlbaum Associates, 1993.

[27] Andrew Blake and Heinrich Bulthoff. Shape from specularities: computation and psychophysics. *Phil. Trans. R. Soc. Lond.*, 331:237–252, 1991.

[28] Andrew Blake, Rupert Curwen, and Andrew Zisserman. A framework for spatio-temporal control in the tracking of visual contours. *Int. J. Computer Vision*, 11(2):127–145, 1993.

[29] Andrew Blake, Michael Taylor, and Adrian Cox. Grasping visual symmetry. In *Proc. Int. Conf. Computer Vision*, pages 724–733, 1993.

[30] Andrew Blake and Alan Yuille, editors. *Active Vision*. MIT Press, 1992.

[31] Andrew Blake, Andrew Zisserman, and Roberto Cipolla. Visual exploration of free-space. In Andrew Blake and Alan Yuille, editors, *Active Vision*, pages 175–188. MIT Press, 1992.

[32] Robert C. Bolles, H. Harlyn Baker, and David H. Marimont. Epipolar-plane image analysis: An approach to determining structure from motion. *Int. J. Computer Vision*, 1:7–55, 1987.

[33] Kevin W. Bowyer and Charles R. Dyer. Three-dimensional shape representation. In Tzay Y. Young, editor, *Handbook of Pattern Recognition and Image Processing: Computer Vision*, pages 17–51. Academic Press, 1994.

[34] Michael Brady, Jean Ponce, Alan Yuille, and Haruo Asada. Describing surfaces. *Computer Graphics and Image Processing*, 32:1–28, 1985.

[35] Michael Brady and Alan Yuille. An extremum principle for shape from contour. *IEEE Trans. Pattern Anal. Machine Intell.*, 6(3):288–301, 1984.

[36] James Callahan and Richard Weiss. A model for describing surface shape. In *Proc. Computer Vision and Pattern Recognition Conf.*, pages 240–245, 1985.

[37] Mario F. M. Campos. *Robotic Exploration of Material and Kinematic Properties Of Objects.* PhD thesis, University of Pennsylvania, 1992.

[38] Manfredo P. Do Carmo. *Differential Geometry of Curves and Surfaces.* Prentice-Hall Inc., Englewood Cliffs, NJ, 1976.

[39] A. Cayley. On contour and slope lines. *Lond., Edinburgh, and Dublin Phil. Mag. and J. Sci.*, 18(120):264–268, 1859. Ser. 4.

[40] C. H. Chien and J. K. Aggarwal. Computation of volume/surface octrees from contours and silhouettes of multiple views. In *Proc. Computer Vision and Pattern Recognition*, pages 250–255, 1986.

[41] Wee-Soon Ching, Peng-Seng Toh, Kap-Luk Chan, and Meng-Hwa Er. Robust vergencewith concurrent detection of occlusion and specular highlights. In *Proc. Int. Conf. Computer Vision*, pages 384–394, 1993.

[42] Roberto Cipolla. *Active Visual Inference of Surface Shape.* PhD thesis, University of Oxford, 1991.

[43] Roberto Cipolla and Andrew Blake. Qualitative surface shape from deformation of image curves. *Int. J. Computer Vision*, 8(1):53–69, 1992.

[44] Roberto Cipolla and Andrew Blake. Surface shape from the deformation of apparent contours. *Int. J. Computer Vision*, 9(2):83–112, 1992.

[45] C. I. Connolly. The determination of next best views. In *Proc. IEEE Robotics Automat. Conf.*, pages 432–435, 1985.

[46] S. D. Conte and Carl de Boor. *Elementary Numerical Analysis.* McGraw-Hill Inc., 1972.

[47] David Coombs and Christopher Brown. Real-time binocular smooth pursuit. *Int. J. Computer Vision*, 11(2):147–164, 1993.

[48] Cregg K. Cowan. Automatic camera and light-source placement using cad models. In *IEEE Workshop on Directions in Automated CAD-Based Vision*, pages 22–31, 1991.

[49] Cregg K. Cowan and Peter D. Kovesi. Automatic sensor placement from vision task requirements. *IEEE Trans. Pattern Anal. Machine Intell.*, 10(3):407–416, 1988.

[50] Harald Cramér. *Mathematical Methods of Statistics*. Princeton University Press, 1946.

[51] R. Curwen, A. Blake, and A. Zisserman. Real-time visual tracking for surveillance and path planning. In *Proc. 2nd European Conf. on Computer Vision*, pages 879–883, 1992.

[52] Rachid Deriche and Gérard Giraudon. A computational approach for corner and vertex detection. *Int. J. Computer Vision*, 10(2):101–124, 1993.

[53] Sven Dickinson, Henrik I. Christensen, John Tsotsos, and Goran Olofsson. Active object recognition integrating attention and viewpoint control. In *Proc. Third European Conference on Computer Vision*, volume 2, pages 3–14, 1994.

[54] Olivier D. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig? In *Proc. 2nd European Conf. on Computer Vision*, pages 563–578, 1992.

[55] John T. Feddema and Owen Robert Mitchell. Vision-guided servoing with feature-based trajectory generation. *IEEE Trans. Robotics Automat.*, 5(5):691–700, 1989.

[56] John C. Fiala, Ronald Lumia, Karen J. Roberts, and Albert J. Wavering. TRICLOPS: A tool for studying active vision. *Int. J. Computer Vision*, 12(2/3):231–250, 1994.

[57] James D. Foley, Andries van Dam, Steven K. Feiner, and John F. Hughes. *Computer Graphics Principles and Practice*. Addison-Wesley Publishing Co., 1990.

[58] David Forsyth, Joseph L. Mundy, Andrew Zisserman, Chris Coelho, and Charles Rothwell. Invariant descriptors for 3-D object recognition and pose. *IEEE Trans. Pattern Anal, Machine Intell.*, 13(10):971–991, 1991.

[59] Jacob Fraden. *AIP Handbook of Modern Sensors*. American Institute of Physics, 1993.

[60] P. Giblin and M. G. Soares. On the geometry of a surface and its singular profiles. *Image and Vision Computing*, 6(4):225–234, 1988.

[61] Peter Giblin and Richard Weiss. Epipolar curves on surfaces. *Image and Vision Computing*. To appear.

[62] Peter Giblin and Richard Weiss. Reconstruction of surfaces from profiles. In *Proc. 1st Int. Conf. on Computer Vision*, pages 136–144, 1987.

[63] Peter Giblin and Richard Weiss. Epipolar fields on surfaces. In *Proc. 3rd European Conf. on Computer Vision*, pages 14–23, 1994.

[64] James J. Gibson. *The Senses Considered as Perceptual Systems.* Houghton Mifflin Company, 1966.

[65] James J. Gibson. *The Ecological Approach to Visual Perception.* Houghton Mifflin, 1979.

[66] James J. Gibson. *The Ecological Approach to Visual Perception.* Laurence Erlbaum Assoc., 1986.

[67] Gene H. Golub and Charles F. van Loan. *Matrix Computations.* The Johns Hopkins University Press, 1983.

[68] Keith D. Gremban and Katsushi Ikeuchi. Planning multiple observations for object recognition. *Int. J. Computer Vision*, 12(2/3):137–172, 1994.

[69] W. Eric L. Grimson. Sensing strategies for disambiguating among multiple objects in known poses. *IEEE J. Robotics Automat.*, 2:196–213, 1986.

[70] Enrico Grosso and Dana H. Ballard. Head-centered orientation strategies in animate vision. In *Proc. 4th Int. Conf. on Computer Vision*, pages 395–402, 1993.

[71] Greg Hager and Seth Hutchinson, editors. *Visual Servoing: Achievements, Applications, and Open Problems*, 1994. Held in conjunction with the 1994 IEEE International Conference on Robotics and Automation.

[72] Greg Hager and Max Mintz. Searching for information. In *Proc. Workshop on Spatial Reasoning and Multi-Sensor Fusion*, pages 313–322, 1987.

[73] Koichi Hashimoto. *Visual Servoing*. World Scientific, 1993.

[74] Martial Hebert and Eric Krotkov. 3d measurements from imaging laser radars: how good are they? *Image and Vision Computing*, 10(3):170–178, 1992.

[75] Jean-Yves Hervé and Yiannis Aloimonos. Exploratory active vision: Theory. In *Proc. Computer Vision and Pattern Recognition*, pages 10–15, 1992.

[76] K. Higuchi, H. Delingette, M. Hebert, and K. Ikeuchi. Merging multiple views using a spherical representation. In *Second CAD-Based Vision Workshop*, pages 124–131, 1994.

[77] Radu Horaud and Michael Brady. On the geometric interpretation of image contours. In *Proc. 1st Int. Conf. on Computer Vision*, pages 374–382, 1987.

[78] B. K. P. Horn. *Robot Vision.* MIT Press, 1986.

[79] D. Huffman. Impossible objects as nonsense sentences. In B. Meltzer and D. Michie, editors, *Machine Intelligence 6*, pages 295–323. Edimburgh University Press, 1971.

[80] Seth Hutchinson. Exploiting visual constraints in robot motion planning. In *Proc. IEEE Robotics Automat. Conf.*, pages 1722–1727, 1991.

[81] Seth A. Hutchinson and Avinash Kak. Planning sensing strategies in a robot work cell with multi-sensor capabilities. *IEEE Trans. Robotics Automat.*, 5(6):765–783, 1989.

[82] Tanuja Joshi, Narendra Ahuja, and Jean Ponce. Structure and motion estimation from dynamic silhouettes. Technical Report UIUC-BI-AI-RCV-94-01, University of Illinois, April 1994.

[83] Takeo Kanade. Recovery of the three-dimensional shape of an object from a single view. *Artificial Intelligence*, 17:409–460, 1981.

[84] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *Int. J. Computer Vision*, 1(4):321–331, 1988.

[85] Yannick Louis Kergosien. La famille des projections orthogonales d'une surface et ses singularités. *C. R. Acad. Sc. Paris*, 292:929–932, 1981.

[86] Hwang-Soo Kim, Ramesh C. Jain, and Richard A. Volz. Object recognition using multiple views. In *Proc. IEEE Robotics Automat. Conf.*, pages 28–33, 1985.

[87] Roberta L. Klatzky and Susan J. Lederman. The intelligent hand. *The Psychology of Learning and Motivation*, 21:121–151, 1987.

[88] Jan J. Koenderink. What does the occluding contour tell us about solid shape? *Perception*, 13:321–330, 1984.

[89] Jan J. Koenderink. An internal representation for solid shape based on the topological properties of the apparent contour. In Whitman Richards and Shimon Ullman, editors, *Image Understanding 1985-86*, pages 257–285. Ablex Publishing Co., Norwood, NJ, 1987.

[90] Jan J. Koenderink. *Solid Shape*. MIT Press, 1990.

[91] Jan J. Koenderink and A. J. van Doorn. The singularities of the visual mapping. *Biological Cybernetics*, 24:51–59, 1976.

[92] Jan J. Koenderink and A. J. van Doorn. The internal representation of solid shape with respect to vision. *Biological Cybernetics*, 32:211–216, 1979.

[93] Jan J. Koenderink and A. J. van Doorn. Photometric invariants to solid shape. *Optica Acta*, 27(7):981–996, 1980.

[94] Jan J. Koenderink and A. J. van Doorn. A description of the structure of visual images in terms of an ordered hierarchy of light and dark blobs. In *Proc. Second Int. Visual Psychophysics and Medical Imaging Conf.*, pages 173–176, 1981.

[95] Jan J. Koenderink and Andrea J. van Doorn. Affine structure from motion. *J. Opt. Soc. Am.*, A(2):377–385, 1991.

[96] David J. Kriegman, 1993. Personal communication.

[97] David J. Kriegman and Jean Ponce. On recognizing and positioning curved 3-d objects from image contours. *IEEE Trans. Pattern Anal. Machine Intell.*, 12(12):1127–1137, 1990.

[98] David J. Kriegman, B. Vijayakumar, and Jean Ponce. Reconstruction of HOT curves for image sequences. In *Proc. Computer Vision and Pattern Recognition Conf.*, pages 20–26, 1993.

[99] Eric Krotkov. Focusing. *Int. J. Computer Vision*, 1:223–237, 1987.

[100] Eric Krotkov and Ruzena Bajcsy. Visual behaviors for reliable ranging: Cooperating focus, stereo, and vergence. *Int. J. Computer Vision*, 11(2):187–203, 1993.

[101] Kiriakos N. Kutulakos, Vladimir J. Lumelsky, and Charles R. Dyer. Vision-guided exploration: A step toward general motion planning in three dimensions. In *Proc. IEEE Robotics Automat. Conf.*, pages 289–296, 1993.

[102] Kiriakos N. Kutulakos, W. Brent Seales, and Charles R. Dyer. Building global object models by purposive viewpoint control. In *Proc. Second CAD-Based Vision Workshop*, pages 169–182, 1994.

[103] Jean Claude Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, 1991.

[104] C. Laugier, A. Ijel, and J. Troccaz. Combining vision based information and partial geometric models in automatic grasping. In *Proc. IEEE Robotics Automat. Conf.*, pages 676–682, 1990.

[105] Aldo Laurentini. The visual hull concept for silhouette-based image understanding. *IEEE Trans. Pattern Anal. Machine Intell.*, 16(2):150–162, 1994.

[106] Sang Wook Lee and Ruzena Bajcsy. Detection of specularity using color and multiple views. In *Proc. Second European Conference on Computer Vision*, pages 99–114, 1992.

[107] Sukhan Lee and Hernsoo Hahn. An optimal sensing strategy for recognition and localization of 3-d natural quadric objects. *IEEE Trans. Pattern Anal. Machine Intell.*, 13(10):1018–1037, 1991.

[108] Michael Leyton. A process grammar for shape. *Artificial Intelligence*, 34:213–247, 1988.

[109] H. C. Longuet-Higgins and K. Prazdny. The interpretation of a moving retinal image. *Proc. Royal Society of London*, 208:385–397, 1980. Ser. B.

[110] David G. Lowe. Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, 31:355–395, 1987.

[111] Carl Machover and Steve E. Tice (guest editor). Special issue on virtual reality. *IEEE Computer Graphics and Applications*, 14(1), 1994.

[112] Claus B. Madsen and Henrik I. Christensen. Localizing un-calibrated, reactive camera motion in an object centered coordinate system. In *Proc. Workshop on Visual Behaviors*, pages 119–123, 1994.

[113] Michael Magee and Michell Nathan. Spatial reasoning, sensor repositioning and disambiguation in 3d model based recognition. In *Proc. Workshop on Spatial Reasoning and Multi-Sensor Fusion*, pages 262–271, 1987.

[114] Jitendra Malik. Interpreting line drawings of curved objects. *Int. J. Computer Vision*, 1:73–103, 1987.

[115] David Marr. *Vision*. Freeman, 1982.

[116] David Marr and H. Keith Nishihara. Visual information processing: Artificial intelligence and the sensorium of sight. *Technology Review*, 81:2–23, 1978.

[117] Jasna Maver and Ruzena Bajcsy. Occlusions as a guide for planning the next view. *IEEE Trans. Pattern Anal. Machine Intell.*, 15(5):417–433, 1993.

[118] J. L. Mundy and A. Zisserman, editors. *Geometric Invariance in Computer Vision*. MIT Press, 1992.

[119] David W. Murray, Fenglei Du, Philip F. McLauchlan, Ian D. Reid, Paul M. Sharkey, and Michael Brady. Design of stereo heads. In Andrew Blake and Alan Yuille, editors, *Active Vision*, pages 155–172. MIT Press, 1992.

[120] Lee R. Nackman. Two-dimensional critical point configuration graphs. *IEEE Trans. Pattern Anal. Machine Intell.*, 6(4):442–450, 1984.

[121] Ramakant Nevatia and Mourad Zerroug. Recovery of three-dimensional shape of curved objects from a single image. In Tzay Y. Young, editor, *Handbook of Pattern Recognition and Image Processing: Computer Vision*, pages 101–129. Academic Press, 1994.

[122] Timothy S. Newman and Anil K. Jain. CAD-Based inspection of 3D objects using range images. In *Proc. Second CAD-Based Vision Workshop*, pages 236–243, 1994.

[123] Kourosh Pahlavan and Jan-Olof Eklundh. A head-eye system — analysis and design. *Computer Vision, Graphics, and Image Processing: Image Understanding*, 56(1):41–56, 1992. Special Issue on Purposive, Qualitative, Active Vision.

[124] Kourosh Pahlavan, Tomas Uhlin, and Jan-Olof Eklundh. Active vision as a methodology. In Yiannis Aloimonos, editor, *Active Perception*, pages 19–46. Lawrence Erlbaum Associates, 1993.

[125] Kourosh Pahlavan, Tomas Uhlin, and Jan-Olof Eklundh. Dynamic fixation. In *Proc. Int. Conf. on Computer Vision*, pages 412–419, 1993.

[126] Alex P. Pentland and Martin Bichel. Extracting shape-from-shading. In Tzay Y. Young, editor, *Handbook of Pattern Recognition and Image Processing: Computer Vision*, pages 161–183. Academic Press, 1994.

[127] Sylvain Petitjean, Jean Ponce, and David J. Kriegman. Computing exact aspect graphs of curved objects: Algebraic surfaces. *Int. J. Computer Vision*, 9(3):231–255, 1992.

[128] Les Piegl. On nurbs: A survey. *IEEE Computer Graphics and Applications*, 11(1):55–71, 1991.

[129] O. A. Platonova. Singularities of the mutual disposition of a surface and a line. *Russian Math. Surveys*, 36(1):248–249, 1981.

[130] O. A. Platonova. Singularities of projections of smooth surfaces. *Russian Math. Surveys*, 39(1):177–178, 1984.

[131] Jean Ponce and David Chelberg. Finding the limbs and cusps of generalized cylinders. *Int. J. Computer Vision*, 1:195–210, 1987.

[132] Jean Ponce, David Chelberg, and Wallace B. Mann. Invariant properties of straight homogeneous generalized cylinders and their contours. *IEEE Trans. Pattern Anal. Machine Intell.*, 11(9):951–966, 1990.

[133] Jean Ponce, Sylvain Petitjean, and David Kriegman. Computing exact aspect graphs of curved objects: Algebraic surfaces. In *Proc. Second European Conference on Computer Vision*, 1992.

[134] John Porrill and Stephen Pollard. Curve matching and stereo calibration. *Image and Vision Computing*, 9(1):45–50, 1991.

[135] Aristides A. G. Requicha and Jarek R. Rossignac. Solid modeling and beyond. *IEEE Computer Graphics and Applications*, 12(5):31–44, 1992.

[136] Whitman Richards, Jan J. Koenderink, and D. D. Hoffman. Inferring 3d shapes from 2d silhouettes. In Whitman Richards, editor, *Natural Computation*, pages 125–136. MIT Press, Cambridge, MA, 1988.

[137] M. Richetin, M. Dhome, J. T. Lapreste, and G. Rives. Inverse perspective transform using zero-curvature contour points: Application to the localization of some generalized cylinders from a single view. *IEEE Trans. Pattern Anal. Machine Intell.*, 13(2):185–192, 1991.

[138] J. H. Rieger. Three-dimensional motion from fixed points of a deforming profile curve. *Optics Letters*, 11(3):123–125, 1986.

[139] J. H. Rieger. On the classification of views of piecewise smooth objects. *Image and Vision Computing*, 5(2):91–97, 1987.

[140] J. H. Rieger. The geometry of view space of opaque objects bounded by smooth surfaces. *Artificial Intelligence*, 44:1–40, 1990.

[141] J. H. Rieger and D. H. Lawton. Processing differential image motion. *J. Opt. Soc. Am.*, 2(2):354–359, 1985.

[142] Raymond D. Rimey and Christopher M. Brown. Control of selective perception using bayes nets and decision theory. *Int. J. Computer Vision*, 12(2):173–207, 1994.

[143] Shigeyuki Sakane, Masaru Ishii, and Masayoshi Kakikura. Occlusion avoidance of visual sensors based on a hand-eye action simulator system: Heaven. *Advanced Robotics*, 2(2):149–165, 1987.

[144] G. Sandini, F. Gandolfo, and E. Tistarelli. Vision during action. In Yiannis Aloimonos, editor, *Active Perception*, pages 151–190. Lawrence Erlbaum Associates, 1993.

[145] W. Brent Seales and Charles R. Dyer. Modeling the rim appearance. In *Proc. Int. Conf. on Computer Vision*, pages 698–701, 1990.

[146] W. Brent Seales and Olivier D. Faugeras. Building three-dimensional cad/cam models from image sequences. In *Proc. Second CAD-Based Vision Workshop*, pages 116–123, 1994.

[147] Steven A. Shafer. *Shadows and silhouettes in computer vision*. Kluwer Academic Publishers, 1985.

[148] Amnon Shashua. A geometric invariant for visual recognition and 3d reconstruction from two perspective/orthographic views. In *Proc. IEEE Workshop on Qualitative Vision*, pages 107–117, 1993.

[149] Christopher E. Smith and Nikolaos P. Papanikolopoulos. Computation of shape through controlled active exploration. In *Proc. IEEE Robotics Automat. Conf.*, pages 2516–2521, 1994.

[150] Tarek M. Sobh, J. Owen, C. Jaynes, M. Dekhil, and T. C. Henderson. Industrial inspection and reverse engineering. In *Proc. Second CAD-Based Vision Workshop*, pages 228–235, 1994.

[151] S. A. Stansfield. A robotic perceptual system utilizing passive vision and active touch. *Int. J. Robotics Res.*, 7(6):138–161, 1988.

[152] J. Ross Stenstrom and C. Ian Connolly. Constructing object models from multiple images. *Int. J. Computer Vision*, 9(3):185–212, 1992.

[153] Kent A. Stevens. The visual interpretation of surface contours. *Artificial Intelligence*, 17:47–73, 1981.

[154] Michael J. Swain and Markus A. Stricker. Promising directions in active vision. *Int. J. Computer Vision*, 11(2):109–126, 1993.

[155] Michael J. Swain (guest editor). Special issue on active vision I. *Int. J. Computer Vision*, 11(2), 1993.

[156] Michael J. Swain (guest editor). Special issue on active vision II. *Int. J. Computer Vision*, 12(2/3), 1994.

[157] Richard Szeliski and Richard Weiss. Robust shape recovery from occluding contours using a linear smoother. In *Proc. Computer Vision and Pattern Recognition Conf.*, pages 666–667, 1993.

[158] Kostantinos Tarabanis, Peter K. Allen, and Roger Y. Tsai. A survey of sensor planning in computer vision. Technical report, IBM T. J. Watson Research Center, 1991.

[159] Kostantinos Tarabanis and Roger Y. Tsai. Computing viewpoints that satisfy optical constraints. In *Proc. Computer Vision and Pattern Recognition*, pages 152–158, 1991.

[160] Michael J. Taylor and Andrew Blake. Grasping the apparent contour. In *Proc. Third European Conf. on Computer Vision*, pages 25–34, 1994.

[161] W. B. Thompson and J. L. Mundy. Three-dimensional model matching from an unconstrained viewpoint. In *Proc. IEEE Robotics Automat. Conf.*, pages 208–220, 1987.

[162] Carlo Tomasi and Takeo Kanade. Shape and motion from image streams under orthography: A factorization method. *Int. J. Computer Vision*, 9(2):137–154, 1992.

[163] R. Y. Tsai. A versatile camera calibration technique for high accuracy 3d machine vision metrology using off-the shelf tv cameras and lenses. *IEEE Trans. Robotics and Automat.*, 3(4):323–344, 1987.

[164] Roger Y. Tsai and Kostantinos Tarabanis. Occlusion free sensor placement planning. In Herbert Freeman, editor, *Machine Vision for Three-Dimensional Scenes*, pages 301–339. Academic Press, 1990.

[165] John K. Tsotsos. On the relative complexity of active vs. passive visual search. *Int. J. Computer Vision*, 7(2):127–141, 1992.

[166] Greg Turk and Marc Levoy. Zippered polygon meshes from range images. In *Proc. SIGGRAPH'94*, pages 311–318, 1994.

[167] Shimon Ullman and Ronen Basri. Recognition by linear combinations of models. *IEEE Trans. on Pattern Anal. and Mach. Intell.*, 13(10):992–1006, 1991.

[168] Fatih Ulupinar and Ramakant Nevatia. Using symmetries for analysis of shape from contour. In *Proc. 2nd Int. Conf. on Computer Vision*, pages 414–426, 1988.

[169] Régis Vaillant. *Geometrie differentielle et vision par ordinateur: Detection et reconstruction des contours d'occultation de la surface d'un objet non-polyedrique*. PhD thesis, L' Universite de Paris-Sud Centre D'Orsay, 1990.

[170] Régis Vaillant and Olivier D. Faugeras. Using extremal boundaries for 3-d object modeling. *IEEE Trans. Pattern Anal. Machine Intell.*, 14(2):157–173, 1992.

[171] Stephen Waldon and Charles R. Dyer. Dynamic shading, motion parallax and qualitative shape. In *Proc. IEEE Workshop on Qualitative Vision*, pages 61–70, 1993.

[172] C. T. C. Wall. Geometric properties of generic differentiable manifolds. In A. Dold and B. Eckmann, editors, *Geometry and Topology*, pages 707–774. Springer-Verlag, 1977.

[173] Daphna Weinshall. Model-based invariants for 3-d vision. *Int. J. Computer Vision*, 10(1):27–42, 1993.

[174] Daphna Weinshall and Carlo Tomasi. Linear and incremental acquisition of invariant shape models from image sequences. In *Proc. 4th Int. Conf. on Computer Vision*, pages 675–682, 1993.

[175] Isaac Weiss. Projective invariants of shapes. In *Proc. Image Understanding Workshop*, pages 1125–1134, 1988.

[176] Lee E. Weiss, Arthur C. Sanderson, and Charles P. Neuman. Dynamic sensor-based control of robots with visual feedback. *IEEE J. Robotics Automat.*, 3(5):404–417, 1987.

[177] P. Whaite and F. P. Ferrie. From uncertainty to visual exploration. *IEEE Trans. Pattern Anal. Machine Intell.*, 13(10):1038–1049, 1991.

[178] P. Whaite and F. P. Ferrie. Uncertain views. In *Proc. Computer Vision and Pattern Recognition*, pages 3–9, 1992.

[179] P. Whaite and F. P. Ferrie. Active exploration: Knowing when we're wrong. In *Proc. Int. Conf. Computer Vision*, pages 41–48, 1993.

[180] P. Whaite and F. P. Ferrie. Autonomous exploration: Driven by uncertainty. In *Proc. Computer Vision and Patter Recognition*, pages 339–346, 1994.

[181] Hassler Whitney. On singularities of mappings of euclidean spaces. i. mappings of the plane into the plane. *Ann. Math.*, 62(3):374–410, 1955.

[182] David Wilkes. *Active Object Recognition.* PhD thesis, University of Toronto, 1993.

[183] David Wilkes and John K. Tsotsos. Active object recognition. In *Proc. Computer Vision and Pattern Recognition*, pages 136–141, 1992.

[184] David Wilkes and John K. Tsotsos. Integration of camera motion behaviors for active object recognition. In *Proc. Workshop on Visual Behaviors*, pages 10–19, 1994.

[185] Lance R. Williams. Topological reconstruction of a smooth manifold-solid from its occluding contour. In *Proc. European Conf. on Computer Vision*, pages 36–47, 1994.

[186] Lambert Wixson. Viewpoint selection for visual search. In *Proc. Computer Vision and Pattern Recognition Conf.*, pages 800–805, 1994.

[187] Lambert E. Wixson and Dana H. Ballard. Using intermediate objects to improve the efficiency of visual search. *Int. J. Computer Vision*, 12(2/3):209–230, 1994.

[188] Seungku Yi, Robert M. Haralick, and Linda G. Shapiro. Automatic sensor and light source placement for machine vision. In *Proc. Int. Conf. on Pattern Recognition*, pages 55–59, 1990.

[189] Billibon H. Yoshimi and Peter K. Allen. Active, uncalibrated visual servoing. In *Proc. IEEE Robotics Automat. Conf*, pages 156–161, 1994.

[190] Zhengyou Zhang and Olivier Faugeras. *3D Dynamic Scene Analysis*. Springer-Verlag, 1992.

[191] ChangSheng Zhao and Roger Mohr. Relative 3d regularized b-spline surface reconstruction through image sequences. In *Proc. 3rd European Conf. on Computer Vision*, pages 417–426, 1994.

[192] Jiang Yu Zheng. Acquiring 3-D models from sequences of contours. *IEEE Trans. Pattern Anal. Machine Intell.*, 16(2):163–178, 1994.

[193] A. Zisserman, A. Blake, C. A. Rothwell, L. J. Van Gool, and M. Van Diest. Eliciting qualitative structure from image curve deformations. In *Proc. 4th Int. Conf. on Computer Vision*, pages 340–345, 1993.