Provable Strategies for Vision-Guided Exploration in Three Dimensions

Kiriakos N. Kutulakos Charles R. Dyer Vladimir J. Lumelsky

Computer Sciences Department
University of Wisconsin
Madison, Wisconsin 53706 USA

Abstract

An approach is presented for exploring an unknown, arbitrary surface in three-dimensional (3D) space by a mobile robot. The main contributions are (1) an analysis of the capabilities a robot must possess and the trade-offs involved in the design of an exploration strategy, and (2) two provably-correct exploration strategies that exploit these trade-offs and use visual sensors (e.g., cameras and range sensors) to plan the robot's motion. No such analysis existed previously for the case of a robot moving freely in 3D space. The approach exploits the notion of the occlusion boundary, *i.e., the points separating the visible from the occluded parts of an object. The occlusion boundary is a collection of curves that "slide" over the surface when the robot's position is continuously controlled, inducing the visibility of surface points over which they slide. The paths generated by our strategies force the occlusion boundary to slide over the entire surface. The strategies provide a basis for integrating motion planning and visual sensing under a common computational framework.*

1 Introduction

An important, unsolved problem in robotics is the development of strategies for real-time and provably-correct exploration of unknown environments where three-dimensional reasoning and visual sensing is necessary. Such exploration strategies are necessary when a mobile robot must control its position in order to reach a desired location in an unknown three-dimensional environment, or to produce a map of it.

The main focus of this paper is the development of strategies for solving the *object exploration* task: In this task the goal of the robot is to plan a collision-free path allowing it to sense all points on the surface of an object in the environment. We assume that the robot is a point and that it is equipped with a camera or a range sensor. We assume that the robot's environment contains objects that are finite volumes bounded by closed and connected surfaces of arbitrary shape.

The support of the National Science Foundation under Grant Nos. IRI-9022608 and IRI-9220782 is greatfully acknowledged.

The major contributions of this paper are (1) the development of a framework for solving the object exploration problem that makes explicit both the capabilities a robot must possess in order to successfully explore an object, and the trade-offs involved in the design of an exploration strategy, and (2) two exploration strategies that exploit these trade-offs and rely on visual sensors (e.g., cameras or range sensors) for planning the motion of the robot. These strategies build on computer vision research that has studied the appearance of smooth and piecewise-smooth surfaces [1, 2]. Our results provide a basis for integrating motion planning and visual sensing under a common computational framework.

Our work has been inspired by recent approaches which, instead of assuming that complete information about the environment is available [3], use only the sensor information that is necessary for planning the motion of the robot [4]. These approaches follow a *purposive* [5], *act-while-thinking* strategy and consider robotic motion planning as a continuous process where sensing and action are tightly coupled [6–8].

In our previous work we focused on the path planning problem, i.e., the task of reaching a location within an unknown three-dimensional environment [7]. That work provides the basis for the work we present here. More specifically, in [7] we obtained three main results:

- In order to solve the path planning problem a robot must, in general, be capable of exploring the surface of an arbitrary object. This result shows the importance of solving the object exploration problem.
- A provably-correct path planning strategy was developed under the assumption that an object exploration strategy is available to the robot. However, an exploration strategy was not presented.
- Through a theoretical analysis of the type of sensing information that is *necessary* and *sufficient* for solving the path planning and exploration problems, we showed that in order to solve these problems a robot must plan its motion based on the perceived shape (i.e., appearance) of the objects and must use sensors that provide information for at least a one-dimensional set of object points

from every position of the robot. This result implies that *visual sensors* (i.e., sensors that provide shape information about the visible portions of object surfaces[1]) are important for planning a robot's motion.

In this paper we develop strategies that guarantee correctness and bounded length paths for exploring arbitrary geometrically-complex, three-dimensional environments. When trying to perform tasks that depend on the appearance of an object (as is the case with three-dimensional exploration and path planning), provable correctness is critical (see Section 2): Object appearance can drastically change depending on the position of the robot, making *ad hoc* strategies unpredictable, incomplete, and even non-terminating

To our knowledge, no strategies currently exist for solving the object exploration problem in non-polyhedral environments when the robot is able to freely move in space (i.e., has three degrees of freedom in position). In the case of polyhedra, the strategies developed (e.g., [9]) rely on the fact that the objects consist of a finite collection of planar faces, and produce paths whose lengths diverge in the limit. A number of recent approaches in computer vision have been suggested for exploring the surfaces of objects [8, 10–12], but their correctness in arbitrary environments has not been investigated.

2 Overview of our Approach

As a robot moves in its environment, the set of visible points changes. In our approach we capture and control these visibility transitions by analyzing the *occlusion boundary* of objects. For a given robot position, this is the collection of closed curves bounding the occluded object points (Section 3). The shape and position of these surface curves almost always depends continuously on the robot's position: When the robot's position is continuously controlled, these curves can be thought of as sliding on the surface. Furthermore, their topology (i.e., connectivity) can change only at a discrete set of robot positions. If an occlusion boundary curve slides over a previously-occluded point during the robot's motion, that point becomes visible. This allows us to formulate exploration as the task of forcing the occlusion boundary to slide over the points occluded from the robot's initial position (Section 3).

Intuitively, the main difficulty in solving the exploration problem is that although the robot has some control over the motion of the occlusion boundary over the surface, this control is not complete; the motion of the occlusion boundary also depends on the shape of the surface itself. In addition, the topology of the occlusion boundary can change unexpectedly, further complicating the exploration process.

[1] Visible portions of an object surface contain all points for which the open line segment connecting them to the robot does not intersect any object. In our discussion, "visual sensors" include cameras, where only information about the *projected* shape of the visible surfaces is directly available, as well as more powerful mechanisms such as range sensors, which directly provide distances to visible surface points within a finite ball around the robot.

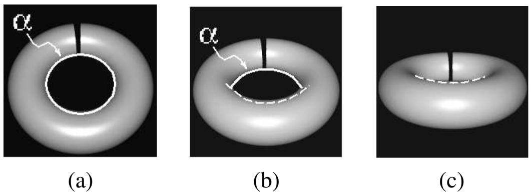


Figure 1: Forcing the occlusion boundary curve α, shown as a white circle in the first image, to slide over the dark curve drawn on the torus.

To illustrate the difficulties involved in exploring an object, consider the torus shown in Figure 1(a). In order to explore the torus, the robot must force the visibility of all points along the dark curve drawn on its surface. One way to proceed is to move downward, forcing the smooth curve α, which is part of the occlusion boundary, to slide over the segment of the dark curve that is initially occluded. As the robot moves downward, however, α shrinks to a point and disappears (Figure 1(c)), changing the topology of the occlusion boundary and making any further downward motion of the robot ineffective. The robot must now move differently in order to continue the exploration process. Similar difficulties occur due to geometrical changes in the occlusion boundary (i.e., even when no topological changes of the occlusion boundary occur).

This simple example illustrates that a number of basic steps are necessary to solve the exploration problem. It is therefore necessary to ask how many times such actions need to be executed, whether the whole surface is always explored, and whether the exploration process is guaranteed to terminate. These issues are precisely the reasons why provably-correct exploration strategies are necessary: Since the answers to these questions are not evident even for geometrically-simple surfaces such as the torus, provably-correct exploration strategies are necessary if one hopes to use them for exploring real objects.

To explore the surface of an object we present an iterative strategy (Section 4) that repeatedly performs a sequence of basic motions in order to incrementally "grow" the set of points made visible on an object surface. These motions are designed in a way that guarantees (1) they will be executed only a finite number of times, and (2) when the process terminates the surface will be completely explored.

Clearly, the robot's motions at each iteration of the exploration process depend on the robot's capabilities and the sensor

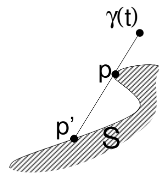


Figure 2: Points p and p' belong to the occlusion boundary. Only point p belongs to the visible rim.

input. These motions determine the class of objects that the robot can provably explore. To study the trade-offs involved in the design of an exploration strategy we consider two different types of basic motions. The first type uses a sensor capable of providing the three-dimensional coordinates of all visible points (e.g., a range sensor), and requires the robot to move very close to an object at every iteration (Section 5). With these capabilities we show that arbitrary objects can be explored. The second type of basic motions uses a camera that provides only the shape of the *projection* of the occlusion boundary, and requires the robot to get close to the surface only when the surface geometry makes this necessary (Section 6). In this case we show that the resulting exploration strategy can only be used to explore non-concave objects or the non-concave regions of a general object. We then briefly discuss the trade-offs in these two exploration strategies, and also describe how they can be combined (Section 7).

3 Visibility and Occlusion

In order to solve the exploration problem in an unknown environment one must understand the structure of the occlusion boundary, its relation to object geometry, and the way its structure changes when the robot moves. This section presents the basic concepts required for this analysis.

Without loss of generality, we consider in rest of the paper the case where the environment contains a single object whose surface S is connected, has finite area, and bounds an open volume V_S in $\Re^3$. Furthermore, we limit our discussion to the case where S is smooth; for these surfaces every surface point p has a uniquely-defined outward surface normal, $\mathbf{n}(p)$, and tangent plane, $T_p(S)$, both of which depend smoothly on p.[2]

Given a robot path $\gamma(t) \in \Re^3 - V_S, 0 \leq t \leq T$, a point $p \in S$ is *occluded* from position $\gamma(t)$ if and only if the line segment $p\gamma(t)$ intersects V_S; otherwise p is visible.

[2] The strategies we present can be extended to handle piecewise-smooth surfaces. These surfaces are smooth except for zero- or one-dimensional sets whose points have a finite number of normals assigned to them.

Definition 3.1 The *visible rim*, $\mathfrak{R}_{\gamma(t)}$, is the set of points p for which the line segment $p\gamma(t)$ is tangent to S at p and does not intersect V_S.

Definition 3.2 The *occlusion boundary*, $\mathfrak{O}_{\gamma(t)}$, bounds the visible and the occluded points. It is the set of points p for which the line segment $p\gamma(t)$ contains at least one visible rim point and does not intersect V_S.

Clearly, the occlusion boundary is a superset of the visible rim (Figure 2). A convenient way to visualize and study the occlusion boundary and the visible rim is to look at their spherical projections onto an image centered at the robot's position (Figure 3). Their projections are identical; this projected contour is known as the *occluding contour* [1].

In general, the occlusion boundary is collection of closed, piecewise-smooth curves, $\alpha_1, \ldots, \alpha_n$, whose number and shape depends on the robot's position. The endpoints of the smooth segments of these curves project to cusps or T-junctions on the occluding contour (Figure 3). Given a smooth path $\gamma(t), 0 < t < T$, for the robot, the curves α_i move on the surface in a way that depends smoothly on t, except for a discrete set of values for which their topology changes. These changes occur precisely when the topology of the occluding contour changes [13–15].

Between any two consecutive topological transitions the occlusion boundary can be thought of as a collection of smooth curves that "slide" over the surface of the object as the robot moves, causing deformations to the occluding contour. In particular, given an instantaneous direction of motion, $\gamma'(t)$, we can express the motion of the occlusion boundary using the *epipolar parameterization*. The details of this parameterization are not important here and the reader is referred to [16]. The important point is that given a smooth segment $\beta(t)$ of

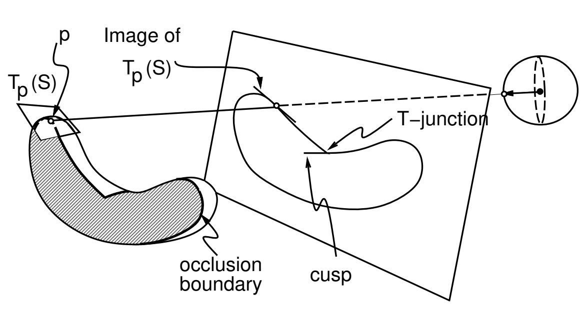


Figure 3: The occluding contour. Each point p on the occlusion boundary projects to a single point on the spherical image. For simplicity, the image of the occlusion boundary on a plane tangent to the spherical image is shown.

$\mathfrak{O}_{\gamma(t)}$, this parameterization allows us to define the segment $\beta(t+\delta t)$ of $\mathfrak{O}_{\gamma(t+\delta t)}$ that corresponds to $\beta(t)$.

The motion of the occlusion boundary induced by the robot's motion directly affects the set of surface points becoming visible. When the occlusion boundary at time T slides over a surface point that was occluded from positions $\gamma(t), 0 \leq t < T$, that point becomes visible. The following theorem (see [17] for a proof) gives a qualitative characterization of these visibility transitions (Figure 4):

Theorem 3.1 *Let $p \in \mathfrak{O}_{\gamma(t)}$ and let q_p be a point of tangency of the segment $p\gamma(t)$ with S. If $\beta(t)$ is a smooth occlusion boundary segment whose points p satisfy $\mathbf{n}(q_p) \cdot \gamma'(t) > 0$, all points on $\beta(t+\delta t)$ are occluded from position $\gamma(t)$. Conversely, if $\mathbf{n}(q_p) \cdot \gamma'(t) < 0$, all points on $\beta(t+\delta t)$ are visible from position $\gamma(t)$.*

Once a point becomes visible, it may be possible to recover its three-dimensional coordinates using a visual sensor. We consider two types of sensors:

- *Range sensors*. With such a sensor the robot can determine the coordinates of all visible points.
- *Cameras*. With such a sensor the robot can detect the occluding contour. Furthermore, when the robot moves along a path γ and knows its speed and acceleration, it can determine the coordinates of all points on the visible rim from the deformation of the occluding contour [16].

The exploration problem can now be formulated as follows:

Definition 3.3 (Range sensor-based exploration) Given a connected surface S and an initial position $\mathbf{s} \in \Re^3 - V_S$, generate a finite-length path γ such that the occlusion boundary slides over all points on S that were occluded at $\mathbf{s}$. In other words, if U is the set of points occluded at $\mathbf{s}$ and $\gamma = \{\gamma(t), 0 \leq t \leq T\}$, γ must satisfy $\bigcup_{0 \leq t \leq T} \mathfrak{O}_{\gamma(t)} \supseteq U$.

Definition 3.4 (Camera-based exploration) Given a connected surface S and an initial position $\mathbf{s} \in \Re^3 - V_S$, generate a finite-length path γ such that if $\gamma = \{\gamma(t), 0 \leq t \leq T\}$, γ must satisfy $\bigcup_{0 \leq t \leq T} \mathfrak{R}_{\gamma(t)} = S$.

To facilitate our discussion we use the following definition:

Definition 3.5 A point p on S is *explored* if and only if the three-dimensional coordinates of p were determined by the robot from some position along its path. The boundary of the explored points is the *exploration frontier*.

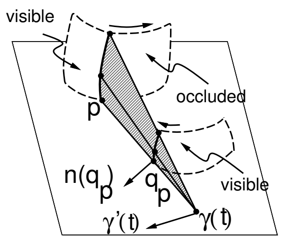


Figure 4:

4 Incremental Exploration

Having provided a formulation for the exploration problem, this section considers a strategy for exploring an object of arbitrary shape. The idea is to control the robot's motion so that the set of explored points incrementally "grows" on the surface. To achieve this objective, the robot performs the following actions during the n-th iteration: It selects a point p_n on the exploration frontier, moves to a position where it is visible, and finally performs a sequence of motions forcing all points in a neighborhood of p_n to be explored. This process terminates when the set of explored points is identical to the object's surface, i.e., when the exploration frontier degenerates to a collection of isolated points.

To completely specify our exploration strategy we need to answer four questions: (1) How to select point p_n, (2) how to find a position where p_n is visible, (3) how to explore a neighborhood of p_n, and (4) how to guarantee that only a finite number of iterations are necessary. The crucial point in our approach is that the third and fourth questions, which lie at the heart of the exploration problem, can be answered using a *local* strategy, i.e., one that does not depend on the history of the exploration process (the set of points explored and the path generated in the previous iterations). We discuss each of the four questions below.

In our approach, the point p_n is simply chosen to be any point on the exploration frontier. The coordinates of the points comprising the frontier are available to the robot because the exploration frontier bounds the set of explored points.

The answer to the second question is also simple. If the robot's sensor is a camera, all points on the exploration frontier were points on the visible rim at some previous position of the robot; if it is a range-sensor, all points on the exploration frontier were points on the occlusion boundary at some previous position. Hence, it suffices for the robot to retrace its path back to a position where the selected point p_n was on the occlusion boundary or the visible rim. This can be achieved if, along its path, the robot saves its current position together with the set of points belonging to the visible rim (or the occlusion boundary, if a range sensor is used) at that position.[3] This information allows the robot to (1) associate

[3] It is, in fact, possible to show that the coordinates of only a discrete set of occlusion boundary or visible rim points need to be stored from each position.

with each exploration frontier point a set of previous positions from which that point is visible, and (2) plan a collision-free path to any such position.

Now suppose that the selected point is visible. We use a *neighborhood exploration strategy* to answer the third question. The only input to this strategy is the selected point p_n. The goal is to generate a path, based on information provided by the robot's sensor, such that an open neighborhood W_n of p_n becomes explored.

In order to explore the surface in a finite amount of time, the neighborhood W_n must be large enough so that only a finite number of such neighborhoods is necessary to cover the whole surface of the object.

Definition 4.1 Given a class of surfaces $\mathcal{S}$, a neighborhood exploration strategy is *complete for* $\mathcal{S}$ if and only if for all surfaces $S \in \mathcal{S}$ and all sequences $(p)_n, p_n \in \partial\left(\bigcup_{k\leq n} W_k\right)$, the sequence $(W)_n$ of explored neighborhoods is such that $\bigcup_{n\leq K} W_n = S$ for some finite K.

Equipped with a complete exploration strategy $\mathcal{A}$ for $\mathcal{S}$, the robot uses the following strategy to explore a surface $S \in \mathcal{S}$:

Surface Exploration Strategy

Step 1: If a range sensor is available, select a point p on the occlusion boundary; otherwise select p on the visible rim. (Initially the exploration frontier is identical to the occlusion boundary.)

Step 2: Use $\mathcal{A}$ to explore a neighborhood of p. During this step, the set of points explored on the surface is expanded.

Step 3: Select any point p' on the current exploration frontier. If no such point exists, stop; the exploration process is complete.

Step 4: If a range sensor is available, move to a previous position, $\gamma(t_0)$ at which p' is on the occlusion boundary; otherwise move to a previous position $\gamma(t_0)$ where p' is on the visible rim.

Step 5: Repeat steps 2-5.

The only step left unspecified in the above strategy is Step 2. The next two sections describe two neighborhood exploration strategies that can be used to implement this step. These strategies give rise to two different solutions to the exploration problem. The first strategy, $\mathcal{A}_r$, which is the most general, uses a range sensor. It is complete for the class of smooth and connected surfaces of finite area (Section 5). The second one, $\mathcal{A}_c$, uses a camera and is complete only for the subclass of surfaces with no concavities, e.g., the torus or, more generally, surfaces produced by appropriately deforming spheres with h handles (Section 6).

5 Exploration Using a Range Sensor

Suppose that the robot needs to explore the neighborhood of a selected occlusion boundary point p_n on S with the help of a range sensor. Strategy $\mathcal{A}_r$ is based on the following observation: If the robot looks at p_n from the direction of its surface normal and is positioned close enough to p_n, all points in a neighborhood of p_n must be visible. Intuitively, the reason for this is that the smoothness of S around p_n severely constrains the variation of the surface normals in the point's neighborhood; this limits the sets of points that can be occluded when the robot is close to p_n along $\mathbf{n}(p_n)$.

Using this observation, strategy $\mathcal{A}_r$ prescribes exactly how the robot should move to reach a position along $\mathbf{n}(p_n)$, while at the same time guaranteeing the completeness of the strategy for arbitrary smooth surfaces. In particular, the strategy specifies (1) how to reach positions along $\mathbf{n}(p_n)$, and (2) how far away from p_n the robot should go.

Reaching a position close to p_n along $\mathbf{n}(p_n)$ is trivial. Since p_n is visible from the robot's initial position, the robot moves along a straight line towards p_n, determines the surface normal at p_n when p_n is reached, and then moves along that normal.

The following lemma shows that after p_n is reached, the robot can guarantee that a neighborhood of p_n is explored by performing an arbitrarily small position adjustment along $\mathbf{n}(p_n)$. See [18] for the proof.

Lemma 5.1 *Let $\mathcal{E}(d)$ be the set of visible points when the robot is at distance d from p_n along $\mathbf{n}(p_n)$. Then, there is an $\epsilon > 0$ such that $\mathcal{E}(d)$ contains a neighborhood of p_n for all $0 < d < \epsilon$.*

Unfortunately, an arbitrarily small motion away from p_n can leave $\mathcal{E}(d)$ arbitrarily small. The robot must therefore move sufficiently far from p_n so that $\mathcal{E}(d)$ contains a neighborhood of p_n that is large enough to guarantee completeness. In [18] we show that the robot can guarantee completeness by simply moving away from p_n along $\mathbf{n}(p_n)$ until either a collision with the surface does not allow further motion, or an *a priori* defined distance $\Delta > 0$ is reached. Furthermore, the selection of Δ does not affect the completeness of the strategy. This result follows from the smoothness of S, which ensures that if the robot moves away from p_n along $\mathbf{n}(p_n)$, the distance traveled before a surface collision occurs cannot be arbitrarily small.

The above strategy can be improved by adding an additional stopping condition that reduces the amount of robot motion away from p_n. This condition uses Theorem 3.1 and requires the robot to monitor the occlusion boundary while moving away from p_n. In particular, during the robot's motion away from p_n its instantaneous velocity, $\gamma'(t)$, is along $\mathbf{n}(p_n)$. The occlusion boundary curves at any time t during this motion can be partitioned into open segments falling into two categories,

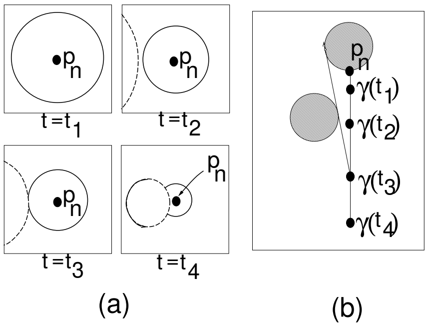


Figure 5: Executing strategy $\mathcal{A}_r$. (a) Views of an environment with two spheres, as the robot moves away from point p_n on the sphere shown in the first image, along $\mathbf{n}(p_n)$. The solid curves correspond to occlusion boundary points in B^+ and the dashed curves to points in B^-. Condition (c) in Step 3 forces the robot to stop at time t_3. (b) Top view of a horizontal plane containing p_n and $\mathbf{n}(p_n)$.

B^+ and B^-, whose points r satisfy $\gamma'(t) \cdot r > 0$ and $\gamma'(t) \cdot r < 0$, respectively. If d^+, d^- are the distances of B^+ and B^- from p_n, respectively, the robot can stop moving when $d^- < d^+$ (Figure 5).

Intuitively, this constraint allows the robot to move away from p_n as long as the set of visible points near p_n strictly expands. To see this, note that according to Theorem 3.1, the segments in B^+ slide over previously-occluded points on S, expanding the set of points that are visible in p_n's neighborhood, while the segments in B^- cause a contraction of this set. The distance $d = \min\{d^+, d^-\}$ determines the size of the visible neighborhood of p_n, and increases monotonically as long as $d = d^+$.

Given that p_n is on the occlusion boundary from the robot's position, $\gamma(t_0)$, the following neighborhood exploration strategy can now be specified:

Strategy $\mathcal{A}_\mathbf{r}$

Step 1: Starting from $\gamma(t_0)$, move on a straight line to p_n.

Step 2: Determine the surface normal at p_n, $\mathbf{n}(p_n)$.

Step 3: Move in direction $\mathbf{n}(p_n)$, until one of the following conditions is satisfied:

- **a.** The surface obstructs the robot's motion.
- **b.** An *a priori* defined distance Δ from p_n is reached.
- **c.** $d^- < d^+$, where d^+, d^- are the distances of B^+ and B^- to p_n, respectively.

The following theorem establishes the correctness of the strategy. The interested reader is referred to [18] for the proof. It is based on the fact that the surface is smooth and does not have infinite curvature at p_n.

Theorem 5.1 *Strategy $\mathcal{A}_r$ is complete for smooth and connected surfaces of finite area.*

6 Exploration Using a Camera

This section considers an alternative strategy called $\mathcal{A}_c$. The strategy has three characteristics: (1) It uses a camera instead of a range sensor, (2) it moves the robot close to the selected point p_n only when this is dictated by the shape of the object's surface, and (3) it is complete for a subset of the surfaces handled by strategy $\mathcal{A}_r$, namely those that are non-concave and generic.[4] When the robot's sensor can provide information only about the shape of the occluding contour, $\mathcal{A}_c$ allows the exploration of almost all surfaces that can be explored with such a sensor: It is easy to see that points in a surface concavity cannot belong to the visible rim, and hence cannot become explored by *any* strategy using only information derived from the occluding contour.

Strategy $\mathcal{A}_c$ is based on the following simple observation: Suppose the robot is positioned level with a point on a hill top; then this point will belong to the visible rim. By moving up or down, the robot will see just over the hill or not quite to the top, respectively. These motions cause the visible rim to slide over a neighborhood of that point, according to Theorem 3.1. In general, this motion corresponds to motion on the plane Π defined by the selected point p_n, the surface normal $\mathbf{n}(p_n)$, and the robot's position [17]. Strategy $\mathcal{A}_c$ uses this observation to explore a neighborhood W_n of p_n while at the same time guaranteeing that W_n is large enough to ensure completeness.

To achieve this, consider why W_n can be arbitrarily small. There are two possibilities:

- The size of W_n depends on the length of the visible rim curve containing p_n and the distance of its endpoints from p_n, if such endpoints exist. Depending on the robot's position, c, these lengths can be arbitrarily small; they can also become arbitrarily small because of a topological change in the visible rim when c is infinitesimally perturbed on Π.
- The size of W_n is constrained by the amount of robot motion on Π.

[4] A precise definition of generic surfaces is beyond the scope of this paper, and the reader is referred to [19]. Intuitively, generic surfaces are surfaces whose geometrical characteristics do not change if they are infinitesimally perturbed. All smooth surfaces (except for a zero-measure set) are generic.

The main idea of strategy $\mathcal{A}_c$ is to exploit the following result, which ensures the length of the visible rim curve containing p_n never becomes arbitrarily small:

Proposition 6.1 *Let S be a smooth, non-concave, generic surface. Given a sequence of points, $(p)_n \in S$, and a sequence of positions, $(c)_n$, from which p_n is on the visible rim, there exists a sequence of positions, $(\hat{c})_n$, satisfying the following four conditions:*

1. *$\hat{c}_n$ is on $T_{p_n}(S)$ and is reachable from c_n.*
2. *p_n is visible from $\hat{c}_n$.*
3. *There exists an $\epsilon_1 > 0$ such that the length of the visible rim curve, β_n, containing p_n at c_n is larger than ϵ_1 for all n.*
4. *There exists an $\epsilon_2 > 0$ such that for all n and all infinitesimal perturbations, c, of $\hat{c}_n$ on the plane containing p_n and $\mathbf{n}(p_n)$, the distance between the endpoints of the visible rim curve containing p_n at c is greater than ϵ_2.*

Given a point p_n, $\mathcal{A}_c$ determines a set of positions, $\Sigma_n = \{\hat{c}_n^1, \dots, \hat{c}_n^{k_n}\}$, such that at least one of the resulting sequences $(\hat{c})_n^i$ satisfies Proposition 6.1 (Section 6.1).

Once Σ_n is determined, ensuring that W_n never diminishes is fairly simple. If Π is the plane defined by $\hat{c}_n^i$, p_n, and $\mathbf{n}(p_n)$, it suffices for the robot to move on a circle in Π centered at p_n of radius $\|\hat{c}_n^i - p_n\|$. Because the robot can collide with the surface, this motion also involves moving closer to p_n along a curve in the intersection $S \cap \Pi$, if such a collision occurs. Due to lack of space a description of this strategy is omitted. The interested reader is referred to [18] for the details.

Given that p_n is on the visible rim from the robot's position, $\gamma(t_0)$, the following neighborhood exploration strategy can now be specified:

Strategy $\mathcal{A}_\mathbf{c}$

Step 1: Starting from $\gamma(t_0)$, plan a path to determine a set of positions $\Sigma_n = \{\hat{c}_n^1, \dots, \hat{c}_n^{k_n}\}$ on $T_{p_n}(S)$, such that for at least one $i, i = 1, \dots, k_n$, $(\hat{c})_n^i$ satisfies Proposition 6.1.

Step 2: For each $i = 1, \dots, k_n$, (a) move to $\hat{c}_n^i$, and (b) starting from $\hat{c}_n^i$, move in the plane defined by $\hat{c}_n^i, p_n$, and $\mathbf{n}(p_n)$ to explore a neighborhood of p_n.

Theorem 6.1 *Strategy $\mathcal{A}_c$ is complete for smooth, non-concave, generic, and connected surfaces of finite area.*

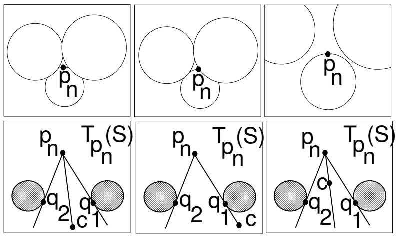


Figure 6: *Top row:* Views of three spheres from three positions on $T_{p_n}(S)$. *Bottom row:* Top views of $T_{p_n}(S)$. Shaded areas correspond to the intersection of $T_{p_n}(S)$ with the three spheres. Each view in the top row corresponds to the robot position c in the view of $T_{p_n}(S)$ below it.

6.1 Positioning the Robot on $\mathbf{T_{p_n}(S)}$

This section briefly describes how to generate a path, starting from an initial position $\gamma(t_0)$, that allows determination of the set Σ_n. The main idea in this process is that every position, $\hat{c}_n^i \in \Sigma_n$, can be characterized by two geometric relationships: (1) The angle formed by segment $\hat{c}_n^i p_n$ and the *asymptotes* [20] and *bitangents*[5] at p_n, and (2) the distance between $\hat{c}_n^i$ and p_n. In the following we show how these two relationships can be used to determine Σ_n. This essentially reduces the determination of Σ_n to the determination of the bitangents and asymptotes at p_n. The following provides an intuitive explanation for the selection process through an example; for formal proofs and more details see [17, 18].

Suppose p_n is on the visible rim when the robot is positioned at c. The angle between segment cp_n and the asymptotes and bitangents affects the distance between p_n and the endpoints of the visible rim curve containing it. To see this, consider the example in Figure 6. By moving the robot close to the bitangent line through p_n and q_1 (without changing its distance to p_n), the right T-junction endpoint of the visible rim curve containing p_n can get arbitrarily close to p_n. To avoid this, c must be such that the segment cp_n forms a large angle with both bitangent lines through p_n.

Similarly, the distance between c and p_n affects the length of the visible rim curve containing p_n. If the robot moves close enough to p_n in the example of Figure 6, the sphere containing p_n will not be occluded by the other two spheres. On the other hand, moving too close to p_n would cause the visible rim curve containing p_n to be arbitrarily short. This implies that $\hat{c}_n^i$ should be closer to p_n than either q_1 or q_2, but

[5] The bitangents at p_n are the lines that are tangent to S at at least two points, one of which is p_n.

should not be arbitrarily close to p_n.

A generalization of the above for the case of a general smooth, non-concave and generic surface S leads to the following result. We only present it here for the case where p_n is elliptic; a similar result holds for the case of parabolic and hyperbolic points.

Theorem 6.2 *Let $B_1, \ldots, B_{k_n}$ be the bitangent half-lines with origin p_n on $T_{p_n}(S)$. Let $q_i^1, \ldots, q_i^{l_i}$ be the points of tangency of B_i with S for which the segments $q_i^l p_n (l = 1, \ldots, l_i)$ do not intersect V_S. Then, we can construct a set Σ_n of positions that is uniquely determined by the half-lines B_i, the distances $\|q_i^l - p_n\|$, and an* a priori *defined constant $M > 0$, and contains at least one position $\hat{c}_n^i$ for which the sequence $(\hat{c})_n^i$ satisfies Proposition 6.1. Furthermore, the size of Σ_n is at most $k_n + 4\sum_{i=1}^{k_n} l_i$.*

Therefore, to construct Σ_n, it suffices to determine the bitangents at p_n and their contacts with S that are visible from p_n. This can be provably achieved by circumnavigating p_n on $T_{p_n}(S)$, until the robot's initial position, $\gamma(t_0)$ is reached, while maintaining the visibility of p_n [17, 18]. During this process, the robot moves closer to p_n only to maintain the visibility of p_n, or when surface collisions prevent the robot from maintaining a fixed distance from p_n.

7 Discussion

The two strategies developed in the previous sections demonstrate that a number of trade-offs are involved when deciding on a strategy to explore an object. Information about object geometry (e.g., whether or not concavities exist) allows the robot to use simpler sensors and allows the robot to stay farther away from the object being explored.

An even more general consequence that follows from our analysis is that the robot can explore different parts of the *same* object in different ways. When multiple strategies are available, a crucial issue is how to decide which strategy to use.

As an example, consider how strategies $\mathcal{A}_r$ and $\mathcal{A}_c$ can be combined. The robot can use a camera to explore from a distance any connected, non-concave region of a surface, even though that surface may also contain concavities. This is because the convergence properties of $\mathcal{A}_c$ change only when the robot attempts to explore the surface near the boundary of such a region [17]: When the robot uses $\mathcal{A}_c$ to explore the surface near a parabolic curve bounding a surface concavity, the set of points explored near the parabolic curve diminishes as the parabolic curve is approached. Consequently, the robot can keep using $\mathcal{A}_c$ until the *rate* of points explored (e.g., the area of newly-explored points) falls below an *a priori* defined threshold. Within such a framework, the robot will have to use $\mathcal{A}_r$ only when necessary, i.e., when exploring close to concavities. An important future direction of our research will be to develop a better understanding of the interaction between these and other exploration strategies.

References

[1] J. J. Koenderink, *Solid Shape*. MIT Press, 1990.

[2] J. H. Rieger, "On the classification of views of piecewise smooth objects," *Image and Vision Computing*, vol. 5, no. 2, pp. 91–97, 1987.

[3] J. H. Reif, "Complexity of the mover's problem and generalizations," in *Proc. Twentieth Symposium on Foundations of Computer Science*, pp. 421–427, 1979.

[4] R. A. Brooks, "A robust layered control system for a mobile robot," *IEEE J. Robotics Automat.*, vol. 2, no. 1, pp. 14–23, 1986.

[5] Y. Aloimonos, "Purposive and qualitative active vision," in *Proc. Int. Conf. on Pattern Recognition*, pp. 346–360, 1990.

[6] V. J. Lumelsky, S. Mukhopadhyay, and K. Sun, "Dynamic path planning in sensor-based terrain acquisition," *IEEE Trans. Robotics Automat.*, vol. 6, no. 4, pp. 462–472, 1990.

[7] K. N. Kutulakos, V. J. Lumelsky, and C. R. Dyer, "Vision-guided exploration: A step toward general motion planning in three dimensions," in *Proc. IEEE Robotics Automat. Conf.*, pp. 289–296, 1993.

[8] A. Blake, A. Zisserman, and R. Cipolla, "Visual exploration of free-space," in *Active Vision* (A. Blake and A. Yuille, eds.), pp. 175–188, MIT Press, 1992.

[9] N. S. Rao, S. S. Iyengar, B. J. Oommen, and R. Kashyap, "Terrain acquisition by point robot amidst polyhedral obstacles," in *Proc. Third Conf. on Artificial Intelligence Applications*, pp. 170–175, 1987.

[10] C. I. Connoly, "The determination of next best views," in *Proc. IEEE Robotics Automat. Conf.*, pp. 432–435, 1985.

[11] J. Maver and R. Bajcsy, "Occlusions as a guide for planning the next view," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, no. 5, pp. 417–433, 1993.

[12] P. Whaite and F. P. Ferrie, "From uncertainty to visual exploration," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, no. 10, pp. 1038–1049, 1991.

[13] J. J. Koenderink and A. J. van Doorn, "The singularities of the visual mapping," *Biological Cybernetics*, vol. 24, pp. 51–59, 1976.

[14] J. Ponce, S. Petitjean, and D. Kriegman, "Computing exact aspect graphs of curved objects: Algebraic surfaces," in *Proc. Second European Conference on Computer Vision*, 1992.

[15] J. H. Rieger, "The geometry of view space of opaque objects bounded by smooth surfaces," *Artificial Intelligence*, vol. 44, pp. 1–40, 1990.

[16] R. Cipolla and A. Blake, "Surface shape from the deformation of apparent contours," *Int. J. Computer Vision*, vol. 9, no. 2, pp. 83–112, 1992.

[17] K. N. Kutulakos and C. R. Dyer, "Global surface reconstruction by purposive control of observer motion," Tech. Rep. 1141, Computer Sciences Department, University of Wisconsin— Madison, April 1993. Available via ftp from ftp.cs.wisc.edu.

[18] K. N. Kutulakos, V. J. Lumelsky, and C. R. Dyer, "Provable strategies for vision-guided exploration in three dimensions," tech. rep., Computer Sciences Department, University of Wisconsin— Madison, 1994. To appear.

[19] Y. L. Kergosien, "La famille des projections orthogonales d'une surface et ses singularités," *C. R. Acad. Sc. Paris*, vol. 292, pp. 929–932, 1981.

[20] M. P. D. Carmo, *Differential Geometry of Curves and Surfaces*. Englewood Cliffs, NJ: Prentice-Hall Inc., 1976.

Provable Strategies for Vision-Guided Exploration in Three Dimensions

Kiriakos N. Kutulakos Charles R. Dyer Vladimir J. Lumelsky

Computer Sciences Department
University of Wisconsin
Madison, Wisconsin 53706 USA

Abstract

An approach is presented for exploring an unknown, arbitrary surface in three-dimensional (3D) space by a mobile robot. The main contributions are (1) an analysis of the capabilities a robot must possess and the trade-offs involved in the design of an exploration strategy, and (2) two provably-correct exploration strategies that exploit these trade-offs and use visual sensors (e.g., cameras and range sensors) to plan the robot's motion. No such analysis existed previously for the case of a robot moving freely in 3D space. The approach exploits the notion of the occlusion boundary*, i.e., the points separating the visible from the occluded parts of an object. The occlusion boundary is a collection of curves that "slide" over the surface when the robot's position is continuously controlled, inducing the visibility of surface points over which they slide. The paths generated by our strategies force the occlusion boundary to slide over the entire surface. The strategies provide a basis for integrating motion planning and visual sensing under a common computational framework.*

1 Introduction

An important, unsolved problem in robotics is the development of strategies for real-time and provably-correct exploration of unknown environments where three-dimensional reasoning and visual sensing is necessary. Such exploration strategies are necessary when a mobile robot must control its position in order to reach a desired location in an unknown three-dimensional environment, or to produce a map of it.

The main focus of this paper is the development of strategies for solving the *object exploration* task: In this task the goal of the robot is to plan a collision-free path allowing it to sense all points on the surface of an object in the environment. We assume that the robot is a point and that it is equipped with a camera or a range sensor. We assume that the robot's environment contains objects that are finite volumes bounded by closed and connected surfaces of arbitrary shape.

The support of the National Science Foundation under Grant Nos. IRI-9022608 and IRI-9220782 is greatfully acknowledged.

The major contributions of this paper are (1) the development of a framework for solving the object exploration problem that makes explicit both the capabilities a robot must possess in order to successfully explore an object, and the trade-offs involved in the design of an exploration strategy, and (2) two exploration strategies that exploit these trade-offs and rely on visual sensors (e.g., cameras or range sensors) for planning the motion of the robot. These strategies build on computer vision research that has studied the appearance of smooth and piecewise-smooth surfaces [1, 2]. Our results provide a basis for integrating motion planning and visual sensing under a common computational framework.

Our work has been inspired by recent approaches which, instead of assuming that complete information about the environment is available [3], use only the sensor information that is necessary for planning the motion of the robot [4]. These approaches follow a *purposive* [5], *act-while-thinking* strategy and consider robotic motion planning as a continuous process where sensing and action are tightly coupled [6–8].

In our previous work we focused on the path planning problem, i.e., the task of reaching a location within an unknown three-dimensional environment [7]. That work provides the basis for the work we present here. More specifically, in [7] we obtained three main results:

- In order to solve the path planning problem a robot must, in general, be capable of exploring the surface of an arbitrary object. This result shows the importance of solving the object exploration problem.
- A provably-correct path planning strategy was developed under the assumption that an object exploration strategy is available to the robot. However, an exploration strategy was not presented.
- Through a theoretical analysis of the type of sensing information that is *necessary* and *sufficient* for solving the path planning and exploration problems, we showed that in order to solve these problems a robot must plan its motion based on the perceived shape (i.e., appearance) of the objects and must use sensors that provide information for at least a one-dimensional set of object points

from every position of the robot. This result implies that *visual sensors* (i.e., sensors that provide shape information about the visible portions of object surfaces[1]) are important for planning a robot's motion.

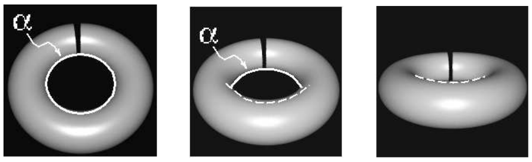
In this paper we develop strategies that guarantee correctness and bounded length paths for exploring arbitrary geometrically-complex, three-dimensional environments. When trying to perform tasks that depend on the appearance of an object (as is the case with three-dimensional exploration and path planning), provable correctness is critical (see Section 2): Object appearance can drastically change depending on the position of the robot, making *ad hoc* strategies unpredictable, incomplete, and even non-terminating

To our knowledge, no strategies currently exist for solving the object exploration problem in non-polyhedral environments when the robot is able to freely move in space (i.e., has three degrees of freedom in position). In the case of polyhedra, the strategies developed (e.g., [9]) rely on the fact that the objects consist of a finite collection of planar faces, and produce paths whose lengths diverge in the limit. A number of recent approaches in computer vision have been suggested for exploring the surfaces of objects [8, 10–12], but their correctness in arbitrary environments has not been investigated.

2 Overview of our Approach

As a robot moves in its environment, the set of visible points changes. In our approach we capture and control these visibility transitions by analyzing the *occlusion boundary* of objects. For a given robot position, this is the collection of closed curves bounding the occluded object points (Section 3). The shape and position of these surface curves almost always depends continuously on the robot's position: When the robot's position is continuously controlled, these curves can be thought of as sliding on the surface. Furthermore, their topology (i.e., connectivity) can change only at a discrete set of robot positions. If an occlusion boundary curve slides over a previously-occluded point during the robot's motion, that point becomes visible. This allows us to formulate exploration as the task of forcing the occlusion boundary to slide over the points occluded from the robot's initial position (Section 3).

Intuitively, the main difficulty in solving the exploration problem is that although the robot has some control over the motion of the occlusion boundary over the surface, this control is not complete; the motion of the occlusion boundary also depends on the shape of the surface itself. In addition, the topology of the occlusion boundary can change unexpectedly, further complicating the exploration process.



(a) (b) (c)

Figure 1: Forcing the occlusion boundary curve α, shown as a white circle in the first image, to slide over the dark curve drawn on the torus.

To illustrate the difficulties involved in exploring an object, consider the torus shown in Figure 1(a). In order to explore the torus, the robot must force the visibility of all points along the dark curve drawn on its surface. One way to proceed is to move downward, forcing the smooth curve α, which is part of the occlusion boundary, to slide over the segment of the dark curve that is initially occluded. As the robot moves downward, however, α shrinks to a point and disappears (Figure 1(c)), changing the topology of the occlusion boundary and making any further downward motion of the robot ineffective. The robot must now move differently in order to continue the exploration process. Similar difficulties occur due to geometrical changes in the occlusion boundary (i.e., even when no topological changes of the occlusion boundary occur).

This simple example illustrates that a number of basic steps are necessary to solve the exploration problem. It is therefore necessary to ask how many times such actions need to be executed, whether the whole surface is always explored, and whether the exploration process is guaranteed to terminate. These issues are precisely the reasons why provably-correct exploration strategies are necessary: Since the answers to these questions are not evident even for geometrically-simple surfaces such as the torus, provably-correct exploration strategies are necessary if one hopes to use them for exploring real objects.

To explore the surface of an object we present an iterative strategy (Section 4) that repeatedly performs a sequence of basic motions in order to incrementally "grow" the set of points made visible on an object surface. These motions are designed in a way that guarantees (1) they will be executed only a finite number of times, and (2) when the process terminates the surface will be completely explored.

Clearly, the robot's motions at each iteration of the exploration process depend on the robot's capabilities and the sensor

[1] Visible portions of an object surface contain all points for which the open line segment connecting them to the robot does not intersect any object. In our discussion, "visual sensors" include cameras, where only information about the *projected* shape of the visible surfaces is directly available, as well as more powerful mechanisms such as range sensors, which directly provide distances to visible surface points within a finite ball around the robot.

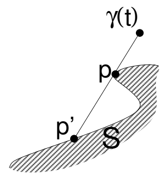


Figure 2: Points p and p' belong to the occlusion boundary. Only point p belongs to the visible rim.

input. These motions determine the class of objects that the robot can provably explore. To study the trade-offs involved in the design of an exploration strategy we consider two different types of basic motions. The first type uses a sensor capable of providing the three-dimensional coordinates of all visible points (e.g., a range sensor), and requires the robot to move very close to an object at every iteration (Section 5). With these capabilities we show that arbitrary objects can be explored. The second type of basic motions uses a camera that provides only the shape of the *projection* of the occlusion boundary, and requires the robot to get close to the surface only when the surface geometry makes this necessary (Section 6). In this case we show that the resulting exploration strategy can only be used to explore non-concave objects or the non-concave regions of a general object. We then briefly discuss the trade-offs in these two exploration strategies, and also describe how they can be combined (Section 7).

3 Visibility and Occlusion

In order to solve the exploration problem in an unknown environment one must understand the structure of the occlusion boundary, its relation to object geometry, and the way its structure changes when the robot moves. This section presents the basic concepts required for this analysis.

Without loss of generality, we consider in rest of the paper the case where the environment contains a single object whose surface S is connected, has finite area, and bounds an open volume V_S in $\Re^3$. Furthermore, we limit our discussion to the case where S is smooth; for these surfaces every surface point p has a uniquely-defined outward surface normal, $\mathbf{n}(p)$, and tangent plane, $T_p(S)$, both of which depend smoothly on p.[2]

Given a robot path $\gamma(t) \in \Re^3 - V_S, 0 \leq t \leq T$, a point $p \in S$ is *occluded* from position $\gamma(t)$ if and only if the line segment $p\gamma(t)$ intersects V_S; otherwise p is visible.

[2]The strategies we present can be extended to handle piecewise-smooth surfaces. These surfaces are smooth except for zero- or one-dimensional sets whose points have a finite number of normals assigned to them.

Definition 3.1 The *visible rim*, $\mathfrak{R}_{\gamma(t)}$, is the set of points p for which the line segment $p\gamma(t)$ is tangent to S at p and does not intersect V_S.

Definition 3.2 The *occlusion boundary*, $\mathfrak{O}_{\gamma(t)}$, bounds the visible and the occluded points. It is the set of points p for which the line segment $p\gamma(t)$ contains at least one visible rim point and does not intersect V_S.

Clearly, the occlusion boundary is a superset of the visible rim (Figure 2). A convenient way to visualize and study the occlusion boundary and the visible rim is to look at their spherical projections onto an image centered at the robot's position (Figure 3). Their projections are identical; this projected contour is known as the *occluding contour* [1].

In general, the occlusion boundary is collection of closed, piecewise-smooth curves, $\alpha_1, \ldots, \alpha_n$, whose number and shape depends on the robot's position. The endpoints of the smooth segments of these curves project to cusps or T-junctions on the occluding contour (Figure 3). Given a smooth path $\gamma(t), 0 < t < T$, for the robot, the curves α_i move on the surface in a way that depends smoothly on t, except for a discrete set of values for which their topology changes. These changes occur precisely when the topology of the occluding contour changes [13–15].

Between any two consecutive topological transitions the occlusion boundary can be thought of as a collection of smooth curves that "slide" over the surface of the object as the robot moves, causing deformations to the occluding contour. In particular, given an instantaneous direction of motion, $\gamma'(t)$, we can express the motion of the occlusion boundary using the *epipolar parameterization*. The details of this parameterization are not important here and the reader is referred to [16]. The important point is that given a smooth segment $\beta(t)$ of

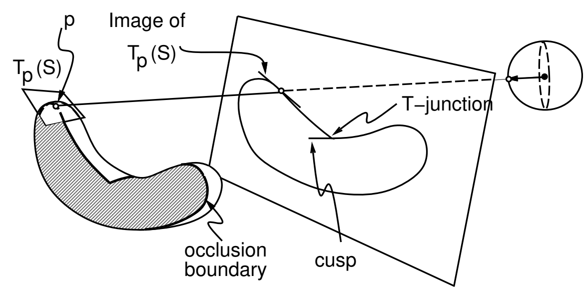


Figure 3: The occluding contour. Each point p on the occlusion boundary projects to a single point on the spherical image. For simplicity, the image of the occlusion boundary on a plane tangent to the spherical image is shown.

$\mathfrak{O}_{\gamma(t)}$, this parameterization allows us to define the segment $\beta(t+\delta t)$ of $\mathfrak{O}_{\gamma(t+\delta t)}$ that corresponds to $\beta(t)$.

The motion of the occlusion boundary induced by the robot's motion directly affects the set of surface points becoming visible. When the occlusion boundary at time T slides over a surface point that was occluded from positions $\gamma(t), 0 \leq t < T$, that point becomes visible. The following theorem (see [17] for a proof) gives a qualitative characterization of these visibility transitions (Figure 4):

Theorem 3.1 *Let $p \in \mathfrak{O}_{\gamma(t)}$ and let q_p be a point of tangency of the segment $p\gamma(t)$ with S. If $\beta(t)$ is a smooth occlusion boundary segment whose points p satisfy $\mathbf{n}(q_p) \cdot \gamma'(t) > 0$, all points on $\beta(t+\delta t)$ are occluded from position $\gamma(t)$. Conversely, if $\mathbf{n}(q_p) \cdot \gamma'(t) < 0$, all points on $\beta(t+\delta t)$ are visible from position $\gamma(t)$.*

Once a point becomes visible, it may be possible to recover its three-dimensional coordinates using a visual sensor. We consider two types of sensors:

- *Range sensors*. With such a sensor the robot can determine the coordinates of all visible points.
- *Cameras*. With such a sensor the robot can detect the occluding contour. Furthermore, when the robot moves along a path γ and knows its speed and acceleration, it can determine the coordinates of all points on the visible rim from the deformation of the occluding contour [16].

The exploration problem can now be formulated as follows:

Definition 3.3 (Range sensor-based exploration) Given a connected surface S and an initial position $\mathbf{s} \in \Re^3 - V_S$, generate a finite-length path γ such that the occlusion boundary slides over all points on S that were occluded at $\mathbf{s}$. In other words, if U is the set of points occluded at $\mathbf{s}$ and $\gamma = \{\gamma(t), 0 \leq t \leq T\}$, γ must satisfy $\bigcup_{0 \leq t \leq T} \mathfrak{O}_{\gamma(t)} \supseteq U$.

Definition 3.4 (Camera-based exploration) Given a connected surface S and an initial position $\mathbf{s} \in \Re^3 - V_S$, generate a finite-length path γ such that if $\gamma = \{\gamma(t), 0 \leq t \leq T\}$, γ must satisfy $\bigcup_{0 \leq t \leq T} \mathfrak{R}_{\gamma(t)} = S$.

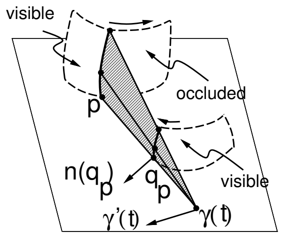


Figure 4:

To facilitate our discussion we use the following definition:

Definition 3.5 A point p on S is *explored* if and only if the three-dimensional coordinates of p were determined by the robot from some position along its path. The boundary of the explored points is the *exploration frontier*.

4 Incremental Exploration

Having provided a formulation for the exploration problem, this section considers a strategy for exploring an object of arbitrary shape. The idea is to control the robot's motion so that the set of explored points incrementally "grows" on the surface. To achieve this objective, the robot performs the following actions during the n-th iteration: It selects a point p_n on the exploration frontier, moves to a position where it is visible, and finally performs a sequence of motions forcing all points in a neighborhood of p_n to be explored. This process terminates when the set of explored points is identical to the object's surface, i.e., when the exploration frontier degenerates to a collection of isolated points.

To completely specify our exploration strategy we need to answer four questions: (1) How to select point p_n, (2) how to find a position where p_n is visible, (3) how to explore a neighborhood of p_n, and (4) how to guarantee that only a finite number of iterations are necessary. The crucial point in our approach is that the third and fourth questions, which lie at the heart of the exploration problem, can be answered using a *local* strategy, i.e., one that does not depend on the history of the exploration process (the set of points explored and the path generated in the previous iterations). We discuss each of the four questions below.

In our approach, the point p_n is simply chosen to be any point on the exploration frontier. The coordinates of the points comprising the frontier are available to the robot because the exploration frontier bounds the set of explored points.

The answer to the second question is also simple. If the robot's sensor is a camera, all points on the exploration frontier were points on the visible rim at some previous position of the robot; if it is a range-sensor, all points on the exploration frontier were points on the occlusion boundary at some previous position. Hence, it suffices for the robot to retrace its path back to a position where the selected point p_n was on the occlusion boundary or the visible rim. This can be achieved if, along its path, the robot saves its current position together with the set of points belonging to the visible rim (or the occlusion boundary, if a range sensor is used) at that position.[3] This information allows the robot to (1) associate

[3] It is, in fact, possible to show that the coordinates of only a discrete set of occlusion boundary or visible rim points need to be stored from each position.

with each exploration frontier point a set of previous positions from which that point is visible, and (2) plan a collision-free path to any such position.

Now suppose that the selected point is visible. We use a *neighborhood exploration strategy* to answer the third question. The only input to this strategy is the selected point p_n. The goal is to generate a path, based on information provided by the robot's sensor, such that an open neighborhood W_n of p_n becomes explored.

In order to explore the surface in a finite amount of time, the neighborhood W_n must be large enough so that only a finite number of such neighborhoods is necessary to cover the whole surface of the object.

Definition 4.1 Given a class of surfaces $\mathcal{S}$, a neighborhood exploration strategy is *complete for* $\mathcal{S}$ if and only if for all surfaces $S \in \mathcal{S}$ and all sequences $(p)_n, p_n \in \partial\left(\bigcup_{k\leq n} W_k\right)$, the sequence $(W)_n$ of explored neighborhoods is such that $\bigcup_{n\leq K} W_n = S$ for some finite K.

Equipped with a complete exploration strategy $\mathcal{A}$ for $\mathcal{S}$, the robot uses the following strategy to explore a surface $S \in \mathcal{S}$:

Surface Exploration Strategy

Step 1: If a range sensor is available, select a point p on the occlusion boundary; otherwise select p on the visible rim. (Initially the exploration frontier is identical to the occlusion boundary.)

Step 2: Use $\mathcal{A}$ to explore a neighborhood of p. During this step, the set of points explored on the surface is expanded.

Step 3: Select any point p' on the current exploration frontier. If no such point exists, stop; the exploration process is complete.

Step 4: If a range sensor is available, move to a previous position, $\gamma(t_0)$ at which p' is on the occlusion boundary; otherwise move to a previous position $\gamma(t_0)$ where p' is on the visible rim.

Step 5: Repeat steps 2-5.

The only step left unspecified in the above strategy is Step 2. The next two sections describe two neighborhood exploration strategies that can be used to implement this step. These strategies give rise to two different solutions to the exploration problem. The first strategy, $\mathcal{A}_r$, which is the most general, uses a range sensor. It is complete for the class of smooth and connected surfaces of finite area (Section 5). The second one, $\mathcal{A}_c$, uses a camera and is complete only for the subclass of surfaces with no concavities, e.g., the torus or, more generally, surfaces produced by appropriately deforming spheres with h handles (Section 6).

5 Exploration Using a Range Sensor

Suppose that the robot needs to explore the neighborhood of a selected occlusion boundary point p_n on S with the help of a range sensor. Strategy $\mathcal{A}_r$ is based on the following observation: If the robot looks at p_n from the direction of its surface normal and is positioned close enough to p_n, all points in a neighborhood of p_n must be visible. Intuitively, the reason for this is that the smoothness of S around p_n severely constrains the variation of the surface normals in the point's neighborhood; this limits the sets of points that can be occluded when the robot is close to p_n along $\mathbf{n}(p_n)$.

Using this observation, strategy $\mathcal{A}_r$ prescribes exactly how the robot should move to reach a position along $\mathbf{n}(p_n)$, while at the same time guaranteeing the completeness of the strategy for arbitrary smooth surfaces. In particular, the strategy specifies (1) how to reach positions along $\mathbf{n}(p_n)$, and (2) how far away from p_n the robot should go.

Reaching a position close to p_n along $\mathbf{n}(p_n)$ is trivial. Since p_n is visible from the robot's initial position, the robot moves along a straight line towards p_n, determines the surface normal at p_n when p_n is reached, and then moves along that normal.

The following lemma shows that after p_n is reached, the robot can guarantee that a neighborhood of p_n is explored by performing an arbitrarily small position adjustment along $\mathbf{n}(p_n)$. See [18] for the proof.

Lemma 5.1 *Let $\mathcal{E}(d)$ be the set of visible points when the robot is at distance d from p_n along $\mathbf{n}(p_n)$. Then, there is an $\epsilon > 0$ such that $\mathcal{E}(d)$ contains a neighborhood of p_n for all $0 < d < \epsilon$.*

Unfortunately, an arbitrarily small motion away from p_n can leave $\mathcal{E}(d)$ arbitrarily small. The robot must therefore move sufficiently far from p_n so that $\mathcal{E}(d)$ contains a neighborhood of p_n that is large enough to guarantee completeness. In [18] we show that the robot can guarantee completeness by simply moving away from p_n along $\mathbf{n}(p_n)$ until either a collision with the surface does not allow further motion, or an *a priori* defined distance $\Delta > 0$ is reached. Furthermore, the selection of Δ does not affect the completeness of the strategy. This result follows from the smoothness of S, which ensures that if the robot moves away from p_n along $\mathbf{n}(p_n)$, the distance traveled before a surface collision occurs cannot be arbitrarily small.

The above strategy can be improved by adding an additional stopping condition that reduces the amount of robot motion away from p_n. This condition uses Theorem 3.1 and requires the robot to monitor the occlusion boundary while moving away from p_n. In particular, during the robot's motion away from p_n its instantaneous velocity, $\gamma'(t)$, is along $\mathbf{n}(p_n)$. The occlusion boundary curves at any time t during this motion can be partitioned into open segments falling into two categories,

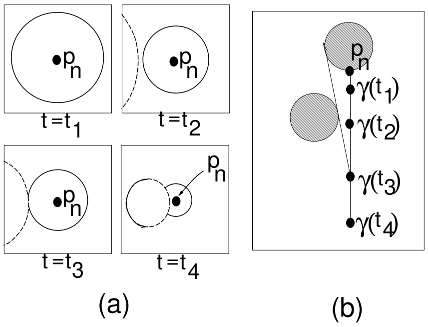


Figure 5: Executing strategy $\mathcal{A}_r$. (a) Views of an environment with two spheres, as the robot moves away from point p_n on the sphere shown in the first image, along $\mathbf{n}(p_n)$. The solid curves correspond to occlusion boundary points in B^+ and the dashed curves to points in B^-. Condition (c) in Step 3 forces the robot to stop at time t_3. (b) Top view of a horizontal plane containing p_n and $\mathbf{n}(p_n)$.

B^+ and B^-, whose points r satisfy $\gamma'(t) \cdot r > 0$ and $\gamma'(t) \cdot r < 0$, respectively. If d^+, d^- are the distances of B^+ and B^- from p_n, respectively, the robot can stop moving when $d^- < d^+$ (Figure 5).

Intuitively, this constraint allows the robot to move away from p_n as long as the set of visible points near p_n strictly expands. To see this, note that according to Theorem 3.1, the segments in B^+ slide over previously-occluded points on S, expanding the set of points that are visible in p_n's neighborhood, while the segments in B^- cause a contraction of this set. The distance $d = \min\{d^+, d^-\}$ determines the size of the visible neighborhood of p_n, and increases monotonically as long as $d = d^+$.

Given that p_n is on the occlusion boundary from the robot's position, $\gamma(t_0)$, the following neighborhood exploration strategy can now be specified:

Strategy $\mathcal{A}_\mathbf{r}$

Step 1: Starting from $\gamma(t_0)$, move on a straight line to p_n.

Step 2: Determine the surface normal at p_n, $\mathbf{n}(p_n)$.

Step 3: Move in direction $\mathbf{n}(p_n)$, until one of the following conditions is satisfied:

- **a.** The surface obstructs the robot's motion.
- **b.** An *a priori* defined distance Δ from p_n is reached.
- **c.** $d^- < d^+$, where d^+, d^- are the distances of B^+ and B^- to p_n, respectively.

The following theorem establishes the correctness of the strategy. The interested reader is referred to [18] for the proof. It is based on the fact that the surface is smooth and does not have infinite curvature at p_n.

Theorem 5.1 *Strategy $\mathcal{A}_r$ is complete for smooth and connected surfaces of finite area.*

6 Exploration Using a Camera

This section considers an alternative strategy called $\mathcal{A}_c$. The strategy has three characteristics: (1) It uses a camera instead of a range sensor, (2) it moves the robot close to the selected point p_n only when this is dictated by the shape of the object's surface, and (3) it is complete for a subset of the surfaces handled by strategy $\mathcal{A}_r$, namely those that are non-concave and generic.[4] When the robot's sensor can provide information only about the shape of the occluding contour, $\mathcal{A}_c$ allows the exploration of almost all surfaces that can be explored with such a sensor: It is easy to see that points in a surface concavity cannot belong to the visible rim, and hence cannot become explored by *any* strategy using only information derived from the occluding contour.

Strategy $\mathcal{A}_c$ is based on the following simple observation: Suppose the robot is positioned level with a point on a hill top; then this point will belong to the visible rim. By moving up or down, the robot will see just over the hill or not quite to the top, respectively. These motions cause the visible rim to slide over a neighborhood of that point, according to Theorem 3.1. In general, this motion corresponds to motion on the plane Π defined by the selected point p_n, the surface normal $\mathbf{n}(p_n)$, and the robot's position [17]. Strategy $\mathcal{A}_c$ uses this observation to explore a neighborhood W_n of p_n while at the same time guaranteeing that W_n is large enough to ensure completeness.

To achieve this, consider why W_n can be arbitrarily small. There are two possibilities:

- The size of W_n depends on the length of the visible rim curve containing p_n and the distance of its endpoints from p_n, if such endpoints exist. Depending on the robot's position, c, these lengths can be arbitrarily small; they can also become arbitrarily small because of a topological change in the visible rim when c is infinitesimally perturbed on Π.

- The size of W_n is constrained by the amount of robot motion on Π.

[4] A precise definition of generic surfaces is beyond the scope of this paper, and the reader is referred to [19]. Intuitively, generic surfaces are surfaces whose geometrical characteristics do not change if they are infinitesimally perturbed. All smooth surfaces (except for a zero-measure set) are generic.

The main idea of strategy $\mathcal{A}_c$ is to exploit the following result, which ensures the length of the visible rim curve containing p_n never becomes arbitrarily small:

Proposition 6.1 *Let S be a smooth, non-concave, generic surface. Given a sequence of points, $(p)_n \in S$, and a sequence of positions, $(c)_n$, from which p_n is on the visible rim, there exists a sequence of positions, $(\hat{c})_n$, satisfying the following four conditions:*

1. *$\hat{c}_n$ is on $T_{p_n}(S)$ and is reachable from c_n.*
2. *p_n is visible from $\hat{c}_n$.*
3. *There exists an $\epsilon_1 > 0$ such that the length of the visible rim curve, β_n, containing p_n at c_n is larger than ϵ_1 for all n.*
4. *There exists an $\epsilon_2 > 0$ such that for all n and all infinitesimal perturbations, c, of $\hat{c}_n$ on the plane containing p_n and $\mathbf{n}(p_n)$, the distance between the endpoints of the visible rim curve containing p_n at c is greater than ϵ_2.*

Given a point p_n, $\mathcal{A}_c$ determines a set of positions, $\Sigma_n = \{\hat{c}_n^1, \ldots, \hat{c}_n^{k_n}\}$, such that at least one of the resulting sequences $(\hat{c})_n^i$ satisfies Proposition 6.1 (Section 6.1).

Once Σ_n is determined, ensuring that W_n never diminishes is fairly simple. If Π is the plane defined by $\hat{c}_n^i, p_n$, and $\mathbf{n}(p_n)$, it suffices for the robot to move on a circle in Π centered at p_n of radius $\|\hat{c}_n^i - p_n\|$. Because the robot can collide with the surface, this motion also involves moving closer to p_n along a curve in the intersection $S \cap \Pi$, if such a collision occurs. Due to lack of space a description of this strategy is omitted. The interested reader is referred to [18] for the details.

Given that p_n is on the visible rim from the robot's position, $\gamma(t_0)$, the following neighborhood exploration strategy can now be specified:

Strategy $\mathcal{A}_\mathbf{c}$

Step 1: Starting from $\gamma(t_0)$, plan a path to determine a set of positions $\Sigma_n = \{\hat{c}_n^1, \ldots, \hat{c}_n^{k_n}\}$ on $T_{p_n}(S)$, such that for at least one $i, i = 1, \ldots, k_n$, $(\hat{c})_n^i$ satisfies Proposition 6.1.

Step 2: For each $i = 1, \ldots, k_n$, (a) move to $\hat{c}_n^i$, and (b) starting from $\hat{c}_n^i$, move in the plane defined by $\hat{c}_n^i, p_n$, and $\mathbf{n}(p_n)$ to explore a neighborhood of p_n.

Theorem 6.1 *Strategy $\mathcal{A}_c$ is complete for smooth, non-concave, generic, and connected surfaces of finite area.*

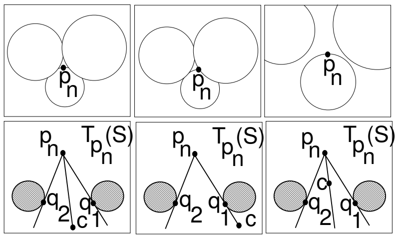


Figure 6: *Top row:* Views of three spheres from three positions on $T_{p_n}(S)$. *Bottom row:* Top views of $T_{p_n}(S)$. Shaded areas correspond to the intersection of $T_{p_n}(S)$ with the three spheres. Each view in the top row corresponds to the robot position c in the view of $T_{p_n}(S)$ below it.

6.1 Positioning the Robot on $\mathbf{T_{p_n}(S)}$

This section briefly describes how to generate a path, starting from an initial position $\gamma(t_0)$, that allows determination of the set Σ_n. The main idea in this process is that every position, $\hat{c}_n^i \in \Sigma_n$, can be characterized by two geometric relationships: (1) The angle formed by segment $\hat{c}_n^i p_n$ and the *asymptotes* [20] and *bitangents*[5] at p_n, and (2) the distance between $\hat{c}_n^i$ and p_n. In the following we show how these two relationships can be used to determine Σ_n. This essentially reduces the determination of Σ_n to the determination of the bitangents and asymptotes at p_n. The following provides an intuitive explanation for the selection process through an example; for formal proofs and more details see [17, 18].

Suppose p_n is on the visible rim when the robot is positioned at c. The angle between segment cp_n and the asymptotes and bitangents affects the distance between p_n and the endpoints of the visible rim curve containing it. To see this, consider the example in Figure 6. By moving the robot close to the bitangent line through p_n and q_1 (without changing its distance to p_n), the right T-junction endpoint of the visible rim curve containing p_n can get arbitrarily close to p_n. To avoid this, c must be such that the segment cp_n forms a large angle with both bitangent lines through p_n.

Similarly, the distance between c and p_n affects the length of the visible rim curve containing p_n. If the robot moves close enough to p_n in the example of Figure 6, the sphere containing p_n will not be occluded by the other two spheres. On the other hand, moving too close to p_n would cause the visible rim curve containing p_n to be arbitrarily short. This implies that $\hat{c}_n^i$ should be closer to p_n than either q_1 or q_2, but

[5] The bitangents at p_n are the lines that are tangent to S at at least two points, one of which is p_n.

should not be arbitrarily close to p_n.

A generalization of the above for the case of a general smooth, non-concave and generic surface S leads to the following result. We only present it here for the case where p_n is elliptic; a similar result holds for the case of parabolic and hyperbolic points.

Theorem 6.2 *Let $B_1, \ldots, B_{k_n}$ be the bitangent half-lines with origin p_n on $T_{p_n}(S)$. Let $q_i^1, \ldots, q_i^{l_i}$ be the points of tangency of B_i with S for which the segments $q_i^l p_n (l = 1, \ldots, l_i)$ do not intersect V_S. Then, we can construct a set Σ_n of positions that is uniquely determined by the half-lines B_i, the distances $\|q_i^l - p_n\|$, and an* a priori *defined constant $M > 0$, and contains at least one position $\hat{c}_n^i$ for which the sequence $(\hat{c})_n^i$ satisfies Proposition 6.1. Furthermore, the size of Σ_n is at most $k_n + 4\sum_{i=1}^{k_n} l_i$.*

Therefore, to construct Σ_n, it suffices to determine the bitangents at p_n and their contacts with S that are visible from p_n. This can be provably achieved by circumnavigating p_n on $T_{p_n}(S)$, until the robot's initial position, $\gamma(t_0)$ is reached, while maintaining the visibility of p_n [17, 18]. During this process, the robot moves closer to p_n only to maintain the visibility of p_n, or when surface collisions prevent the robot from maintaining a fixed distance from p_n.

7 Discussion

The two strategies developed in the previous sections demonstrate that a number of trade-offs are involved when deciding on a strategy to explore an object. Information about object geometry (e.g., whether or not concavities exist) allows the robot to use simpler sensors and allows the robot to stay farther away from the object being explored.

An even more general consequence that follows from our analysis is that the robot can explore different parts of the *same* object in different ways. When multiple strategies are available, a crucial issue is how to decide which strategy to use.

As an example, consider how strategies $\mathcal{A}_r$ and $\mathcal{A}_c$ can be combined. The robot can use a camera to explore from a distance any connected, non-concave region of a surface, even though that surface may also contain concavities. This is because the convergence properties of $\mathcal{A}_c$ change only when the robot attempts to explore the surface near the boundary of such a region [17]: When the robot uses $\mathcal{A}_c$ to explore the surface near a parabolic curve bounding a surface concavity, the set of points explored near the parabolic curve diminishes as the parabolic curve is approached. Consequently, the robot can keep using $\mathcal{A}_c$ until the *rate* of points explored (e.g., the area of newly-explored points) falls below an *a priori* defined threshold. Within such a framework, the robot will have to use $\mathcal{A}_r$ only when necessary, i.e., when exploring close to concavities. An important future direction of our research will be to develop a better understanding of the interaction between these and other exploration strategies.

References

[1] J. J. Koenderink, *Solid Shape*. MIT Press, 1990.

[2] J. H. Rieger, "On the classification of views of piecewise smooth objects," *Image and Vision Computing*, vol. 5, no. 2, pp. 91–97, 1987.

[3] J. H. Reif, "Complexity of the mover's problem and generalizations," in *Proc. Twentieth Symposium on Foundations of Computer Science*, pp. 421–427, 1979.

[4] R. A. Brooks, "A robust layered control system for a mobile robot," *IEEE J. Robotics Automat.*, vol. 2, no. 1, pp. 14–23, 1986.

[5] Y. Aloimonos, "Purposive and qualitative active vision," in *Proc. Int. Conf. on Pattern Recognition*, pp. 346–360, 1990.

[6] V. J. Lumelsky, S. Mukhopadhyay, and K. Sun, "Dynamic path planning in sensor-based terrain acquisition," *IEEE Trans. Robotics Automat.*, vol. 6, no. 4, pp. 462–472, 1990.

[7] K. N. Kutulakos, V. J. Lumelsky, and C. R. Dyer, "Vision-guided exploration: A step toward general motion planning in three dimensions," in *Proc. IEEE Robotics Automat. Conf.*, pp. 289–296, 1993.

[8] A. Blake, A. Zisserman, and R. Cipolla, "Visual exploration of free-space," in *Active Vision* (A. Blake and A. Yuille, eds.), pp. 175–188, MIT Press, 1992.

[9] N. S. Rao, S. S. Iyengar, B. J. Oommen, and R. Kashyap, "Terrain acquisition by point robot amidst polyhedral obstacles," in *Proc. Third Conf. on Artificial Intelligence Applications*, pp. 170–175, 1987.

[10] C. I. Connoly, "The determination of next best views," in *Proc. IEEE Robotics Automat. Conf.*, pp. 432–435, 1985.

[11] J. Maver and R. Bajcsy, "Occlusions as a guide for planning the next view," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, no. 5, pp. 417–433, 1993.

[12] P. Whaite and F. P. Ferrie, "From uncertainty to visual exploration," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, no. 10, pp. 1038–1049, 1991.

[13] J. J. Koenderink and A. J. van Doorn, "The singularities of the visual mapping," *Biological Cybernetics*, vol. 24, pp. 51–59, 1976.

[14] J. Ponce, S. Petitjean, and D. Kriegman, "Computing exact aspect graphs of curved objects: Algebraic surfaces," in *Proc. Second European Conference on Computer Vision*, 1992.

[15] J. H. Rieger, "The geometry of view space of opaque objects bounded by smooth surfaces," *Artificial Intelligence*, vol. 44, pp. 1–40, 1990.

[16] R. Cipolla and A. Blake, "Surface shape from the deformation of apparent contours," *Int. J. Computer Vision*, vol. 9, no. 2, pp. 83–112, 1992.

[17] K. N. Kutulakos and C. R. Dyer, "Global surface reconstruction by purposive control of observer motion," Tech. Rep. 1141, Computer Sciences Department, University of Wisconsin— Madison, April 1993. Available via ftp from ftp.cs.wisc.edu.

[18] K. N. Kutulakos, V. J. Lumelsky, and C. R. Dyer, "Provable strategies for vision-guided exploration in three dimensions," tech. rep., Computer Sciences Department, University of Wisconsin— Madison, 1993. To appear.

[19] Y. L. Kergosien, "La famille des projections orthogonales d'une surface et ses singularités," *C. R. Acad. Sc. Paris*, vol. 292, pp. 929–932, 1981.

[20] M. P. D. Carmo, *Differential Geometry of Curves and Surfaces*. Englewood Cliffs, NJ: Prentice-Hall Inc., 1976.