Simultaneous Precise Solutions to the Visibility Problem of Sculptured Models

Joon-Kyung Seong\textsuperscript{1}, Gershon Elber\textsuperscript{2}, and Elaine Cohen\textsuperscript{1}

\textsuperscript{1} University of Utah, Salt Lake City, UT84112, USA
seong@cs.utah.edu, cohen@cs.utah.edu
\textsuperscript{2} Technion, Haifa 32000, Israel
gershon@cs.technion.ac.il

Abstract. We present an efficient and robust algorithm for computing continuous visibility for two- or three-dimensional shapes whose boundaries are NURBS curves or surfaces by lifting the problem into a higher dimensional parameter space. This higher dimensional formulation enables solving for the visible regions over all view directions in the domain simultaneously, therefore providing a reliable and fast computation of the visibility chart, a structure which simultaneously encodes the visible part of the shape’s boundary from every view in the domain. In this framework, visible parts of planar curves are computed by solving two polynomial equations in three variables ($t$ and $r$ for curve parameters and $\theta$ for a view direction). Since one of the two equations is an inequality constraint, this formulation yields two-manifold surfaces as a zero-set in a 3-D parameter space. Considering a projection of the two-manifolds onto the $t\theta$-plane, a curve’s location is invisible if its corresponding parameter belongs to the projected region. The problem of computing hidden curve removal is then reduced to that of computing the projected region of the zero-set in the $t\theta$-domain. We recast the problem of computing boundary curves of the projected regions into that of solving three polynomial constraints in three variables, one of which is an inequality constraint. A topological structure of the visibility chart is analyzed in the same framework, which provides a reliable solution to the hidden curve removal problem. Our approach has also been extended to the surface case where we have two degrees of freedom for a view direction and two for the model parameter. The effectiveness of our approach is demonstrated with several experimental results.

1 Introduction

A major part of rendering is related to the hidden surface removal problem, i.e., display only those surfaces which should be visible. The main contribution of this work can be summarized as follows:

- The exact boundary between visible and hidden parts of planar curves or surfaces is computed by solving a set of polynomial equations in the parameter space without any piecewise linear approximations.
- All possible view directions in the domain are considered, simultaneously, by lifting the problem into a higher dimensional space and solving a continuous visibility problem. This higher dimensional framework provides a reliable solution to the computation of the visibility chart.
The algorithm is easy to implement and robust by mapping the problem in hand to a zero-set solving that exploits the convex hull and subdivision properties of NURBS. Topological analysis of the visibility chart makes it easier to compute the global structure of the visibility chart.

Research into solving the hidden surface removal problem is one of the earliest areas of activity in computer graphics, computer-aided design and manufacturing, and many different algorithms have been developed [24,19,18,14,19]. Usually they are developed for polygonal data, so curved surfaces have traditionally been preprocessed and approximated as large collections of polygons [22,17]. In this paper, we present an algorithm for eliminating hidden curves or surfaces directly from freeform models without any polygonal approximations. Visibility computations of sculptured models have various applications not only in the area of rendering but also in such areas as mold design, robot accessibility, inspection planning and security.

Given a view direction, the hidden surface removal problem refers to determining which surfaces are occluded from that view direction. Most of the earlier algorithms in the literature are for polygonal data and hidden line removal [8,20,24]. In their work, because the displayed edges of the polygons are linear edges, the displayed curves, such as the silhouettes of an object viewed from a view direction, are not smooth. Curves can be displayed more smoothly by increasing the number of polygons used for the approximation, but this results in memory and computational expense.

Algorithms to resolve the hidden surface removal problem can be classified into those that perform calculations in object-space, those that perform calculations in image-space, and those that work partly in both, list-priority [24]. Object space techniques use geometric tests on the object descriptions to determine which objects overlap and where. Initiated by Appel’s edge-intersection algorithm [1], the idea of quantitative invisibility which determines visible and invisible regions in advance was developed [9,18,11]. Image space approaches compute visibility only to the precision required to decide what is visible at a particular pixel, exemplified by [2]. Catmull develops the depth-buffer or z-buffer image-precision algorithm which uses depth information [4]. Also, Weiler and Atherton [25] and Whitted [26] develop ray tracing algorithms which transform the hidden surface removal problem into ray-surface intersection tests.

Given a model composed of algebraic or parametric surfaces, it can be polygonized and hidden lines can be removed from the polygonized surfaces [22,17]. However, the accuracy of the overall algorithm is limited by the accuracy of the polygon approximation. Further, in both methods [22,17], visibility is determined for the endpoints of straight lines and hence, they fail to detect invisibility occurring in the interior region of a line when both endpoints are visible. To remove hidden lines from curved surfaces without polygonal approximation, Hornung et al. [11] extended the idea of quantitative invisibility to bi-quadratic patches, and Newton’s method was employed to solve for intersections between curves. Elber and Cohen [7] applied Hornung’s technique to nonuniform rational B-splines and extended it to treat trimmed surfaces. In particular, Elber and Cohen [7] extract the curves of interest by considering boundary curves, silhouette curves, iso-parametric curves and curves along $C^1$ discontinuity based on 2D curve-curve intersections. Nishita et al. [21] used their Bezier Clipping technique for the hidden curve elimination. These methods [11,7,21] are aimed at eliminating